



**Calhoun: The NPS Institutional Archive**  
**DSpace Repository**

---

Theses and Dissertations

1. Thesis and Dissertation Collection, all items

---

2008-12

# Breaking barriers to design dimensions in nearly orthogonal Latin hypercubes

Hernandez, Alejandro S.

Monterey California. Naval Postgraduate School

---

<https://hdl.handle.net/10945/10306>

---

*Downloaded from NPS Archive: Calhoun*



Calhoun is the Naval Postgraduate School's public access digital repository for research materials and institutional publications created by the NPS community. Calhoun is named for Professor of Mathematics Guy K. Calhoun, NPS's first appointed -- and published -- scholarly author.

**Dudley Knox Library / Naval Postgraduate School**  
**411 Dyer Road / 1 University Circle**  
**Monterey, California USA 93943**

<http://www.nps.edu/library>



**NAVAL  
POSTGRADUATE  
SCHOOL**

**MONTEREY, CALIFORNIA**

**DISSERTATION**

**BREAKING BARRIERS TO DESIGN DIMENSIONS IN  
NEARLY ORTHOGONAL LATIN HYPERCUBES**

by

Alejandro S. Hernandez

December 2008

Dissertation Supervisor:

Thomas W. Lucas

**Approved for public release; distribution is unlimited**

THIS PAGE INTENTIONALLY LEFT BLANK

<b>REPORT DOCUMENTATION PAGE</b>			<i>Form Approved OMB No. 0704-0188</i>
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.			
<b>1. AGENCY USE ONLY (Leave blank)</b>	<b>2. REPORT DATE</b> December 2008	<b>3. REPORT TYPE AND DATES COVERED</b> Dissertation	
<b>4. TITLE AND SUBTITLE:</b> Breaking Barriers to Design Dimensions in Nearly Orthogonal Latin Hypercubes		<b>5. FUNDING NUMBERS</b>	
<b>6. AUTHOR(S)</b> Alejandro S. Hernandez		<b>8. PERFORMING ORGANIZATION REPORT NUMBER</b>	
<b>7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)</b> Naval Postgraduate School Monterey, CA 93943-5000		<b>10. SPONSORING / MONITORING AGENCY REPORT NUMBER</b>	
<b>9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)</b> N/A		<b>11. SUPPLEMENTARY NOTES</b> The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.	
<b>12a. DISTRIBUTION / AVAILABILITY STATEMENT</b> Approved for public release; distribution is unlimited		<b>12b. DISTRIBUTION CODE</b> A	
<b>13. ABSTRACT (maximum 200 words)</b> A dynamic and extremely complex landscape in security and world events presents problems that challenge all sectors of society to develop efficient means for exploring a wide range of solutions. Similarly, exponential increases in technological capability make it difficult for commercial and governmental leaders to assess those proposed solutions. Computer experimentation is an established method for examining complex models with large numbers of factors. Orthogonal and nearly orthogonal Latin hypercubes are proven techniques for designing simulation experiments. A key property of these efficient, space-filling designs is their ability to explore many factors within a relatively modest number of design points; however, there is a limited inventory of these designs currently available. Those that have been catalogued are usually computationally expensive to produce and have severe restrictions in the number of factors and/or runs that they allow. To remedy this, we present a set of flexible methodologies to create design matrices with little or no correlation—including saturated nearly orthogonal Latin hypercubes. This new family of designs can explore as many factors as there are design points. This research also addresses experiments that include a mixture of continuous and integer variables, some of which have different numbers of value levels.			
<b>14. SUBJECT TERMS</b> Computer Experiments, Design of Experiments, Mixed Integer Programming, Latin Hypercube, Optimization, Nearly Orthogonal		<b>15. NUMBER OF PAGES</b> 158	<b>16. PRICE CODE</b>
<b>17. SECURITY CLASSIFICATION OF REPORT</b> Unclassified	<b>18. SECURITY CLASSIFICATION OF THIS PAGE</b> Unclassified	<b>19. SECURITY CLASSIFICATION OF ABSTRACT</b> Unclassified	<b>20. LIMITATION OF ABSTRACT</b> UU

THIS PAGE INTENTIONALLY LEFT BLANK

**Approved for public release; distribution is unlimited.**

**BREAKING BARRIERS TO DESIGN DIMENSIONS IN  
NEARLY ORTHOGONAL LATIN HYPERCUBES**

Alejandro S. Hernandez  
Colonel, United States Army  
B.S., United States Military Academy, 1985  
M.S., Naval Postgraduate School, 1994

Submitted in partial fulfillment of the  
requirements for the degree of

**DOCTOR OF PHILOSOPHY IN OPERATIONS RESEARCH**

from the

**NAVAL POSTGRADUATE SCHOOL  
December 2008**

Author:

---

Alejandro S. Hernandez

Approved by:

---

Thomas W. Lucas  
Associate Professor of  
Operations Research  
Dissertation Supervisor

---

Susan M. Sanchez  
Professor of Operations Research

---

Lyn R. Whitaker  
Associate Professor of  
Operations Research

---

Paul J. Sanchez  
Senior Lecturer of  
Operations Research

---

Craig W. Rasmussen  
Associate Professor of Mathematics

---

W. Matthew Carlyle  
Associate Professor of  
Operations Research

Approved by:

---

James N. Eagle, Chair, Department of Operations Research

Approved by:

---

Douglas Moses, Vice Provost for Academic Affairs

THIS PAGE INTENTIONALLY LEFT BLANK

## **ABSTRACT**

A dynamic and extremely complex landscape in security and world events presents problems that challenge all sectors of society to develop efficient means for exploring a wide range of solutions. Similarly, exponential increases in technological capability make it difficult for commercial and governmental leaders to assess those proposed solutions. Computer experimentation is an established method for examining complex models with large numbers of factors. Orthogonal and nearly orthogonal Latin hypercubes are proven techniques for designing simulation experiments. A key property of these efficient, space-filling designs is their ability to explore many factors within a relatively modest number of design points; however, there is a limited inventory of these designs currently available. Those that have been catalogued are usually computationally expensive to produce and have severe restrictions in the number of factors and/or runs that they allow. To remedy this, we present a set of flexible methodologies to create design matrices with little or no correlation—including saturated nearly orthogonal Latin hypercubes. This new family of designs can explore as many factors as there are design points. This research also addresses experiments that include a mixture of continuous and integer variables, some of which have different numbers of value levels.



THIS PAGE INTENTIONALLY LEFT BLANK

# TABLE OF CONTENTS

<b>I.</b>	<b>INTRODUCTION.....</b>	<b>1</b>
A.	<b>CHALLENGES IN EXPERIMENTAL DESIGNS FOR SIMULATIONS.....</b>	<b>1</b>
B.	<b>RANDOM LATIN HYPERCUBES AND INHERENT CORRELATION .....</b>	<b>4</b>
1.	<b>Random Latin Hypercube Generation .....</b>	<b>5</b>
2.	<b>Measure of Nonorthogonality for Design Selection .....</b>	<b>6</b>
C.	<b>A PROGRESSION OF OLH AND NOLH CONSTRUCTION METHODS.....</b>	<b>8</b>
D.	<b>DISSERTATION ORGANIZATION .....</b>	<b>11</b>
<b>II.</b>	<b>SHAPING CONDITIONS FOR APPLYING UNCORRECTED RANDOM LATIN HYPERCUBES.....</b>	<b>15</b>
A.	<b>TOOLS FOR DIMENSION SELECTION AND RANDOM LATIN HYPERCUBE GENERATION .....</b>	<b>16</b>
1.	<b>Creating the <math>\overline{\rho_{map}^{\min}}</math> Table.....</b>	<b>16</b>
2.	<b>A Function to Predict <math>\overline{\rho_{map}^{\min}}</math> .....</b>	<b>20</b>
a.	<i>Initial Analysis of <math>\overline{\rho_{map}^{\min}}</math> .....</i>	<b>20</b>
b.	<i>Exploring Simple Linear Regression Models for <math>\overline{\rho_{map}^{\min}}</math> .....</i>	<b>24</b>
c.	<i>A Multiple Linear Regression Model for <math>\overline{\rho_{map}^{\min}}</math> .....</i>	<b>24</b>
d.	<i>Adequacy of the Multiple Linear Regression Model for <math>\overline{\rho_{map}^{\min}}</math> .....</i>	<b>25</b>
e.	<i>Application of the <math>\left(\overline{\rho_{map}^{\min}}\right)^E</math> Model.....</i>	<b>26</b>
B.	<b>DESIGN DEVELOPMENT WITH CORRELATION-BASED TOOLS.....</b>	<b>27</b>
1.	<b>The Process for Dimension Selection and RLH Design Generation .....</b>	<b>27</b>
2.	<b>A Hypothetical Case Requiring an Uncatalogued Design Dimension .....</b>	<b>28</b>
C.	<b>EFFECTS OF FEWER CANDIDATE DESIGNS (G).....</b>	<b>30</b>
D.	<b>SUMMARY .....</b>	<b>32</b>
<b>III.</b>	<b>CORRECTED RLH USING FLORIAN’S METHOD FOR NEW DESIGNS...33</b>	
A.	<b>FLORIAN’S CORRELATION REDUCTION METHOD.....</b>	<b>33</b>
B.	<b>DIMENSIONALITY AND FLORIAN’S METHOD .....</b>	<b>34</b>
C.	<b>CONSTRUCTING NEARLY ORTHOGONAL LATIN HYPERCUBES .....</b>	<b>37</b>
D.	<b>SUMMARY .....</b>	<b>39</b>

<b>IV.</b>	<b>A MIXED INTEGER PROGRAMMING APPROACH TO MINIMIZE <math>\rho_{map}</math></b>	<b>..41</b>
<b>A.</b>	<b>SIMPLIFYING THE COMPUTATION FOR NONORTHOGONALITY</b>	<b>.....42</b>
<b>B.</b>	<b>A LINEAR FORMULATION OF THE MATHEMATICAL MODEL</b>	<b>...42</b>
	<b>1. An Initial Mathematical Model</b>	<b>.....43</b>
	<b>2. Transformation of a Nonlinear Optimization Problem</b>	<b>.....44</b>
	<b>3. A Complete Linear Formulation of the Optimization Problem</b>	<b>....45</b>
<b>C.</b>	<b>OPERATIONALIZING THE LINEAR OPTIMIZATION PROBLEM</b>	<b>.....47</b>
	<b>1. The Size and Complexity of VMIN</b>	<b>.....48</b>
	<b>2. Applying RLH and Florian’s Method to Initiate VMIN</b>	<b>.....51</b>
	<b>3. Iterative Application of VMIN</b>	<b>.....55</b>
<b>D.</b>	<b>NEW DESIGNS FROM VMIN</b>	<b>.....56</b>
<b>E.</b>	<b>SUMMARY</b>	<b>.....66</b>
<b>V.</b>	<b>MANAGING MIXED-FACTOR, MIXED-LEVEL EXPERIMENTS WITH A NEW DESIGN METHODOLOGY</b>	<b>.....69</b>
<b>A.</b>	<b>THE BASE CONTINUOUS DESIGN</b>	<b>.....69</b>
<b>B.</b>	<b>DISCRETE INTEGER VARIABLES AND STACKING METHODOLOGY</b>	<b>.....70</b>
	<b>1. One Subset of Discrete Variables with a Common Value Level</b>	<b>...70</b>
	<b>2. Addressing More Than One Subset of Discrete Variables</b>	<b>.....71</b>
	<b>3. Formalized Stacking Methodologies</b>	<b>.....72</b>
<b>C.</b>	<b>COMBINED AND STACKING METHODOLOGIES</b>	<b>.....75</b>
	<b>1. Steps for Creating a Mixed Factor, Mixed Level Design</b>	<b>.....76</b>
	<b>2. Design Best Practices within the Mixed Factor, Mixed Level Process</b>	<b>.....77</b>
<b>D.</b>	<b>EXPLORING FIRST RESPONSE TO A BOMB ATTACK USING A MIXED-FACTOR, MIXED-LEVEL DESIGN</b>	<b>.....77</b>
	<b>1. The Thesis Problem</b>	<b>.....77</b>
	<b>2. Applying the Mixed-Factor, Mixed-Level Methodology to a Thesis Question</b>	<b>.....79</b>
<b>E.</b>	<b>SUMMARY</b>	<b>.....84</b>
<b>VI.</b>	<b>THE COMPLETE FLEXIBLE OLH/NOLH METHODOLOGY: EXPLORING SOLDIER-LEVEL NETWORKS</b>	<b>.....85</b>
<b>A.</b>	<b>PROBLEM STATEMENT</b>	<b>.....85</b>
<b>B.</b>	<b>DESIGN OF EXPERIMENTS AND SIMULATION RUNS</b>	<b>.....86</b>
<b>C.</b>	<b>ANALYSIS</b>	<b>.....88</b>
<b>D.</b>	<b>WIDESPREAD UTILITY OF NEW NOLH DESIGNS</b>	<b>.....91</b>
<b>VII.</b>	<b>CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE STUDY</b>	<b>.....92</b>
	<b>BIBLIOGRAPHY</b>	<b>.....97</b>
<b>APPENDIX A.</b>	<b>SATURATED NOLH DESIGNS</b>	<b>.....103</b>
<b>APPENDIX B.</b>	<b>PAIRWISE PLOT OF FACTORS IN <math>N_{16}^{17}</math></b>	<b>.....111</b>

<b>APPENDIX C.</b>	<b>MAJOR ROGINSKI'S DESIGN .....</b>	<b>113</b>
<b>APPENDIX D.</b>	<b>MAJOR BAEZ'S MFML DESIGN .....</b>	<b>123</b>
<b>APPENDIX E.</b>	<b>A <math>\rho_{map}^{\min}</math> PREDICTIVE MODEL WITH QUADRATIC TERMS .....</b>	<b>125</b>
<b>INITIAL DISTRIBUTION LIST .....</b>		<b>129</b>

THIS PAGE INTENTIONALLY LEFT BLANK

## LIST OF FIGURES

Figure 1.	Three permutations of the vector $\underline{x}$ results in an RLH design for three variables and four sample design points using Owens' simplified formula to create a lattice LH. ....6
Figure 2.	In this RLH, columns two and three give $ \rho_{23}  = \rho_{map} = 1$ . ....7
Figure 3.	The sample standard deviation of $\rho_{map}^{\min}$ for 1,000 G200 $n$ and $k$ combinations is relatively small compared to its $\overline{\rho_{map}^{\min}}$ value. The largest standard deviation occurs in small designs, while the smallest deviations are for larger designs.....18
Figure 4.	A logarithmic pattern appears between $\overline{\rho_{map}^{\min}}$ and $k$ when $n$ is constant: Groups of $n$ are presented in increasing order. ....21
Figure 5.	An exponential decay or a negative power function relationship appears between $\overline{\rho_{map}^{\min}}$ and $n$ when $k$ is constant. For clarity, only a few sets for fixed $k$ are in the chart.....22
Figure 6.	There is a nearly linear relationship between $\overline{\rho_{map}^{\min}}$ and transformed $n$ . For illustration we show the graph for $k = 7$ . ....23
Figure 7.	A nearly linear relationship appears between $\overline{\rho_{map}^{\min}}$ and transformed $k$ . We illustrate the case for $n = 257$ .....23
Figure 8.	This residual plot of 115 different predicted $\overline{\rho_{map}^{\min}}$ values shows curvature.26
Figure 9.	This flow chart presents the combined process for matrix transformation and optimization of an MIP to construct an OLH or NOLH design.....60
Figure 10.	The structure of the stacked MFML design.....72
Figure 11.	Baltimore harbor vignette terrain: actual photo and within multiagent simulation (Roginski, 2006).....78
Figure 12.	Time in CRSP by convoy number per network type when the rate of convoy arrival is two convoys per hour (Baez, 2008). ....89
Figure 13.	Partition tree for mean time in CRSP, hierarchical network structure (Baez, 2008).....90
Figure 14.	Regression metamodel for mean time in CRSP, hierarchical network structure (Baez, 2008).....91
Figure 15.	The plot for the residuals resulting from a G200 MLR shows curvature that indicates a need to include more complex terms in the regression model.....126
Figure 16.	The residual plot for the new multiple linear regression model with quadratic terms still show curvature, but the sizes of the residuals are much smaller than from previous models. ....127

THIS PAGE INTENTIONALLY LEFT BLANK

## LIST OF TABLES

Table 1.	A comparison of recent construction methods to generate OLH or NOLH designs shows that our new method significantly increases the number of factors, $k$ , that may be examined for the same number of runs, $n$ . ..... xxii
Table 2.	An example design matrix for three factors (i.e., $k = 3$ ) and eight experiments (i.e., $n = 8$ ). .....2
Table 3.	A comparison of recent construction methods to generate OLH or NOLH designs shows that our new method significantly increases the number of factors, $k$ , that may be examined for the same number of runs, $n$ . .....11
Table 4.	This portion of the G200 table shows the $\overline{\rho_{map}^{\min}}$ for different design combinations where the number of designs generated in each trial is 200. We enter not applicable (NA) for design combinations where $n < k$ . .....19
Table 5.	We show a few different design combinations to compare the standard deviations of 1,000 G1000 and G200 RLHs. The greatest difference in standard deviations between G1000 and G200 is for the design, $n = 17$ , $k = 7$ , where G1000's standard deviation is smaller by 0.0039. ....19
Table 6.	A comparison of $\overline{\rho_{map}^{\min}}$ for 1000 G1000 and G200 shows that the differences are within the magnitude of the standard deviations shown in Table 5. ....20
Table 7.	Descriptive statistics from the residuals derived from the MLR predictions show negative skewness and a relatively small range. The model tends to slightly underestimate the value of the minimum maximum absolute pairwise correlation value of 200 RLH designs. ....25
Table 8.	This table is a section of a G200 $\overline{\rho_{map}^{\min}}$ table. It has values for dimensions with $k = 22$ or less. ....29
Table 9.	Calculations for $\left(\overline{\rho_{map}^{\min}}\right)^E$ using the MLR model as $n$ changes. ....30
Table 10.	The best maximum absolute pairwise correlation values from 20 FRLHs of the same design dimension are presented. Gray-shaded design dimensions emphasize intervals that this technique can fill in the OLH and NOLH catalogue, using Ang's (2006) near orthogonal threshold. Diagonal-lined areas identify dimensions yet to meet NOLH criteria. ....35
Table 11.	Comparison of $\rho_{map}$ values and CNs for RLH, $F_{16}^{65}$ , and $N_{16}^{65}$ .....37
Table 12.	These are the best maximum absolute pairwise correlation values from 20 FRLHs of the same design dimension. These designs are a continuation of Table 10. All of these design dimensions, as well as the design combinations between their intervals, fill many of the gaps in the OLH and NOLH catalogue. ....38



Table 13.	We present how the parameters of a design and the design’s initial values translate into the constraints in VMIN to optimize the values of the targeted variable column ( $m = 3, k = 4, n = 5$ ), where value levels for column 3 remain as variables and all other values are fixed. ....48
Table 14.	This $n = 17, k = 7$ design, $\rho_{map} = 0.066$ , is a result from generating numerous RLHs, selecting the design with the least nonorthogonality, and applying Florian’s method iteratively. It is the initial design before using VMIN.....53
Table 15.	Binary variable values for factor $m = 6$ in the initial design show the sparseness of this single column optimization problem. A set of these variables corresponds to each factor.....54
Table 16.	An excerpt from Cioppa and Lucas (2007) compares RLHs and Cioppa’s (2002) OLH and NOLH designs with respect to their orthogonal characteristics.....57
Table 17.	Results from a small design after iterative application of Florian’s method on the left-hand is not an OLH design. It can be used as the initial design for VMIN. The final design is orthogonal, as shown on the right-hand design.....58
Table 18.	The result from iterative application of Florian’s method on a randomly generated $n = 14, k = 7$ design is the initial design for VMIN, as shown on the left-hand side of the table. It does not meet NOLH orthogonality criteria. The final design for $n = 14, k = 7$ after VMIN is a new NOLH design, presented on the right-hand side.....61
Table 19.	The $N_{12}^{14}$ developed by RLH, FRLH, and VMIN emphasizes the flexibility of our methodology to construct any number of different NOLH designs (to include saturated ones). Note that none of the columns of this design are identical to the design on the right-hand side of Table 19. It shows that this new design is not merely an extension of the $n = 14, k = 7$ design, although our technique can certainly extend smaller designs.....62
Table 20.	A comparison of different approaches to develop a design that can explore 32 variables shows that our new S-NOLH design requires the least number of runs, and is therefore most efficient. Other approaches can theoretically develop experimental designs to explore 32 or more variables, but to the author’s knowledge, none are currently in the catalogue of OLH and NOLH designs. ....64
Table 21.	This is a new design from an original FRLH for 17 runs. We extend the number of factors to explore until it is fully saturated. It allows an experimenter to explore 16 factors within 17 runs. The $k:n$ ratio of 0.94 is large and makes the design a good screening plan. ....65
Table 22.	A summary of techniques from Cioppa, Ang, Steinberg and Lin, and our new method shows that the FRLH-MIP method is a viable option to extend the library of OLH and NOLH designs. A label of “NA” designates dimensions not currently available from the technique. ....66

Table 23.	Forty of the variables in Major Roginski’s experiment are displayed. The variables are all integers, but some have enough value levels to be considered as continuous, as the shaded rows indicate.....	80
Table 24.	Excerpt of the DV subset design for the MFML, containing 24 of the 144 total design points, shows how a full, two-level factorial design for three binary variables aligns with three design points of two discrete variables, each with three value levels. We shade sections of the table to emphasize the match between a single design point for the discrete, non-binary variables and a full two-level factorial for the binary variables. ....	82
Table 25.	A saturated design for $n = 16$ and $k = 15$ is the foundation for creating a base design for an MFML design. Its correlation measure is 0.0471, with a condition number of 1.319. ....	87

THIS PAGE INTENTIONALLY LEFT BLANK

## LIST OF SYMBOLS, ACRONYMS, AND/OR ABBREVIATIONS

CRSP	Centralized Receiving and Shipping Point
DHS	Department of Homeland Security
DoD	Department of Defense
DOE	Design of Experiment
FRLH	Florian-improved Random Latin Hypercube
GAMS	General Algebraic Modeling System
GB	Gigabyte
GHz	Gigahertz
ITV	In-Transit Visibility
LCM	Least Common Multiple
LCOP	Logistics Common Operating Picture
LH	Latin Hypercube
IWARS	Infantry Warrior Simulation
LHS	Latin Hypercube Sampling
MHPCC	Maui High Performance Computer Center
MIP	Mixed Integer Program
MFML	Mixed Factor, Mixed Level
NOLH	Nearly Orthogonal Latin Hypercube
OLH	Orthogonal Latin Hypercube
RAM	Random Access Memory
RLH	Random Latin Hypercube
rms	Root Mean Square
SLR	Simple Linear Regression
S-NOLH	Saturated Nearly Orthogonal Latin Hypercube
SOP	Standard Operating Procedure

THIS PAGE INTENTIONALLY LEFT BLANK

## EXECUTIVE SUMMARY

Experiments using computer models are of great importance to scientific research, national defense, and public policy debates. This follows from continual improvements in computational power against a backdrop of rising costs and other challenges frequently associated with physical experimentation. Moreover, often it is not practical to conduct many, or even any, physical experiments—e.g., the long-term effects of various policies on global climate, emergency response to large-scale nuclear accidents, potential major military conflicts, etc. In situations with a dearth of real-world experimental data, computer models are frequently used to help understand these complex issues.

Computer experimentation in the above areas may contain thousands of input variables and/or take a long time (e.g., many days) to run (Kleijnen et al., 2005). Researchers have several techniques to effectively extract information from these models. Among them are designs of experiments that are specifically developed for efficiently exploring high-dimensional computer models.

Latin hypercube (LH) sampling (McKay et al., 1979) has proven to be an invaluable design technique. In fact, LHs are reported to be the predominant design for experiments involving computer simulations (Buyske & Trout, 2001). A key reason for this is that they come with minimal restrictions on the number of factors (i.e., input variables) and sampling budget. In addition, the resultant output data allow us to fit many different models to multiple outputs from a single experimental set. These designs permit us to simultaneously screen many factors for significance and to fit very complex meta-models (including nonparametric) to a handful of dominant variables. This flexibility extends to visual investigations of the data (Sanchez & Lucas, 2002), as we get many viewpoints from which to observe the relationships between inputs and outputs.

There are a large number of LHs for any given *experimental condition*, by which we mean the number of design points ( $n$ ) and variables ( $k$ ). A *design point* is a unique combination of the values of the input variables. Some LH designs possess better properties than others. For example, LHs can have unacceptable correlations among the input factors, thus hindering many statistical procedures that we might wish to apply in

analyzing the relationships between inputs and outputs. To mitigate this problem, various researchers (e.g., Owen, 1994; Ye, 1998; Cioppa, 2002; Ang, 2006; and Steinberg & Lin, 2006) have developed algorithms to eliminate or significantly reduce correlations among input variables—thereby developing orthogonal and nearly orthogonal Latin hypercubes (OLH, NOLH). A design is *nearly orthogonal* if the maximum absolute pairwise correlation ( $\rho_{map}$ ) among the columns of the design matrix is  $\leq .05$ .

The success of these recent efforts to create OLH and NOLH designs is extensive, but all of these approaches are subject to stringent constraints in their dimensionality, usually requiring that  $n$  be a power of two or a power of two plus one. Steinberg and Lin (2006) state, “[t]he primary limitation to our method is the severe sample size constraint.” For instance, to explore 12 factors Steinberg and Lin require only 16 runs. To explore 13 factors, they require a design with 64 runs.

This dissertation details an algorithm for generating NOLHs for a greatly expanded set of  $n$  and  $k$ , including situations where  $k = n - 1$ , i.e., fully saturated designs, as well as methodologies that expand on the usage of LHs to include accommodating discrete variables into nearly orthogonal designs. We further specify the conditions under which these approaches are appropriate. Our research goals are listed below.

- Provide analysts the ability to apply uncorrected LH designs to a number of situations in which scientists do not have the advantage of developing OLH or NOLH designs, or may not wish to use OLH or NOLH designs.
- Provide analysts the ability to generate OLH and NOLH designs for many more combinations of  $n$  and  $k$  than previous approaches allowed—with a particular emphasis on maximizing  $k$  for any given  $n$ .
- Provide analysts the ability to incorporate different factor types (discrete and continuous) with different value levels (mixed factor, mixed level) into a nearly orthogonal design matrix.
- Provide analysts the ability to quickly generate new, robust, special purpose designs to meet new and changing design requirements.

To meet these goals, we quantify when random Latin hypercubes (RLHs) are acceptable and develop a number of progressive techniques to construct and use NOLHs.

Tools and methods, based on the maximum absolute pairwise correlation, help select an appropriate dimension for the experimental design and obtain an uncorrected

RLH with a suitable  $\rho_{map}$ . Astute selection of a design dimension ( $n$  = sample runs,  $k$  = factors) will often allow an experimenter to quickly generate a useful design. We present a procedure that systematically directs the selection of an appropriate design dimension and provides guidance for accepting an uncorrected RLH, as RLH designs are commonly used by practitioners. This includes a parsimonious function that relates the expected maximum absolute pairwise correlation to transformed values of  $n$  and  $k$ . This method requires no specialized algorithms or software packages, making it applicable in most situations and an attractive option for experimenters with limited resources.

We expand the application of Florian's method (1992) to produce new OLH and NOLH designs. Our studies reveal that iterative application of Florian's method can be extremely powerful. We demonstrate that the number of candidate designs needed is often much less than 1,000 for even large experiments. When  $n \geq 50$  and  $k \leq \frac{n}{3}$ , Florian's method is usually sufficient to transform one randomly generated LH to meet our NOLH criteria. Even if we break these loose constraints, we often construct NOLHs.

We use optimization techniques to construct OLH and NOLH designs for dimensions beyond what Florian's method allows. Optimization presents difficulties for solving the experimental design problem because our objective function is nonlinear and we have integer constraints. As  $n$  and  $k$  increase, the dimensionality of LHs adds to the challenges. Therefore, we develop a construction methodology based on a focused optimization routine that greatly expands the set of  $n$  and  $k$  for which orthogonal and nearly orthogonal designs are available. Specifically, we combine RLH generation, Florian's correlation reduction method, and optimization of a mixed integer problem—using a heuristic that focuses on one column at a time—to minimize  $\rho_{map}$  for a specified  $n$  and  $k$ . By minimizing the worst-case correlation, we control all other pairwise correlations. To best illustrate the power of this new methodology, we concentrate on design dimensions that are absent from existing catalogues and ones that are the most difficult to generate with previous methods; that is, experiments with  $n \leq 50$  as  $k$  approaches  $n$ .



Our results show significant improvements over the efficiency of past experimental designs for the same number of runs. Table 1 compares our new technique with other recent methods to create orthogonal or nearly orthogonal Latin hypercube designs. For a given  $n$ , our methods produce designs that can explore many more factors.

Table 1. A comparison of recent construction methods to generate OLH or NOLH designs shows that our new method significantly increases the number of factors,  $k$ , that may be examined for the same number of runs,  $n$ .

$n$	Maximum $k$ for Each Method				
	Ye	Cioppa	Ang	Steinberg and Lin*	New
17	6	7	8	12	16
33	8	11	16	NA	32
65	10	16	32	56	63**

\* Steinberg and Lin's method uses  $n - 1$  runs.

\*\* New design uses 64 runs.

In addition to increasing the number of NOLHs, we exploit the flexibility of our new designs to develop an approach that handles mixed-factor, mixed-level (MFML) experiments. In accordance with LH sampling construct (McKay et al., 1979), OLH and NOLH designs assume that all variables are continuous. In practice, this assumption is frequently false. Many experiments include a number of discrete variables (that do not necessarily have the same number of value levels), along with continuous factors. We designate these as MFML experiments.

MFML experiments can diminish the advantages of OLH and NOLH designs. Converting (by rounding) the actual values of these discrete variables onto a raw OLH or NOLH design, especially when there are a small number of runs, often results in an overall design with high correlations among input variables. To remedy the problems that MFML experiments present scientists, we exploit the dimensional flexibility of our new designs by combining them with stacking methods. We also combine our method with proven design techniques. The resulting MFML designs retain much of the orthogonality properties of the basic NOLH, thereby maintaining their utility.

The principal shortfall of previous methods to develop OLH and NOLH designs is their strict limitations in dimensionality. Our new techniques and procedures overcome

many of these restrictions, thereby enabling scientists and analysts to apply these powerful designs in more situations.

THIS PAGE INTENTIONALLY LEFT BLANK

## **ACKNOWLEDGEMENTS**

This dissertation would not have been possible without the guidance and encouragement of Dr. Tom Lucas. His abundant patience and fountain of knowledge were key in shaping and directing my efforts. The teamwork that he forged among committee members and his alignment of my endeavors with them were critical. I am truly appreciative of the committee's consideration and dedication. I recognize all the students who trusted our new designs. Their successes provide validity and great satisfaction to the value of this work. It was the great understanding and indulgence of Dean Peter Purdue that enabled me to keep balance between my academic aspirations and organizational duties. I dedicate this work to my wife, Gerri, and children: Greg, Tara, Ronnie, Andrea, and Cara. Their love and unwavering support give meaning to my daily toil. They are central to all that I ever achieve in life.

THIS PAGE INTENTIONALLY LEFT BLANK

# I. INTRODUCTION

Latin hypercubes (LHs) have proven to be a powerful tool for conducting high-dimensional computational experiments in many of the sciences and national security studies. Our research expands the set of orthogonal Latin hypercube (OLH) and nearly orthogonal Latin hypercube (NOLH) designs available to those involved in experimentation. We accomplish this by greatly increasing the feasible combinations of design points<sup>1</sup> ( $n$ ) and factors<sup>2</sup> ( $k$ ) that are available for experimentation—which heretofore have been highly constrained. Furthermore, we offer methods to create nearly orthogonal designs when some of the input variables are discrete—perhaps with a large number of possible values. Moreover, our work provides a descriptive study of the correlation structures inherent in random and specially constructed LHs. The culmination of this dissertation provides the first high-dimensional, fully saturated NOLHs.

## A. CHALLENGES IN EXPERIMENTAL DESIGNS FOR SIMULATIONS

Experimentation is fundamental to knowledge acquisition. Unfortunately, physical experimentation is often infeasible due to safety, money, time, or resource constraints. Consequently, experimenters often turn to computational experimentation, such as computer simulation, of the system of interest. The ability of computers to simulate increasingly complex problems provides analysts greater potential to assist decision makers, such as those in the Department of Defense (DoD). Network-centric warfare and irregular warfare are but two of the areas that involve vast numbers of quantitative and qualitative variables. Often these simulations contain hundreds, or even many thousands, of input variables (Saeger & Hinch, 2001). The commercial world and natural scientists face similar dilemmas. Studies in human behavior and biomimetics often use computer simulations and rely on efficient designs of experiments (DOEs),

---

<sup>1</sup> A design point is a unique combination of the values of the factors. We use the terms *design point* and *run* interchangeably.

<sup>2</sup> Factors are input variables whose settings or values can be controlled by the experimenter. We use the terms *factor* and *variable* interchangeably.

adopting any technique that allows analysts to “systematically examine a broader range of possible innovations . . .” (Booker, 2005).

Researchers implement a number of methodologies to extract the most useful information from simulations. Among these methods are experimental designs specifically developed for efficiently exploring computer models (Kleijnen et al., 2005). The design specifies the inputs for the experiments. In particular, given that  $n$  experiments are to be conducted over  $k$  input variables, the DOE is usually specified by an  $n \times k$  design matrix  $X$ . Each column of  $X$  represents an input variable and each row specifies the input variable values for a single design points (see Table 2).

Table 2. An example design matrix for three factors (i.e.,  $k = 3$ ) and eight experiments (i.e.,  $n = 8$ ).

<b>Design Point</b>	<b><math>Factor_1</math></b>	<b><math>Factor_2</math></b>	<b><math>Factor_3</math></b>
1	1	0.5	0
2	1	0.5	10
3	1	1.5	0
4	1	1.5	10
5	0	0.5	0
6	0	0.5	10
7	0	1.5	0
8	0	1.5	10

The information that is obtainable by analyzing the data after conducting the experiments depends critically on the design. For example, if a quantitative input variable only has two distinct values, then the expected response cannot be modeled as a quadratic in that variable. If we know in advance what metamodels we want to fit to describe the response surface of the simulation model (Law & Kelton, 2003), and the error structure of the experiments, then an optimal design may exist (Fedorov, 1972). However, in many cases, especially when conducting exploratory analysis, we desire designs that “allow one to fit a variety of models” (Santner et al., 2003). For such situations, LH sampling (McKay et al., 1979) has proven to be an invaluable technique for designing high-dimensional computer experiments. For the past 15 years it has become an “important part of uncertainty analyses” (Wyss & Jorgensen, 1998). Under

general conditions, LH designs perform exceptionally well in comparison to other popular experimental design options (Johnson, 2008). In fact, LHs are purported to be the predominant DOEs involving computer simulations (Buyske & Trout, 2001).

LHs are popular designs for simulation exploration because of their design and analytical flexibility (Lucas & Sanchez, 2006a). An LH design can easily be constructed for any number of continuous input variables ( $k$ ) and sampling budget ( $n$ ) as long as  $k \leq n$ . Indeed, many simulation software packages, even spreadsheet simulation add-ons, can generate LHs (Sugiyama & Chow, 1997). With sufficiently large  $n$ , the output data can be fit to a wide variety of metamodels (from simple to complex) for many different responses. In particular, LHs enable us to simultaneously screen many factors for significance and fit very complex metamodels to a modest number of critical variables. Moreover, LHs have good space-filling properties, i.e., they are good at providing “information about all portions of the experimental region” (Santner et al., 2003).

The existence of desirable properties in LH designs, which correspond with their degree of utility, is dependent on the absence of correlation among the columns of the design matrix. Many analytical techniques that experimenters apply to computer outputs—such as regression modeling and partition trees—suffer when there is multicollinearity among the input variables (Montgomery et al., 2001 and Kim & Loh, 2003). Consequently, analysts usually desire a design matrix with a correlation structure that is close to that of the identity matrix (Iman & Conover, 1982, and Iman & Davenport, 1982). Unfortunately, generating LH designs involves an inherent randomness in the construction of the columns—a random permutation of the value levels, 1 through  $n$ . This construction method invites the possibility of substantial multicollinearity among the columns of the design matrix, especially when  $n$  is not substantially larger than  $k$ .

To mitigate the difficulties that can be caused by correlations among the columns in the design matrix, a succession of methods have been developed that reduce or even eliminate the correlations. An LH is called an *orthogonal Latin hypercube* (OLH) if the correlation coefficient between all pairs of columns in the design matrix is zero. We will denote an OLH with  $k$  variables in  $n$  runs as  $O_k^n$ . In some situations, an OLH may not be



available or it may come with poor space-filling properties (Cioppa & Lucas, 2007). In such a situation, a *nearly orthogonal Latin hypercube* (NOLH) may exist and be preferred. We will define NOLHs precisely later in this section. An NOLH with  $k$  variables in  $n$  runs is denoted as  $N_k^n$ . The absence or near absence of correlations among their design columns makes OLH and NOLH designs attractive. However, there are only a small number of these catalogued designs available and most applications do not use an OLH or NOLH design.

A variety of reasons exist for why OLH and NOLH designs are not used more often than they currently are. Efforts to reduce or eliminate correlation are often computationally expensive and time consuming (Cioppa, 2002). The complexity of algorithms involved in creating them typically requires specialized software and programming (Iman & Conover, 1982). Most detrimental for increasing the use of OLH and NOLH designs are the severe restrictions on the dimensionality of the design matrix (and hence, experiment). In every case, the dimensions of new designs restrict  $n$  to be a function of a power of 2, or a power of 2 plus 1. Furthermore, methods for creating LH designs assume continuous variables, which necessitate rounding off the raw design values to accommodate discrete variables. The resulting design values do not typically retain the orthogonality (or near orthogonality) properties of the raw design.

This dissertation overcomes the above obstacles and greatly expands the set of readily available OLH and NOLH designs. It presents new construction methods, and augments previous techniques, to generate design matrices ( $X$ ) with little or no correlation. While LH designs assume continuous variables, our work also encompasses discrete variables that may have unequal numbers of levels. The resulting design matrices fill substantial gaps in the current library of OLH and NOLH designs.

## **B. RANDOM LATIN HYPERCUBES AND INHERENT CORRELATION**

The foundation of our work resides in the characteristics of the *random Latin hypercube* (RLH). We briefly describe the nature of the RLH as a reference for defining the terminology we use in this dissertation and setting the context for our enhancements.

## 1. Random Latin Hypercube Generation

RLH generation is so named to emphasize the randomness in its construction. Generating an RLH is relatively simple. Consider an  $n$  by  $k$  matrix,  $\mathbf{X}$ . Owen (1994) describes the  $i^{\text{th}}$  element in the  $j^{\text{th}}$  column,  $X_i^j$ , of an LH as

$$X_i^j = F_j^{-1} \left( \frac{(\pi_j(i) - U_{ij})}{n} \right), \text{ for } i = 1, \dots, n \text{ and } j = 1, \dots, k, \text{ where } \pi_j(1), \dots, \pi_j(n) \text{ is one of the}$$

$n!$  possible random permutations of  $1, \dots, n$  in which all  $n!$  permutations are equally likely. The  $F_j, j = 1, \dots, k$  are continuous and invertible distribution functions. Finally, the  $U_{ij}, i = 1, \dots, n, j = 1, \dots, k$  are independent and identically distributed uniform  $[0, 1]$  random variables.

Similar to Owen (1994), Ye (1998), Cioppa (2002), and Steinberg and Lin (2006), we use a lattice version of the LH (Patterson, 1954) and assume that  $F_j$  is the distribution function of a uniform  $[0, 1]$  for  $i = 1, \dots, n, j = 1, \dots, k$ . We further replace the uniform random variable,  $U_{ij}$ , with 0.5, the median of a uniform  $[0, 1]$  distribution; thus creating a center point in each stratum. Therefore, generating an RLH reduces to independently generating  $k$  columns, each a permutation of the integers  $i = 1, \dots, n$ , where the  $n!$  permutations are equally likely. For a given column,  $j = 1, 2, \dots, k$ , this yields:

$$X_i^j = \frac{(\pi_j(i) - .5)}{n}, \text{ for } i = 1, \dots, n. \text{ The columns of the resulting design matrix, } X^j, \text{ for}$$

$j = 1, \dots, k$ , are  $k$  independent permutations of the vector  $\underline{x}$ , which contains elements derived from the following simplified formula (Owen, 1994):  $x_i = \frac{(i - .5)}{n}$ , for  $i = 1, \dots, n$ .

We illustrate this method by examining three factors ( $k = 3$ ) with four sample runs ( $n = 4$ ). The experimenter calculates the first element,  $i = 1$ , from the simplified formula  $\frac{i - .5}{n} = \frac{1 - .5}{4} = \frac{1/2}{4} = \frac{1}{8}$ . Computation of the remaining elements, for  $i = 2, 3$ , and 4,

results in the vector  $\underline{x} = \left( \frac{1}{8}, \frac{3}{8}, \frac{5}{8}, \frac{7}{8} \right)$ . A random permutation of  $\underline{x}$  for each column

produces a lattice LH for three factors and four design points, as shown in Figure 1.

$$X = \begin{pmatrix} \frac{3}{8} & \frac{1}{8} & \frac{5}{8} \\ \frac{5}{8} & \frac{3}{8} & \frac{7}{8} \\ \frac{7}{8} & \frac{5}{8} & \frac{1}{8} \\ \frac{1}{8} & \frac{7}{8} & \frac{3}{8} \end{pmatrix}.$$

Figure 1. Three permutations of the vector  $\underline{x}$  results in an RLH design for three variables and four sample design points using Owens' simplified formula to create a lattice LH.

The columns in  $X$  are often translated and scaled for a particular application. For a given column, the values are equally spaced between the minimum and maximum values allowable for the corresponding variable. Given this lattice construction, a total of  $(n!)$  permutations are possible for each column. This does not create  $(n!)^k$  unique LHs because some designs can be mapped to others by permuting rows and columns. However, the total number of unique designs is astronomically large. For example, there are over 87 billion unique designs with  $k=2$  and  $n=14$ .

The simplicity and flexibility of this sampling scheme has a price. It is precisely the randomness of column generation that opens the possibility for strong multicollinearity among the columns of the design matrix.

## 2. Measure of Nonorthogonality for Design Selection

Along with Tang (1998), we recognize that selecting a design based on the correlations between the columns of the design matrix is a reasonable way to obtain a design with acceptable nonorthogonality. Cioppa (2002) specifically uses the maximum absolute pairwise correlation between columns of an LH to obtain OLH and NOLH designs.

Following Cioppa, we designate the maximum absolute pairwise correlation ( $\rho_{map}$ ) as the key measure for discriminating between designs. There are  $\binom{k}{2}$  pairwise correlations in a design with  $k$  variables. Computation of the correlation coefficient

between any two vector columns in design matrix  $\mathbf{X}$  is straightforward. Using Pearson's equation, the correlation between  $X^i$  and  $X^j$  is given by

$$\rho_{ij} = \frac{\sum_{b=1}^n [(x_b^i - \bar{x}^i)(x_b^j - \bar{x}^j)]}{\sqrt{\sum_{b=1}^n (x_b^i - \bar{x}^i)^2 \sum_{b=1}^n (x_b^j - \bar{x}^j)^2}},$$

where  $\bar{x}^i$  and  $\bar{x}^j$  are the mean values of the  $i^{\text{th}}$  and  $j^{\text{th}}$  columns and  $x_b^i$  is the  $b^{\text{th}}$  value of the  $i^{\text{th}}$  column.

The largest absolute correlation among the columns gives the most extreme pairwise correlation. We use this value to define the degree of nonorthogonality of a design. By minimizing the worst-case pairwise correlation, we control all other pairwise correlations. Hence, we focus attention on  $\rho_{map}$  throughout our study and creation of new OLH and NOLH designs. Specifically,

$$\rho_{map} = \max_{i \neq j} \left\{ \left| \rho_{ij} \right| \right\}.$$

Our collection of techniques minimizes  $\rho_{map}$ . We further use  $\rho_{map}$  as a guide for combining reliable design approaches with our new designs, thereby expanding the utility of OLH and NOLH.

The role of  $\rho_{map}$  in our study is evident in the large values in which it appears in RLHs, especially when  $k$  is not too much smaller than  $n$ . Consider the design matrix in Figure 2. Three new permutations of vector  $\underline{x}$  result in this design matrix, where columns 2 and 3 have a correlation value of  $-1$ , i.e.,  $\rho_{map} = 1$ .

$$\begin{pmatrix} \frac{3}{8} & \frac{1}{8} & \frac{7}{8} \\ \frac{5}{8} & \frac{3}{8} & \frac{5}{8} \\ \frac{7}{8} & \frac{5}{8} & \frac{3}{8} \\ \frac{1}{8} & \frac{7}{8} & \frac{1}{8} \end{pmatrix}.$$

Figure 2. In this RLH, columns two and three give  $|\rho_{23}| = \rho_{map} = 1$ .

In a sampling method in which all possible RLH designs are equally probable (Satterthwaite, 1959), the probability that a highly correlated design occurs is high. To illustrate, we randomly generate 1,000  $4 \times 3$  RLH designs and measure the  $\rho_{map}$  of each. Over 77% of the designs have a  $\rho_{map} \geq 0.80$ , with nearly 25% having  $\rho_{map} = 1$ . Clearly, the possibility of incurring high correlations in an RLH can be problematic.

Whereas randomness creates a concern of possible high correlations among the columns in the design matrix, it may also result in RLH designs that have very small correlations. This observation offers an opportunity to develop a systematic process for selecting suitable RLH designs and applying a variety of transformation and generation techniques to minimize their nonorthogonality. In doing so, possibilities to reduce restrictions in dimensionality also emerge.

### C. A PROGRESSION OF OLH AND NOLH CONSTRUCTION METHODS

Techniques to create new OLH and NOLH designs have produced incremental increases in feasible  $n$  and  $k$  combinations. A succession of studies extends the foundational paper by McKay et al. (1979) to produce methods that can reduce the correlations of a given LH design. Iman and Conover (1982) induce rank correlation in the design to correspond with the correlation that one would expect from the input variables, resulting in a means to control the correlation of the design matrix. Similar to Iman and Conover (1982), Florian (1992) updates LH sampling designs through matrix transformation. Owen (1994) introduces a process based on Gram-Schmidt orthogonalization to control correlations among variables. These methods attempt to convert LH designs to correspond with correlation matrices that approach the identity matrix. They do not guarantee the complete elimination of correlation from the design, although uncorrelated designs are achieved in some cases.

A relatively recent wave of techniques completely eliminates correlation among the columns of an LH design during its construction. Ye (1998) introduces a class of LHs with no correlation among its columns and designated them OLHs. Ye finds that it is possible to develop OLH designs when the number of runs,  $n$ , is a power of 2 plus 1. Through Ye's method, experimenters can generate an orthogonal design ( $\rho_{map} = 0$  and

condition number = 1)<sup>3</sup> with  $k = 2m - 2$  for an experiment with  $n = 2^m + 1$  runs, where  $m$  is a positive integer. Each column in the design uses a permutation matrix that is based on the Kronecker product (Graham, 1981) of identity matrices and a transposition matrix. Ye uses all  $m$  of his main permutation matrices and  $m - 2$  permutation matrices from the  $\binom{m-1}{2}$  possible pairwise multiplications of the main permutation matrices. Ye (1998) understands that the sizes of his OLH designs are somewhat inflexible, but asserts that computer power mitigates this disadvantage.

Cioppa (2002) extends Ye's method for constructing OLH designs by increasing the number of permutation matrices that generate mutually orthogonal columns. Instead of using  $m - 2$  of the pairwise combinations, Cioppa uses all  $\binom{m-1}{2}$  pairwise matrix multiplication combinations. In so doing, Cioppa increases the number of columns ( $k$ ) in the design matrix from  $2m - 2$  to  $m + \binom{m-1}{2}$ , while maintaining their full orthogonality for the same number of runs.

Cioppa, however, notices that this approach can yield an OLH that does not have good space-filling properties. Hence, Cioppa develops a computationally intensive algorithm that improves on the space-filling properties of Ye's designs by accepting a small amount of nonorthogonality. Cioppa selectively catalogues designs with space-filling qualities that are better than Ye's designs, in terms of the modified  $L_2$  discrepancy ( $ML_2$ ) and Euclidean maximin (Mm) measures. The amount of nonorthogonality that Cioppa accepts in terms of  $\rho_{map}$  is at most 0.03 and  $\mathbf{X}$  must also have a condition number less than 1.13 (when the columns of  $\mathbf{X}$  are scaled to span the interval  $[-1,1]$ ). We recall that an OLH has  $\rho_{map}$  equal to 0 and condition number equal to 1. The slight nonorthogonality of the design earns its name as an NOLH. Cioppa demonstrates that his method works for up to  $k = 67$  and conjectures that for any positive

---

<sup>3</sup> The condition number is the ratio of the largest to smallest eigenvalues of  $\mathbf{X}'\mathbf{X}$ , where  $\mathbf{X}$  is the  $n \times k$  design matrix (Cioppa & Lucas, 2007). The condition number represents the design matrix's sensitivity to small changes in its values (Leon, 2002).

integer value of  $m$ , an NOLH design can be constructed whenever  $n = 2^m + 1$  and  $k = m + \binom{m-1}{2}$ . Existing space-filling NOLH designs from Cioppa's method are catalogued for up to 29 factors within 257 runs.

Ang (2006) augments Cioppa's methodologies to create larger OLH and NOLH dimensions. While Cioppa uses all  $\binom{m-1}{2}$  *pairwise* matrix multiplication combinations, Ang uses all three-way, and larger, product combinations to create new permutation matrices. Each new permutation matrix generates a new column in the OLH design. The number of factors that Ang's designs can explore is  $k = 1 + \sum_{j=1}^p \binom{m-1}{j}$ , where  $p$  (with the maximum  $p = m - 1$ ) is the  $p$ -way combinations used for developing the permutation matrices and  $m$  is a positive integer. The number of runs required to explore  $k$  factors is still  $n = 2^m + 1$ . For instance, a design with  $m = 5$  can explore  $k = 16$  factors when using the maximum ( $p = 4$ ) combinations. Such a design requires  $n = 2^5 + 1 = 33$  runs. This technique results in designs that can explore more factors with the same number of runs. These designs begin to approach a certain saturation level, with a noted decrease in their space-filling properties as  $p$  increases. Additionally, Ang (2006) redefines the NOLH criteria as  $\rho_{map} \leq 0.05$  and a condition number of no more than 1.20. We use these thresholds for our new methodologies, with a particular emphasis on minimizing  $\rho_{map}$ .

Steinberg and Lin (2006) rotate two-level factorial designs to construct OLHs for  $n = 2^h$ , with  $h$  a power of 2, and the maximum number of factors being  $B_h \times h$ , where  $B_h = \left\lfloor \frac{n-1}{h} \right\rfloor$ , with  $\lfloor c \rfloor =$  the maximum integer less than or equal to  $c$ . For instance, for  $n = 16$  runs,  $h = 4$  and  $B_h = 3$ , so a  $O_{12}^{16}$  is possible. We note that these OLHs do not include a center point, as do previously discussed OLHs.

The success that these recent efforts have had in creating OLH and NOLH designs is extensive, but OLH and NOLH designs are still subject to stringent constraints in their dimensionality. As Steinberg and Lin (2006) state, "[t]he primary limitation to

our method is the severe sample size constraint.” We see that to explore 12 factors with their algorithm requires only 16 runs. To explore 13 factors, Steinberg and Lin require a design with 64 runs. In comparison, Ang (2006) needs 33 runs to explore up to 16 factors. However, to explore 17 factors, Ang must increase the number of runs to 65. Table 3 compares our new method with recent methods to create orthogonal or nearly orthogonal designs.

Table 3. A comparison of recent construction methods to generate OLH or NOLH designs shows that our new method significantly increases the number of factors,  $k$ , that may be examined for the same number of runs,  $n$ .

$n$	Maximum $k$ for Each Method				
	Ye	Cioppa	Ang	Steinberg and Lin*	New
17	6	7	8	12	16
33	8	11	16	None	32
65	10	16	32	56	63**

\* Steinberg and Lin’s method uses  $n - 1$  runs.

\*\* New design uses 64 runs.

All existing OLH techniques have a common inflexibility in dimensionality. Our new approach constructs flexible NOLH designs—that is, sample sizes need not be related to a power of 2 (any  $n$  is allowable) and many feasible  $k$  (often all  $k = 1, \dots, n - 1$ ) are possible. Indeed, in many cases, we create fully saturated designs that are orthogonal or nearly orthogonal. Furthermore, the flexibility of our OLH and NOLH designs make it possible to combine them with other methods to create experimental designs that include discrete variables.

#### D. DISSERTATION ORGANIZATION

The flow of this dissertation is a progression of techniques and procedures that we build to meet our research goals. The resulting tools and products greatly expand our ability to explore broad regions of a complex simulation model containing a high-dimensional input space, characterized by a response surface that may be highly nonlinear. We introduce several methodologies that enable analysts to use OLHs and NOLHs in more situations. Furthermore, we specify the conditions under which the



various approaches are appropriate. We also develop algorithms for accommodating discrete variables into nearly orthogonal designs. Our research goals are listed below.

- Provide analysts the ability to apply uncorrected LH designs to a number of situations in which scientists do not have the advantage of developing OLH or NOLH designs, or may not wish to use OLH or NOLH designs.
- Provide analysts the ability to generate OLH and NOLH designs for many more combinations of  $n$  and  $k$  than previous approaches allowed—with a particular emphasis on maximizing  $k$  for any given  $n$ .
- Provide analysts the ability to incorporate different factor types (discrete and continuous) with different value levels (mixed factor, mixed level) into a nearly orthogonal design matrix.
- Provide analysts the ability to quickly generate new, robust, special purpose designs to meet new and changing design requirements.

The following four chapters each describe techniques to reach our study goals. Chapter II discusses RLHs more thoroughly, to provide the reader with a better understanding of their utility and to build a foundation for other techniques. It contains the mathematical theory underlying our designs and the details necessary to construct them. We discuss  $\rho_{map}$  as the nonorthogonality measure that is our focal point for all design techniques. We create tools based on  $\rho_{map}$  to select dimensions for an RLH design that has a very good chance of possessing an acceptable nonorthogonality measure. Chapter III quantifies the effectiveness of Florian's (1992) correlation reduction method, as well as its limitations, for producing OLH and NOLH designs. Florian's method is an earlier technique that has great utility in constructing a number of nearly orthogonal designs that do not conform to previous restrictions in their dimensions. Chapter IV contains a combined application of RLH generation, Florian's (1992) method, and mixed integer programming (MIP) to produce new OLH and NOLH designs that greatly expand on what was previously available. The difficulty in using optimization to solve combinatorial problems, such as DOEs, prevents scientists from pursuing it as a practical solution method. We formulate the MIP problem to make optimization a viable option. Application of these combined techniques results in a new family of designs that we call saturated nearly orthogonal Latin hypercubes (S-NOLH). We discuss in detail the advantages that this family of designs offers to scientists.

Chapter V considers *mixed-factor, mixed-level* experiments that include (perhaps many) discrete variables that do not necessarily have the same number of possible values. The mixture of these variables utilizes a systematic process for leveraging OLH and NOLH designs constructed using other techniques. We apply crossed designs, column permutation, stacking methods, and logical lines of processes to produce designs that retain much of the orthogonal properties of the *base design*. The base design is the set of continuous variables from which we create an OLH or NOLH design, as articulated from techniques in earlier chapters.

Chapter VI demonstrates a case study for our designs. We survey a number of studies, over a variety of applications, which use our new designs. Finally, Chapter VII summarizes our conclusions in this research area and provides directions for future research.

THIS PAGE INTENTIONALLY LEFT BLANK

## II. SHAPING CONDITIONS FOR APPLYING UNCORRECTED RANDOM LATIN HYPERCUBES

Latin hypercubes (LHs) are one of the most commonly used designs for computer experiments (Kleijnen et al., 2005). However, the regularity in which high correlation occurs in some of these designs complicates analysis of the results. Efforts to reduce or eliminate correlation are often computationally and monetarily expensive, require special purpose software, and can be time consuming. Many scientists frequently use uncorrected<sup>4</sup> random LH (RLH) designs in their experiments and attempt to work with the inefficiencies that may result from them.

We develop tools and techniques, based on the maximum absolute pairwise correlation, to select an appropriate dimension for the experimental design and obtain an uncorrected RLH with acceptable correlations. This method requires no specialized algorithms or software packages, making it applicable in most situations. The speed with which our method can be employed makes it an attractive option for experimenters with limited resources.

We propose an approach to find designs with acceptable correlations among their columns: *dimension selection and RLH generation*. Current methods to minimize correlation in LH designs generally fall into two broad categories: (1) those that transform an original design to have an acceptable degree of correlation, or (2) those that eliminate correlation during construction of the design columns. Our technique falls into neither category. We accept the natural proclivity of RLH production to frequently have high correlation among the columns and use the randomness in column generation to our advantage.

We assert that astute selection of a design dimension ( $n$  = sample runs,  $k$  = factors) and a predictive model for an RLH's expected maximum absolute pairwise correlation will often allow an experimenter to quickly generate an acceptable design. Our method takes advantage of the ease with which an RLH design is created. Coupling rapid RLH generation, as we described in Chapter I, with the degree to which we can

---

<sup>4</sup> An uncorrected LH is one to which no correlation reduction method is applied.

predict the maximum absolute pairwise correlation, our procedure systematically directs the selection of an appropriate design dimension and provides guidance for accepting an uncorrected RLH. Because this method requires no manipulation of the RLH, any investigator can employ it with common software and computing tools.

## A. TOOLS FOR DIMENSION SELECTION AND RANDOM LATIN HYPERCUBE GENERATION

We earlier established  $\rho_{map}$  as our nonorthogonality measure for an LH design. We use this measure to select an RLH design with an acceptable degree of nonorthogonality. First, we use  $\rho_{map}$  as a guide to select an appropriate design dimension for an experiment. Second, with the design dimension fixed, the nonorthogonality measure serves as a threshold for selecting an acceptable design from iterative generations of RLH. This section describes the development of correlation-based tools and the procedures for their application.

We use these  $\rho_{map}$ -based tools when more complex algorithms associated with correlation reduction methods are not readily available to the experimenter. The ease of generating an LH with comparatively unsophisticated means makes them an attractive option (Sugiyama & Chow, 1997) for creating experimental designs. If a random generation of an LH has acceptable correlation among its columns, an experimenter can reap the benefits from an efficient design with little computational effort. Cioppa (2002) defines efficient experimental designs as those which (i) detect as many significant variables, nonlinear effects, interactions, and their associated ranges as possible, (ii) declare significant as few non-significant variables and interactions as possible, and (iii) accomplishes (i) and (ii) with a minimal number of runs. This chapter introduces the mechanism to create experimental designs that possess acceptable levels of nonorthogonality from uncorrected RLHs.

### 1. Creating the $\overline{\rho_{map}^{\min}}$ Table

We first create a series of correlation tables that combine design dimensions, which correspond to known OLH and NOLH designs (Ye, 1998 and Cioppa, 2002), and

the number of RLHs to generate from which to select a design. Each entry in the correlation table is an average of the minimum  $\rho_{map}$  for a given  $n$  and  $k$ .

We begin our work with an initial set of data that consists of 42  $n$  by  $k$  design combinations. Using Cioppa's (2002) convention for design dimensions, we explore combinations of  $n = 2^m + 1$  for  $m = 1, \dots, 8$  and  $k = m + \binom{m-1}{2}$  for  $m = 1, \dots, 16$ .

Therefore, we examine design dimensions as small as  $\{n = 17, k = 7\}$  and as large as  $\{n = 257, k = 121\}$ . We consider only those designs with  $n > k$ . Designs where the number of runs is less than the number of factors produce unstable models (Leon, 2002). We enter NA (Not Applicable) for  $n \leq k$  combinations in the tables.

The correlation tables also account for the number of RLHs to generate, and from which to select an acceptable design. We let  $G$  be the number of designs from which to select our experimental plan. Tang (1998) uses a similar procedure, comparing several designs before selecting one to apply correlation reduction methods. Other scientists consider many more designs before choosing one for an experimental design. To create the correlation table we generate 200 (annotated as G200) uncorrected RLH designs for a specific  $n, k$  combination and compute  $\rho_{map}$  for each. From this set of generated designs we record the smallest  $\rho_{map}$ , which we designate as  $\rho_{map}^{\min}$ . We repeat this process 1,000 times for each  $n$  by  $k$  combination and compute the average of the 1,000  $\rho_{map}^{\min}$  values,  $\overline{\rho_{map}^{\min}}$ . This value is the entry in the table for the specific design dimension and estimates the corresponding expected  $\rho_{map}^{\min}$  among 200 uncorrected RLH designs. Examination of these estimates show low variability, as one expects from extreme statistics (David & Nagaraja, 2003), and acceptable precision for our work.

The different design dimensions vary greatly in the values of their  $\overline{\rho_{map}^{\min}}$ , but the largest sample standard deviation for any specific design dimension is 0.025 (Figure 3). Therefore, the standard deviation of  $\overline{\rho_{map}^{\min}}$  for any of these design dimensions will also be very small, considering that the average was a computation from 1,000 independent observations. Figure 3 shows that the largest standard deviation occurs for a small design

$n = 17, k = 16$ , corresponding to  $\overline{\rho_{map}^{\min}} = 0.306$ . The smallest standard deviation is for a large RLH,  $n = 257, k = 106$  corresponding to  $\overline{\rho_{map}^{\min}} = 0.204$ . The majority of these have a standard deviation that is less than 0.01, so predictions will be meaningful.

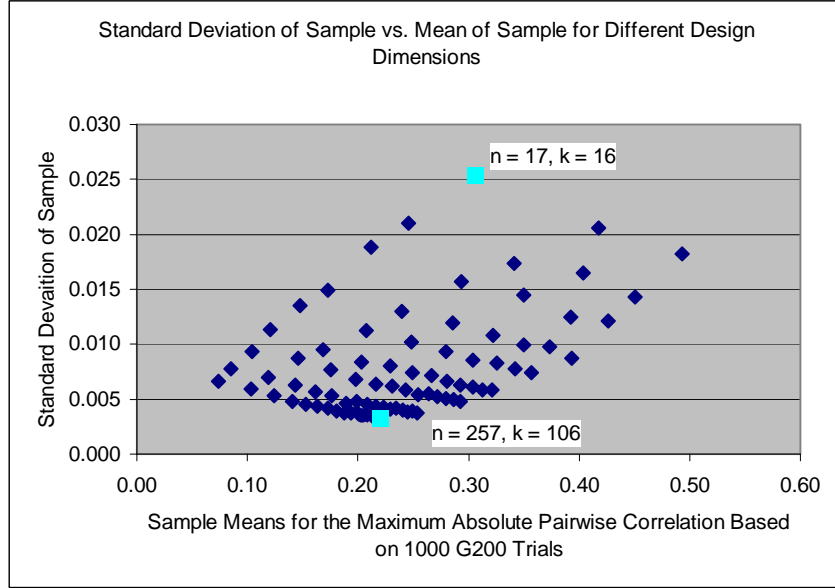


Figure 3. The sample standard deviation of  $\rho_{map}^{\min}$  for 1,000 G200  $n$  and  $k$  combinations is relatively small compared to its  $\overline{\rho_{map}^{\min}}$  value. The largest standard deviation occurs in small designs, while the smallest deviations are for larger designs.

Table 4 is the correlation table for different design combinations for G200. Its organization shows that as the number of runs for a fixed  $k$  increases,  $\overline{\rho_{map}^{\min}}$  decreases, which is consistent with Owen (1994). Conversely,  $\overline{\rho_{map}^{\min}}$  increases as  $k$  increases for a fixed  $n$ . The table provides the design combinations which are likely to produce a design with the required  $\rho_{map}$  by generating 200 RLHs. If the table indicates that the initial design dimension cannot attain the desired correlation within 200 RLHs, then the table guides the experimenter to increase  $n$ , decrease  $k$ , or both, to have a reasonable chance of obtaining an RLH with an acceptable  $\rho_{map}$ . This tool allows the experimenter to ascertain a realistic expectation of  $\rho_{map}$  for a given design dimension.

Table 4. This portion of the G200 table shows the  $\overline{\rho_{map}^{\min}}$  for different design combinations where the number of designs generated in each trial is 200. We enter not applicable (NA) for design combinations where  $n < k$ .

G200	K7	K11	K16	K22	K29	K37	K46	K56	K67	K79	K92	K106	K121
N17	0.3066	0.4188	0.4932	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
N33	0.2108	0.2931	0.3503	0.3923	0.4267	NA	NA	NA	NA	NA	NA	NA	NA
N65	0.1485	0.2073	<b>0.2481</b>	<b>0.2793</b>	0.3038	0.3244	0.3416	0.3567	NA	NA	NA	NA	NA
N129	0.1044	0.1460	<b>0.1756</b>	<b>0.1982</b>	0.2161	0.2310	0.2432	0.2544	0.2638	0.2721	0.2799	0.2869	0.2931
N257	0.0738	0.1032	0.1240	0.1403	0.1530	0.1638	0.1728	0.1805	0.1871	0.1935	0.1988	0.2038	0.2082

Table usage is straightforward. Suppose an analyst wishes to explore 20 factors, while maintaining  $\rho_{map} \leq 0.20$ , but is uncertain to the number of runs to perform. The shaded block of cells in Table 4 indicates that it is possible for the experimenter to obtain an RLH with an acceptable  $\rho_{map}$  within 200 RLHs if the design consists of somewhere between 65 and 129 runs. The table also guides the analyst to consider fewer factors. Additionally, we note that randomly generating an NOLH is unlikely.

One could also increase  $G$ . However, empirical study shows that increasing  $G$ , even to 1,000, gives only marginal improvement from G200. Table 5 gives the standard deviations of G1000 RLHs for design combinations of  $n = 17$  and 257 with  $k = 7$  and 106 and compares them to those for G200 RLHs. The greatest difference in standard deviations in G1000 and G200 designs is for  $n = 17$ ,  $k = 7$ , where G1000 is less than G200 by 0.0039. For the magnitudes of  $\overline{\rho_{map}^{\min}}$  shown in Table 4, 0.0039 is small.

Table 5. We show a few different design combinations to compare the standard deviations of 1,000 G1000 and G200 RLHs. The greatest difference in standard deviations between G1000 and G200 is for the design,  $n = 17$ ,  $k = 7$ , where G1000's standard deviation is smaller by 0.0039.

G1000		
	k7	k106
n17	0.0215	NA
n257	0.0053	0.0028
G200		
	k7	k106
n17	0.0254	NA
n257	0.0066	0.0035



Likewise, the magnitude of  $\overline{\rho_{map}^{\min}}$  for G1000 and G200 in Table 6 shows that the differences are very close to the standard deviations. We conclude that G200 is a reasonable number of RLHs from which to choose the design with the best observed  $\rho_{map}$  since the improvements from a sample size that is five times larger are small.

Table 6. A comparison of  $\overline{\rho_{map}^{\min}}$  for 1000 G1000 and G200 shows that the differences are within the magnitude of the standard deviations shown in Table 5.

<b>G1000</b>		
	<b>k7</b>	<b>k106</b>
<b>n17</b>	0.2737	NA
<b>n257</b>	0.0652	0.1994
<b>G200</b>		
	<b>k7</b>	<b>k106</b>
<b>n17</b>	0.3066	NA
<b>n257</b>	0.0738	0.2038

The general guidance described in this section frames the design dimensions that are necessary to fulfill the analyst's need. However, this approach only provides a rough estimate of the design dimensions. We develop another tool to specify  $n$  and  $k$ .

## 2. A Function to Predict $\overline{\rho_{map}^{\min}}$

Our goal is to develop an equation that uses the values of  $n$  and  $k$  to predict  $\overline{\rho_{map}^{\min}}$  from 200 RLH designs. The objective is an equation that is sufficiently simple to use with a calculator. Using data from the  $\overline{\rho_{map}^{\min}}$  table with the  $\overline{\rho_{map}^{\min}}$  formula allows the experimenter to enter different  $(n, k)$  pairs until a specific design dimension meets an acceptable nonorthogonality value.

### a. Initial Analysis of $\overline{\rho_{map}^{\min}}$

We use the  $\overline{\rho_{map}^{\min}}$  data from which we developed the tables in the previous section to construct a parsimonious predictive equation. Patterns are evident in the relationships between  $\overline{\rho_{map}^{\min}}$  and  $(n, k)$  when either  $n$  or  $k$  is constant and the other

changes. Grouping  $\overline{\rho_{map}^{\min}}$  values based on  $n$  uncovers clear patterns in the data. Figure 4 shows that as  $k$  increases and  $n$  is constant, the relationship between  $\overline{\rho_{map}^{\min}}$  and  $k$  appears logarithmic. The overlaps in the groupings of  $n$  as  $k$  increases also indicate an interaction between  $n$  and  $k$ .

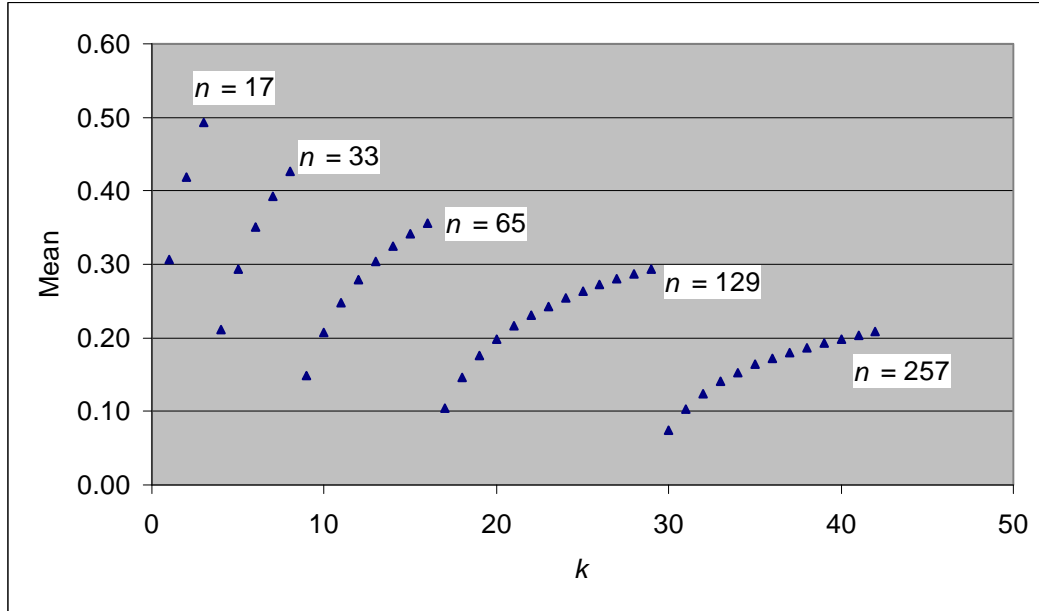


Figure 4. A logarithmic pattern appears between  $\overline{\rho_{map}^{\min}}$  and  $k$  when  $n$  is constant: Groups of  $n$  are presented in increasing order.

Similarly, Figure 5 displays an emerging pattern when  $k$  is constant and  $n$  increases. The relationship between  $\overline{\rho_{map}^{\min}}$  and  $n$  displays an exponential decay. These tendencies for  $\overline{\rho_{map}^{\min}}$  regarding  $n$  and  $k$ , respectively, provide a clue to their transformation in order to draw a linear relationship with each.

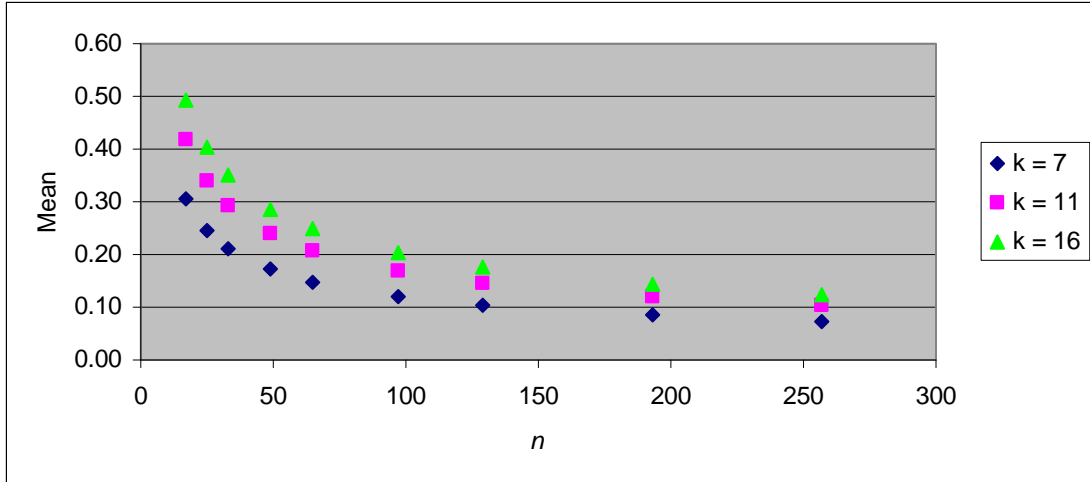


Figure 5. An exponential decay or a negative power function relationship appears between  $\overline{\rho_{map}^{\min}}$  and  $n$  when  $k$  is constant. For clarity, only a few sets for fixed  $k$  are in the chart.

Transforming  $n$  and  $k$  results in each having a nearly linear relationship with  $\overline{\rho_{map}^{\min}}$ . Owen (1994) shows that the variance of the root mean square correlation ( $\rho_{rms}$ ) of an uncorrected LH design is related to  $n^{-3}$ . We examine different transformations of  $n$  to determine its linear relationship with  $\overline{\rho_{map}^{\min}}$ . A transformation of  $n^{-2/3}$  shows a nearly linear relationship when  $k$  is constant. Figure 6 shows an example of a nearly linear relationship between  $\overline{\rho_{map}^{\min}}$  and  $n^{-2/3}$  when  $k = 7$ .

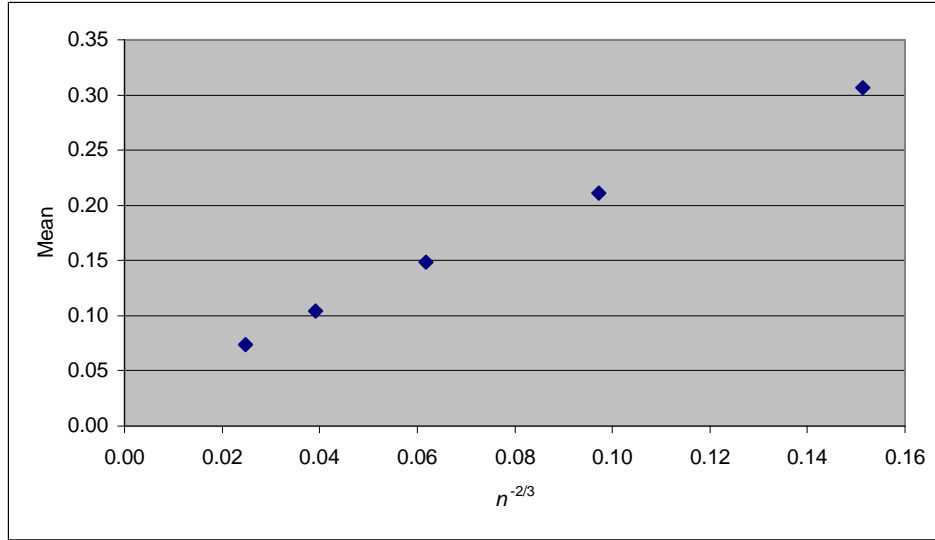


Figure 6. There is a nearly linear relationship between  $\overline{\rho_{map}^{\min}}$  and transformed  $n$ . For illustration we show the graph for  $k = 7$ .

Similarly, a transformation of  $k$  to  $k^{-1/3}$  reveals a nearly linear relationship with  $\overline{\rho_{map}^{\min}}$ , when keeping  $n$  constant. Figure 7 shows an example of a nearly linear relationship between  $\overline{\rho_{map}^{\min}}$  and  $k^{-1/3}$  for a set of data where  $n = 257$ .

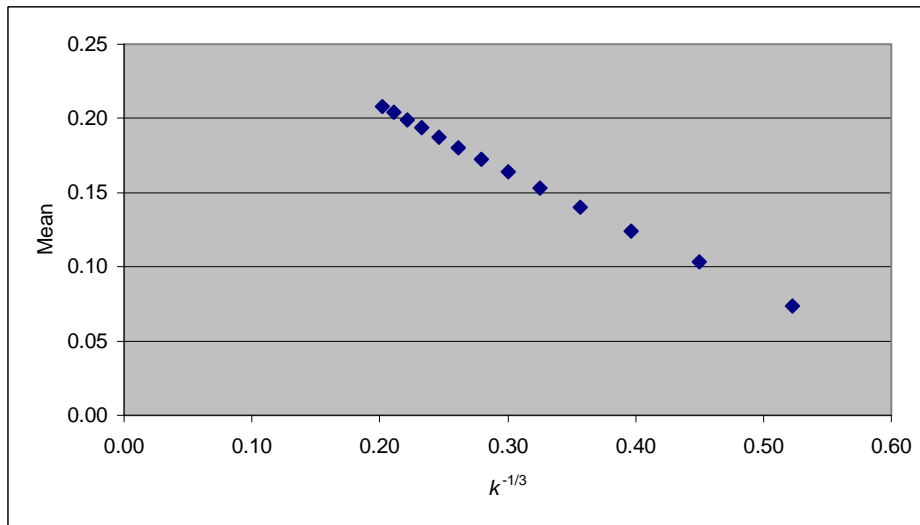


Figure 7. A nearly linear relationship appears between  $\overline{\rho_{map}^{\min}}$  and transformed  $k$ . We illustrate the case for  $n = 257$ .

Owen (1994) studies the relationship between the root mean square of an uncorrected Latin hypercube sample (LHS) and  $n$ , as  $n$  increases. Owen shows empirically that the relationship between the  $\rho_{rms}$  of an uncorrected LHS and  $n$  to be

$$\log(\rho_{rms}) = -0.5 \log(n).$$

Owen's results support the exponentially decaying relationship that we find between  $\overline{\rho_{map}^{\min}}$  and  $n$ . We note that this equation is independent of  $k$ . To complete our study, we explore the simultaneous relationships of  $n$  and  $k$  with  $\overline{\rho_{map}^{\min}}$ .

**b. Exploring Simple Linear Regression Models for  $\overline{\rho_{map}^{\min}}$**

Examination of the relationship between  $\overline{\rho_{map}^{\min}}$  and the two transformed variables,  $n^{-2/3}$  and  $k^{-1/3}$ , reveals that it is possible to fit a multiple linear regression model to the data. A study of simple linear relationships between  $\overline{\rho_{map}^{\min}}$  and each of the transformed values shows that they are nearly linear. Because of an earlier indication that an interaction exists between  $n$  and  $k$ , we also explore the relationship between  $\overline{\rho_{map}^{\min}}$  and  $n^{-2/3}k^{-1/3}$ . Fitting simple linear regression models of  $\overline{\rho_{map}^{\min}}$  versus  $n^{-2/3}$ ,  $k^{-1/3}$ , and  $n^{-2/3}k^{-1/3}$  results in each model having an  $R^2$  of 0.98 or greater.

**c. A Multiple Linear Regression Model for  $\overline{\rho_{map}^{\min}}$**

The above results indicate that a multiple linear regression model with two main effect terms and one interaction term can be a good predictor for  $\overline{\rho_{map}^{\min}}$ . To fit a multiple linear regression model we use  $\overline{\rho_{map}^{\min}}$  data from 115 different design dimensions.

The least squares fit gives an  $R^2 = 0.99$ . Our estimated predictor, which we denote by

$\left(\overline{\rho_{map}^{\min}}\right)^E$ , is given by

$$\left(\overline{\rho_{map}^{\min}}\right)^E = 0.0873 + 7.859n^{-2/3} - 0.109k^{-1/3} - 11.702n^{-2/3}k^{-1/3}.$$

*d. Adequacy of the Multiple Linear Regression Model for  $\overline{\rho_{map}^{\min}}$*

The multiple linear regression model fit adequately estimates expected  $\rho_{map}^{\min}$ . The statistics from the predicted values' residuals are given in Table 7. Of course, the residual mean is zero. A standard deviation of 0.00966 and a range of 0.05 are relatively small for the values that the model is predicting. The median and the skewness value indicate that the model tends to slightly underestimate the actual  $\overline{\rho_{map}^{\min}}$  values.

Table 7. Descriptive statistics from the residuals derived from the MLR predictions show negative skewness and a relatively small range. The model tends to slightly underestimate the value of the minimum maximum absolute pairwise correlation value of 200 RLH designs.

<b>Statistics for Residuals</b>	
Median	0.0029624
Standard Deviation	0.0096594
Kurtosis	0.8707658
Skewness	-1.0013506
Range	0.0501953
Minimum	-0.0364850
Maximum	0.0137103
Count	115.0000000

It is evident from Figure 8 that there are some curvilinear tendencies in the residual plot and the residuals do not follow a normal distribution, but the residuals are not overly large with respect to the predicted values. Another model that includes quadratic terms improves the predictive capabilities of the model, but requires a more complex expression (see Appendix E). Considering the parsimony that we desire and the relatively small errors, we accept this model.

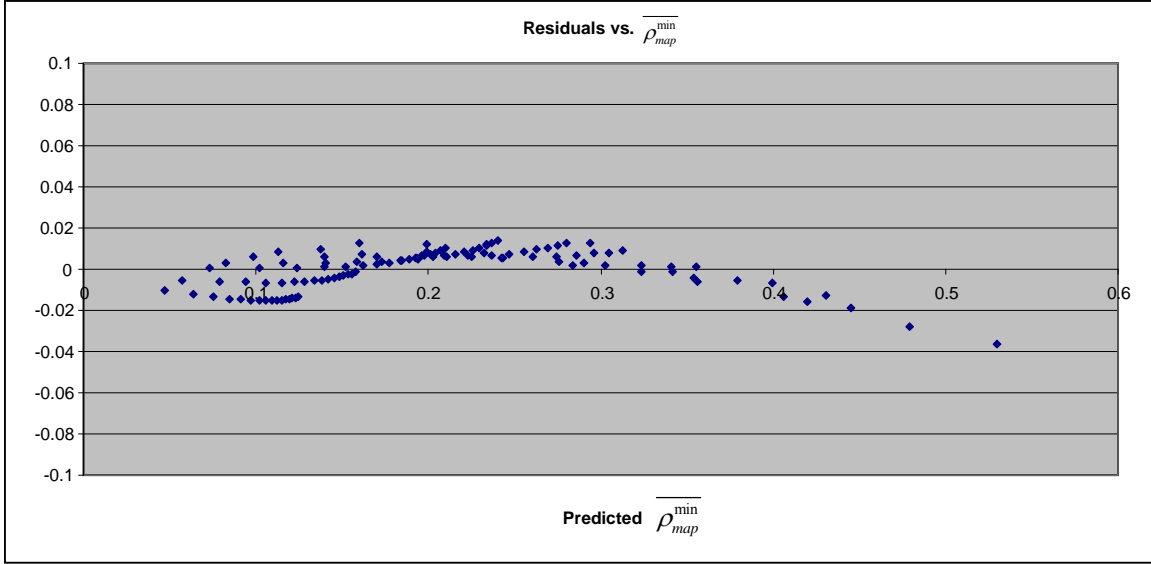


Figure 8. This residual plot of 115 different predicted  $\overline{\rho_{map}^{\min}}$  values shows curvature.

*e. Application of the  $\left(\overline{\rho_{map}^{\min}}\right)^E$  Model*

We now discuss the use of the  $\left(\overline{\rho_{map}^{\min}}\right)^E$  equation to estimate expected  $\rho_{map}^{\min}$ , in conjunction with the  $\overline{\rho_{map}^{\min}}$  table. The  $\overline{\rho_{map}^{\min}}$  table provides general guidance for the range of  $n$  and  $k$  in which a specific correlation value may reside. Because of its simplicity, the  $\left(\overline{\rho_{map}^{\min}}\right)^E$  equation can easily be incorporated into a spreadsheet to compute estimated values of  $\rho_{map}^{\min}$ , as the experimenter varies  $n$  and  $k$ . Calculations stop when the experimenter finds an  $n$  and  $k$  combination that results in an acceptable correlation value. These are the design dimensions that afford the analyst the best chance of finding an uncorrected RLH with the desired  $\rho_{map}$ . Because the equation is sufficiently simple, one can also solve for any of the three values, when the other two are fixed.

To produce the design for designated  $n$  and  $k$  we generate an RLH and compute  $\rho_{map}$ . The experimenter may use any number of software programs to produce RLH

designs and to measure  $\rho_{map}$ . Within  $G$  generations of RLH designs, we assert that the analyst will find a design with an acceptable  $\rho_{map}$ .

## B. DESIGN DEVELOPMENT WITH CORRELATION-BASED TOOLS

We employ the correlation-based tools that we have developed in Section A. A step-by-step procedure guides experimenters in implementing these tools. A constructive example illustrates the leverage that experimenters can gain from RLH designs.

### 1. The Process for Dimension Selection and RLH Design Generation

Our procedure for obtaining an experimental RLH design with an acceptable nonorthogonality measure has the following steps.

**Step 1:** Enter the  $\overline{\rho_{map}^{\min}}$  table to obtain general guidance about design dimensions. Entry into the table may occur in one of four ways:

- The experimenter may have a correlation threshold for a specific number of factors.
- The experimenter may have a limit for the number of sample runs and a correlation threshold.
- The experimenter may have a correlation threshold and an idea for the design dimensions.
- The experimenter may have a correlation threshold with little or no idea about the design dimensions.

**Step 2:** From the  $\overline{\rho_{map}^{\min}}$  table, select the range of  $n$  and  $k$  based on the correlation threshold. If required, rough interpolation within the table values is sufficient to define the ranges.

**Step 3:** Establish a spreadsheet to compute  $\left(\overline{\rho_{map}^{\min}}\right)^E$  for each design combination.

**Step 4:** Calculate  $\left(\overline{\rho_{map}^{\min}}\right)^E$  values. Best practices reveal that fixing  $k$  first, while changing  $n$ , is a good approach.



**Step 5:** Stop searching combinations when a value of  $\left(\overline{\rho_{map}^{\min}}\right)^E$  that approximately meets the correlation threshold is calculated.

**Step 6:** Record the design combination  $(n, k)$  that meets the correlation threshold.

**Step 7:** Fix the design dimension and generate up to  $G$  number of RLH designs. Measure  $\rho_{map}$  for each new design and stop when the actual  $\rho_{map}$  is satisfactorily close to the correlation threshold or all  $G$  designs have been generated. Note that this does not guarantee that the threshold is met.

Following these steps, an experimenter can obtain an acceptable design without a large investment in computational costs and time. We designate the resulting RLH design as  $R_k^n$ .<sup>5</sup>

## 2. A Hypothetical Case Requiring an Uncatalogued Design Dimension

This example clarifies the steps for determining the design dimensions and generating an acceptable RLH. An experimenter seeks an experimental design to assess  $k = 20$  factors with, at most,  $n = 100$  runs. Since a readily available  $N_{20}^{100}$  or  $O_{20}^{100}$  design does not exist, the researcher could try our methodology. The experimenter determines that an experimental design with  $\rho_{map} \leq 0.22$  is acceptable. To get an initial idea if a design that the scientist needs is possible, we enter Table 8 with correlation value, 0.22,  $k = 20$ , and  $n = 100$ . Because there are 20 factors to assess, the experimenter looks down the columns of K16 and K22, which bracket  $k = 20$ . The shaded part of Table 8 shows that the RLH design that can meet the correlation threshold will have between 65 and 129 runs, because the  $\overline{\rho_{map}^{\min}}$  values at these two sample run values contain the desired  $\rho_{map}$ . Since the number of factors is fixed,  $k = 20$ , the experimenter is concerned with the range  $65 \leq n \leq 100$ .

---

<sup>5</sup> We pattern this naming convention similar to Cioppa and Ye: Cioppa (2002) designated NOLH designs as  $N_k^n$  and Ye (1998) designated OLH designs as  $O_k^n$ . The symbol  $R_k^n$  emphasizes its random nature.

Table 8. This table is a section of a G200  $\overline{\rho_{map}^{\min}}$  table. It has values for dimensions with  $k = 22$  or less.

<b>G200</b>	<b>K7</b>	<b>K11</b>	<b>K16</b>	<b>K22</b>
<b>N17</b>	0.3066	0.4188	0.4932	NA
<b>N33</b>	0.2108	0.2931	0.3503	0.3923
<b>N65</b>	0.1485	0.2073	<b>0.2481</b>	<b>0.2793</b>
<b>N129</b>	0.1044	0.1460	<b>0.1756</b>	<b>0.1982</b>
<b>N257</b>	0.0738	0.1032	0.1240	0.1403

After approximating the design dimensions, the experimenter uses the equation  $\left(\overline{\rho_{map}^{\min}}\right)^E = 0.0873 + 7.859n^{-2/3} - 0.109k^{-1/3} - 11.702n^{-2/3}k^{-1/3}$  to compute values from different design combinations until one is satisfactorily close to the correlation threshold. As we noted earlier, if the experimenter fixes two of the values in the equation, the third may be found. We recognize that this equation provides an estimate of the expected  $\rho_{map}^{\min}$ . Since the standard deviation is small, the actual  $\rho_{map}^{\min}$  will usually be close to this average value. However, about half of the time  $\rho_{map}^{\min}$  will be above it. If the requirement is strict, we can repeat the process several times with a high probability (approximately  $1 - (0.5)^{\#trials}$ ) of obtaining the threshold. Another approach is to use a conservative value of the target  $\rho_{map}^{\min}$ .

The experimenter holds  $k = 20$  constant and incrementally increases  $n$ . Table 8 shows that at  $n = 65$ , the correlation is between 0.2481 and 0.2793. The spreadsheet results in Table 9 show that an  $R_{20}^{91}$  design can have a  $\rho_{map}$  that is approximately 0.22. In this case, it is possible to find a design that does not require all 100 sample runs. Depending on the cost of each run, the experimenter can extend the budget.

Table 9. Calculations for  $\left(\overline{\rho_{map}^{\min}}\right)^E$  using the MLR model as  $n$  changes.

$n$	$k$	$\left(\overline{\rho_{map}^{\min}}\right)^E$
81	20	0.236
83	20	0.233
85	20	0.230
87	20	0.228
89	20	0.225
<b>91</b>	<b>20</b>	<b>0.222</b>

To validate the estimate for  $\rho_{map}$  for the  $R_{20}^{91}$  design, we use R, a free software version<sup>6</sup> of the statistical package S-Plus, to generate 200  $R_{20}^{91}$  designs, while recording the  $\rho_{map}$  values. We successfully produce an RLH design with  $\rho_{map}$  equal to 0.219. It may be possible to increase  $n$  to 93, but the current result is acceptable.

Any experimenter can follow the process that we have described to construct a design that does not exist, or when there is a lack of computing power or software. An experimenter may also not require an orthogonal design and desire some correlation (Iman & Shortencarrier, 1985). Iman and Conover (1982) discuss ways to induce correlation, but RLH designs are an available alternative. Additionally, the new design dimensions that these tools provide are advantageous. The combination of  $\overline{\rho_{map}^{\min}}$  tables,  $\left(\overline{\rho_{map}^{\min}}\right)^E$  equations, and RLH generation give researchers a simple, fast, and effective means to construct experimental designs with acceptable nonorthogonality. Of course, this procedure requires a good random number generator (Law & Kelton, 1999).

### C. EFFECTS OF FEWER CANDIDATE DESIGNS ( $G$ )

There may be cases in which the experimenter does not wish to generate  $G = 200$  RLHs before selecting a suitable design. The manner in which the experimenter

---

<sup>6</sup> R is freeware. The interested reader can contact the author or check the SEED Center for Data Farming website at <http://harvest.nps.edu> for the R code.

generates RLH designs may also be a constraint. To mitigate the impact of such circumstances, we provide  $\overline{\rho_{map}^{\min}}$  tables and  $\left(\overline{\rho_{map}^{\min}}\right)^E$  expressions for different values of  $G$ . We introduce the symbol  $\left(\rho_{map}^{\min}\right)_G$  as the smallest maximum absolute pairwise correlation value in  $G$  trials. The corresponding equation to estimate the expected smallest maximum absolute pairwise correlation value in  $G$  iterations is designated as  $\left(\overline{\rho_{map}^{\min}}\right)_G^E$ .

Investigating the impact of different values of  $G$  reveals two notable observations. We empirically show that for  $G > 200$ , the values of  $\rho_{map}^{\min}$  vary only slightly. Conversely, as  $G$  decreases, the variance in  $\rho_{map}^{\min}$  increases, but not overly so. As a result, the expressions show similarity. However, the differences are important. To retain utility to experimenters, we set the lower bound for  $G$  at 10 and investigate  $G \in \{10, 25, 50, 75, 100, 125, 150, 175, 200\}$ .

With some slight modifications we use the same methodology to explore the relationship of transformed  $n$  and  $k$  values, as well as their interaction term. We fit multiple linear regressions for each  $G$ . We catalogue  $\left(\overline{\rho_{map}^{\min}}\right)_G$  tables and corresponding  $\left(\overline{\rho_{map}^{\min}}\right)_G^E$  models. In our initial effort, we use 42 design dimensions to develop the  $\left(\overline{\rho_{map}^{\min}}\right)_{G200}^E$  model. We refine the  $\left(\overline{\rho_{map}^{\min}}\right)_{G200}^E$  model fit using all 115 design combinations and further use these data to develop the equations that follow.

$$\left(\overline{\rho_{map}^{\min}}\right)_{G200}^E = 0.0873 + 7.859n^{-2/3} - 0.109k^{-1/3} - 11.702n^{-2/3}k^{-1/3} \quad (2.1)$$

$$\left(\overline{\rho_{map}^{\min}}\right)_{G175}^E = 0.0873 + 7.864n^{-2/3} - 0.109k^{-1/3} - 11.682n^{-2/3}k^{-1/3} \quad (2.2)$$

$$\left(\overline{\rho_{map}^{\min}}\right)_{G150}^E = 0.0874 + 7.870n^{-2/3} - 0.108k^{-1/3} - 11.650n^{-2/3}k^{-1/3} \quad (2.3)$$

$$\left(\overline{\rho_{map}^{\min}}\right)_{G125}^E = 0.0875 + 7.872n^{-2/3} - 0.107k^{-1/3} - 11.611n^{-2/3}k^{-1/3} \quad (2.4)$$

$$\left(\overline{\rho_{map}^{\min}}\right)_{G100}^E = 0.0875 + 7.883n^{-2/3} - 0.106k^{-1/3} - 11.578n^{-2/3}k^{-1/3} \quad (2.5)$$

$$\left(\overline{\rho_{map}^{\min}}\right)_{G75}^E = 0.0877 + 7.886n^{-2/3} - 0.105k^{-1/3} - 11.502n^{-2/3}k^{-1/3} \quad (2.6)$$

$$\left(\overline{\rho_{map}^{\min}}\right)_{G50}^E = 0.0877 + 7.912n^{-2/3} - 0.103k^{-1/3} - 11.423n^{-2/3}k^{-1/3} \quad (2.7)$$

$$\left(\overline{\rho_{map}^{\min}}\right)_{G25}^E = 0.0881 + 7.945n^{-2/3} - 0.0988k^{-1/3} - 11.270n^{-2/3}k^{-1/3} \quad (2.8)$$

$$\left(\overline{\rho_{map}^{\min}}\right)_{G10}^E = 0.0883 + 7.996n^{-2/3} - 0.0902k^{-1/3} - 11.014n^{-2/3}k^{-1/3} \quad (2.9)$$

These equations have similarity in their coefficients, but the subtleties in each expression are important. The differences result in fairly precise estimates of expected  $\rho_{map}^{\min}$  from a given  $G$  RLHs. This set of equations provide the experimenter with an option for the number of RLH designs to generate, along with design choices of  $n$ ,  $k$ , and  $\rho_{map}$ .

#### D. SUMMARY

Scientists often use uncorrected LH designs and deal with effects of having high correlation among the columns of the design matrix. Our efforts simplify the process for any experimenter to obtain a design that meets a correlation threshold, which may not necessarily be orthogonal, or nearly orthogonal. We define the  $\rho_{map}$  of an LH as a single measure of its nonorthogonality. We describe tools based on this measure, and present an algorithm that combines them with RLH generation to obtain designs with acceptable nonorthogonality. Tables for  $\overline{\rho_{map}^{\min}}$  for specific values of  $G$  are reusable, as are the formulas for  $\left(\overline{\rho_{map}^{\min}}\right)_G^E$ . The availability of packages that easily generate LHs and the ubiquity of off-the-shelf analytical software offer scientists a powerful, but simple, set of tools to obtain a desired experimental design.

### III. CORRECTED RLH USING FLORIAN'S METHOD FOR NEW DESIGNS

This chapter explores the utility of applying Florian's method to reduce correlation in LHs. We iteratively apply Florian's method to produce new OLH and NOLH designs. Cioppa (2002) conjectures that it is necessary to generate a large number of candidate LH designs from which to choose a few that meet nonorthogonality thresholds before applying correlation reduction methods. In some cases, Cioppa suggests a million or more candidate designs. We explain the relationship of the size of the experiment and the number of candidate RLH designs to generate before applying Florian's method. A study of this relationship reveals that the number of candidate designs is often much less than 1,000 for even large experiments.

#### A. FLORIAN'S CORRELATION REDUCTION METHOD

This section briefly describes Florian's method to understand its mathematical foundations (Florian, 1992). Consider an initial RLH design,  $X$ , with dimension  $n$  and  $k$ . Because the number of unique value levels for any factor in an LH is equal to the number of runs ( $n$ ) in the design matrix, there are no ties. We convert the design into a matrix of ranks,  $R$ , with the same dimension as the design matrix. The entries in each column in  $R$  are the ranks of the values for the corresponding variable.

We use the rank correlation matrix of  $R$  to construct the transformation matrix. The rank correlation matrix,  $T$ , consists of the pairwise correlations of the columns of  $R$ . The value  $T_{ij}$  is Spearman's coefficient of correlation (Conover, 1999) for the columns  $R_i$  and  $R_j$  for all pairs  $(i,j)$  in  $R$ , with

$$T_{ij} = 1 - \frac{6 \sum_{l=1}^n (R_l^i - R_l^j)^2}{n(n^2 - 1)},$$

where  $R_l^i$  is the rank of the  $l^{\text{th}}$  element in the  $i^{\text{th}}$  column. Spearman's computation of the correlation is a modification of Pearson's sample coefficient of correlation, based on

ranks when there are no ties. Thus, it is appropriate for our use since each value in the design column is unique.

Clearly  $T$  is symmetric and positive definite. Therefore, it is factorable through Cholesky's decomposition (Leon, 2002). We seek a transformation matrix,  $S'$ , such that  $STS'$  is equal to the identity matrix, where  $S = Q^{-1}$  and  $T = QQ'$ .

Determining  $Q$  involves Cholesky's method. Because  $T$  is symmetric, a lower triangular matrix,  $L$ , exists such that  $T = LDL'$ , where  $D$  is a diagonal matrix of positive values. Cholesky's decomposition states that  $L_1 = LD^{1/2}$  and that  $L_1L_1' = T$ . Therefore,  $Q = L_1$  and  $S = Q^{-1} = L_1^{-1} = (LD^{1/2})^{-1}$ , so  $S' = \left( (LD^{1/2})^{-1} \right)'$ .

Florian uses  $S'$  to update  $R$ . The new rank matrix is  $R_{new} = RS'$  and the factor level values of  $X$  are rearranged to meet the new rank structure, which usually results in reduced correlation among its columns (Florian, 1992).

## **B. DIMENSIONALITY AND FLORIAN'S METHOD**

A Florian-improved RLH (FRLH) design is the result of a relatively simple method for creating uncatalogued nearly orthogonal designs. We produce one RLH, compute its  $\rho_{map}$ , and apply Florian's method. We compute  $\rho_{map}$  for the transformed matrix and compare the two nonorthogonality measures. If there is an improvement, we reapply Florian's method. We iterate until  $\rho_{map}$  no longer improves and store the last improved design. Table 10 contains the best  $\rho_{map}$  from 20 separate instances of this process for each of many different RLH design dimensions. We highlight a number of the design dimensions that previous size restrictions prevent. Notably, we find an improved  $n = 33$ ,  $k = 16$  design that has a  $\rho_{map}$  of 0.031, which meets Ang's (2006) NOLH criteria. With Cioppa's (2002) NOLH convention, exploring 16 factors requires at least 65 runs. Ang (2006) reduces the number of runs to 33. Our unique  $n = 33$ ,  $k = 16$  design equals Ang's achievement. Additionally, our method increases  $k$  and introduces a new  $n = 33$ ,  $k = 22$  design with  $\rho_{map}$  equal to 0.034. Again using Cioppa's (2002) and

Ye's (1998) naming convention, we use the symbol  $F_k^n$  to denote an FRLH design with  $n$  runs and  $k$  factors since it may not necessarily result in an OLH or NOLH.

Table 10. The best maximum absolute pairwise correlation values from 20 FRLHs of the same design dimension are presented. Gray-shaded design dimensions emphasize intervals that this technique can fill in the OLH and NOLH catalogue, using Ang's (2006) near orthogonal threshold. Diagonal-lined areas identify dimensions yet to meet NOLH criteria.

	<i>K</i>							
<i>N</i>	7	11	16	22	29	37	46	56
17	0.054	0.061	0.105	NA	NA	NA	NA	NA
25	0.039	0.039	0.053	0.080	NA	NA	NA	NA
33	0.028	0.030	0.031	0.034	0.054	NA	NA	NA
49	0.019	0.018	0.018	0.020	0.022	0.036	0.056	NA
65	0.013	0.014	0.013	0.015	0.015	0.017	0.019	0.039
97	0.009	0.009	0.010	0.010	0.010	0.010	0.010	0.012
129	0.007	0.008	0.006	0.007	0.007	0.006	0.007	0.007
193	0.005	0.005	0.004	0.004	0.005	0.005	0.005	0.005
257	0.003	0.004	0.003	0.003	0.003	0.004	0.003	0.003
513	0.002	0.002	0.002	0.002	0.002	0.002	0.002	0.002

For design dimensions that meet our guidance, Florian's method is often sufficient to produce an NOLH design from a single RLH. Cioppa (2002) generates millions of designs before selecting a few on which to apply Florian's method. Tang (1998) empirically studies the effects of initial designs when reducing correlation. Tang finds that for small values of  $n$  and  $k$ , the initial design has a large effect on the final design's correlation value. To mitigate this effect, Tang generates three designs and chooses the one with the least correlation as the initial design. As  $n$  increases, the effects of the initial design diminish. We provide empirical evidence that, for smaller design dimensions, the initial design has greater impact.

We produce many new NOLH designs with iterative applications of Florian's method. While previous methods are constrained to one exact design for a given dimension, our technique can result in many NOLH designs for the same dimension. Additionally, this method results in many new large NOLH designs. For instance, we create an  $N_{506}^{759}$  design that has  $\rho_{map} = 0.00267$  in 50 minutes using a standard 2 gigahertz desktop computer with 3 gigabytes of RAM.



We further examine the role that design dimensions play in the effectiveness of Florian's method. Using only one candidate RLH for each design dimension, we examine 53 different design dimensions in which the ratio of  $k$  and  $n$  is less than or equal to 0.333. Of the 53 designs, 52 met NOLH criteria after iterative application of Florian's method. In all but three of the designs,  $n$  was greater than or equal to 49. After empirical evaluation, we develop a rule of thumb that any design dimension, where  $n > 50$  and  $k \leq \frac{n}{3}$ , likely results in a design that meets NOLH criteria.

A comparison of these new  $F_k^n$  designs with catalogued NOLH designs provides some gauge for the veracity of our claim regarding Florian's method. Table 11 gives  $\rho_{map}$  before and after application of Florian's method to a single RLH. We compare an  $F_{16}^{65}$  design with the actual  $\rho_{map}$  value of the catalogued NOLH design. In ten trials, we see that all ten  $F_{16}^{65}$  designs each have a  $\rho_{map}$  that is less than the  $N_{16}^{65}$   $\rho_{map}$ . We acknowledge that Cioppa also sought space-filling characteristics for his designs. We include the  $F_{16}^{65}$  condition numbers (CN), which also show improvement. Other trials for different design dimensions yield similar results. Although not all comparisons favor the  $F_k^n$  designs as having better properties than  $N_k^n$  designs, they are very similar in measured values and still meet nonorthogonality thresholds.

Table 11. Comparison of  $\rho_{map}$  values and CNs for RLH,  $F_{16}^{65}$ , and  $N_{16}^{65}$ .

	Before Florian		$F_{16}^{65}$		$N_{16}^{65}$	
	$\rho_{map}$	CN	$\rho_{map}$	CN	$\rho_{map}$	CN
Trial 1	0.3012	6.7312	0.0176	1.0929	0.0219	1.1030
Trial 2	0.4272	5.5712	0.0191	1.0992	0.0219	1.1030
Trial 3	0.4176	7.5441	0.0149	1.0852	0.0219	1.1030
Trial 4	0.3831	5.9205	0.0215	1.1028	0.0219	1.1030
Trial 5	0.3216	6.5351	0.0156	1.0818	0.0219	1.1030
Trial 6	0.3475	6.2600	0.0157	1.0853	0.0219	1.1030
Trial 7	0.3096	6.6278	0.0172	1.0893	0.0219	1.1030
Trial 8	0.4438	7.5144	0.0158	1.0949	0.0219	1.1030
Trial 9	0.4882	7.9720	0.0179	1.0911	0.0219	1.1030
Trial 10	0.3221	6.7882	0.0198	1.0901	0.0219	1.1030

These results lead to a loose constraint for applying Florian's method. When  $n > 50$  and  $k \leq \frac{n}{3}$ , we can usually transform RLH designs to meet Cioppa's (2002) NOLH criteria:  $\rho_{map} \leq 0.03$  and condition number less than or equal to 1.13, as well as Ang's (2006) thresholds of  $\rho_{map} \leq 0.05$  and condition number less than or equal to 1.20. Ang's criteria allow more latitude and we therefore adopt them as our own. We note that a departure from these guidelines still results in a reasonable reduction of the column correlation in the design. Later discussions in this study show new methods that further relax even these loose constraints.

### C. CONSTRUCTING NEARLY ORTHOGONAL LATIN HYPERCUBES

A methodical process to generate NOLH designs with desired orthogonal characteristics may be easily implemented. The first iteration of Florian results in a new design matrix,  $\mathbf{X}_{new}$ . However,  $\rho_{map}$  for this new design matrix may not be acceptable and it may be possible to improve  $\mathbf{X}_{new}$  even further. Steps to iteratively update the latest design to create a design with the smallest  $\rho_{map}$  follow.

**Step 1:** Generate an RLH,  $\mathbf{X}_{old}$ .

**Step 2:** Compute  $\rho_{map}$  of  $\mathbf{X}_{old}$  and designate as  $(\rho_{map})_{old}$ .

**Step 3:** Apply Florian's method to  $X_{old}$  to create  $X_{new}$ .

**Step 4:** Measure  $\rho_{map}$  of  $X_{new}$  and designate as  $(\rho_{map})_{new}$ .

**Step 5:** If  $(\rho_{map})_{new}$  is less than  $(\rho_{map})_{old}$ , then rename  $X_{new}$  to  $X_{old}$  and  $(\rho_{map})_{new}$  to  $(\rho_{map})_{old}$  and return to Step 3.

**Step 6:** At termination,  $X_{old}$  is the best design.

Florian's method overcomes the dimension limits that other correlation reduction methods face. To illustrate the extent that Florian's method fills the gaps in NOLH design dimensions, we explore multiple LH designs that receive this treatment. Table 12 shows  $\rho_{map}$  values for a collection of new designs after treatment with Florian's method. As in previous discussions, we do not develop designs where  $n < k$ . As expected (Owen, 1994), designs in which  $k \rightarrow n$  demonstrate an increase in  $\rho_{map}$ . This table includes a number of design dimensions that do not exist in the current catalogue of OLH and NOLH designs. To follow the guidelines for NOLH design dimensions, a design with 22 factors ( $m = 7$ ) requires 129 runs. We show that a design for 49 runs and 22 factors with  $\rho_{map}$  of 0.020 can be obtained by Florian's method. In fact, we catalogue NOLH designs for 97 and more runs, as well as for values of  $n$  between these values.

Table 12. These are the best maximum absolute pairwise correlation values from 20 FRLHs of the same design dimension. These designs are a continuation of Table 10. All of these design dimensions, as well as the design combinations between their intervals, fill many of the gaps in the OLH and NOLH catalogue.

	<i>K</i>															
<i>N</i>	7	11	16	22	29	37	46	56	67	79	92	106	121	137	154	172
17	0.054	0.061	0.105	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
25	0.039	0.039	0.053	0.080	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
33	0.028	0.030	0.031	0.034	0.054	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
49	0.019	0.018	0.018	0.020	0.022	0.036	0.056	NA	NA	NA	NA	NA	NA	NA	NA	NA
65	0.013	0.014	0.013	0.015	0.015	0.017	0.019	0.039	NA	NA	NA	NA	NA	NA	NA	NA
97	0.009	0.009	0.010	0.010	0.010	0.010	0.010	0.012	0.012	0.053	0.052	NA	NA	NA	NA	NA
129	0.007	0.008	0.006	0.007	0.007	0.006	0.007	0.007	0.008	0.010	0.018	0.030	0.058	NA	NA	NA
193	0.005	0.005	0.004	0.004	0.005	0.005	0.005	0.005	0.006	0.005	0.005	0.006	0.010	0.012	0.042	0.037
257	0.003	0.004	0.003	0.003	0.003	0.004	0.003	0.003	0.004	0.004	0.004	0.004	0.006	0.005	0.005	0.012

We conclude that, for an experiment requiring  $n > 50$  and  $k \leq \frac{n}{3}$ , we can produce many new NOLH designs with Florian's method. As Table 12 shows, we can sometimes violate this rule of thumb and still produce an NOLH, such as the  $n = 49$ ,  $k = 22$  design (shaded in Table 12) discussed previously. We also see that as  $k$  approaches  $n$ , Florian's method finds it difficult to produce a design with  $\rho_{map} \leq 0.05$ , such as the  $n = 25$ ,  $k = 22$  design that has  $\rho_{map} = 0.08$ . We explore a new approach to solve this new aspect of the experimental design problem.

#### **D. SUMMARY**

The body of work to reduce or eliminate correlations in LH designs is extensive. OLHs (Ye, 1998) and NOLHs (Cioppa, 2002) have earned reputations as highly useful designs. Historically, construction of these designs is computer intensive and time consuming. Our findings show that Florian's method and RLH generation can save the analyst a significant amount of time and computational cost. Moreover, by starting with new RLHs, we can generate many such designs and use other criteria (e.g., space-filling properties or higher-order correlations) to discriminate between multiple NOLHs. Given our simple rule of thumb, where  $n > 50$  and  $k \leq \frac{n}{3}$ , many new NOLH designs are possible with this technique. In many cases where this rule of thumb does not hold, Florian's method is still effective enough to produce a design that meets NOLH nonorthogonality criteria.

THIS PAGE INTENTIONALLY LEFT BLANK

#### IV. A MIXED INTEGER PROGRAMMING APPROACH TO MINIMIZE $\rho_{map}$

This chapter explores optimization as a means to construct orthogonal Latin hypercube (OLH) and nearly OLH (NOLH) designs, thereby further expanding the situations in which they can be used. We are aware of the difficulties that an optimization approach presents for solving the experimental design problem. As  $n$  and  $k$  increase, the dimensionality of LHs makes optimization an unattractive option. As Chapter I explains, a vast number of unique LHs are possible for even moderate values of  $n$  and  $k$ . If an exhaustive search were applied, even moderately large values of  $n$  and  $k$  would prove to be impossible to solve to optimality. For instance, to develop an  $N_{11}^{33}$  design with an optimization model that considers all possible permutations may involve an algorithm that can explore up to  $(33!)^{11}$  LHs to guarantee an optimal solution. Techniques such as the simplex method (Nash & Sofer, 1996) are more efficient in finding an optimal answer when all variables are continuous. However, nonlinear and integer methods run into a “combinatorial explosion” (Wolsey, 1998) that make solving the complete design of experiment problem computationally intractable as  $n$  and  $k$  grow large. Therefore, we develop a construction methodology based on a focused optimization routine, which greatly expands the range of values for  $n$  and  $k$  for which we can create orthogonal and nearly orthogonal designs.

We combine RLH generation, Florian’s correlation reduction method, and optimization of a mixed integer programming problem to develop a new algorithm that relaxes existing size restrictions and fills many gaps that exist in the OLH and NOLH library. We continue to use  $\rho_{map}$  as the key measure for discriminating between designs. To best illustrate the power of this new methodology, we concentrate on design dimensions that are absent and most difficult to generate with previous methods; that is, experiments with  $n \leq 50$  and  $k$  approaches  $n$ .

## A. SIMPLIFYING THE COMPUTATION FOR NONORTHOGONALITY

In explaining our optimization approach to address the DOE problem, we find it is more useful to discuss LHs in terms of the ranks that correspond to each factor's value levels. Because LH designs assume all variables are continuous, the experimenter simply sets the number of unique values to equal the number of runs. Unique factor values ensure that there are no ties in the ranks of any column in the design. Consequently, for a design with  $n$  runs, each column in the LH design is merely a permutation of integers 1 to  $n$ .

Considering values as ranks from 1 to  $n$  simplifies computations. For instance, the average value in any column,  $l$ , is always  $\bar{x}_l = \bar{x} = \frac{n+1}{2}$ . The corresponding variance for any column is  $\sigma_l^2 = \sigma^2 = \frac{1}{n} \sum_{i=1}^n (x_i^l - \bar{x})^2 = \frac{n^2-1}{12}$ , i.e., a constant, for any  $n$ . The covariance between columns  $l$  and  $m$  is  $\sigma_{lm} = \frac{1}{n-1} \sum_{i=1}^n (x_i^l - \bar{x})(x_i^m - \bar{x})$ . Therefore, the correlation between columns  $l$  and  $m$  is

$$\rho_{lm} = \frac{\sigma_{lm}}{\sqrt{\sigma_l^2 \sigma_m^2}} = \frac{\sum_{i=1}^n \left(x_i^l - \frac{n+1}{2}\right) \left(x_i^m - \frac{n+1}{2}\right)}{\sum_{i=1}^n \left(i - \frac{n+1}{2}\right)^2}. \quad (4.1)$$

For any dimension  $n$  and  $k$ , we wish to find a design matrix,  $X_{n \times k}$ , with the smallest maximum absolute pairwise correlation  $\rho_{map}$ . We use an optimization model and algorithm to help minimize  $\rho_{map}$ .

## B. A LINEAR FORMULATION OF THE MATHEMATICAL MODEL

From Equation 4.1, the denominator is constant for all  $l$  and  $m$ . The most important features of the equation are in the numerator. Therefore, minimizing  $\rho_{map}$  is equivalent to minimizing

$$v_{map} \equiv \max_{l \neq m} \left| \sum_{i=1}^n (x_i^l - \bar{x})(x_i^m - \bar{x}) \right|. \quad (4.2)$$

Our task to minimize  $v_{map}$  is constrained by the requirement that each column of  $\mathbf{X}$  is a permutation of the first  $n$  natural numbers. Difficulties in formulation arise in the nonlinearity and nondifferentiability of the absolute-value function. To construct an OLH or NOLH with dimensions  $n$  and  $k$  using optimization techniques, we develop a linear mathematical model for Equation 4.2.

## 1. An Initial Mathematical Model

We follow the Naval Postgraduate School Operations Research format (Dell, 2004) for presenting an optimization problem. We annotate the objective and constraint equations in these formulations for reference. Our initial formulation of the problem follows.

### Model 1 ( $n, k$ ):

#### INDICES

$i$  runs (alias  $j$ )<sup>7</sup>  $i = 1, \dots, n$

$l$  factors (alias  $m$ )  $l = 1, \dots, k$

#### VARIABLES

$X_i^l$  level of  $l^{\text{th}}$  factor on the  $i^{\text{th}}$  run

$V$  variable for  $v_{map}$

#### FORMULATION

$$\min \quad V \quad (\text{A0})$$

$$\text{s.t.} \quad V \geq \sum_{i=1}^n (X_i^l - \bar{x})(X_i^m - \bar{x}) \quad \forall l \neq m \quad (\text{A1})$$

$$V \geq -\sum_{i=1}^n (X_i^l - \bar{x})(X_i^m - \bar{x}) \quad \forall l \neq m \quad (\text{A2})$$

$$X_i^l \neq X_j^l \quad \forall i \neq j, l \quad (\text{A3})$$

$$1 \leq X_i^l \leq n, \text{ integer} \quad \forall i, l \quad (\text{A4})$$

The objective function A0 is simply  $V$ ; our constraints force  $V$  to assume the value  $v_{map}$  at our approved solution. Because the absolute value is a nonlinear expression,

---

<sup>7</sup> The alias of an index,  $i$ , allows the programmer to use a different index,  $j$ , to refer to the same set of numbers.



constraints A1 and A2 separate it into terms that we can more readily transform into linear terms. Constraints A3 and A4 require that each column of  $X$  be a permutation of the integers from 1 to  $n$ . Constraints A1 and A2 are nonlinear, and constraints A3 and A4 are nonconvex.

## 2. Transformation of a Nonlinear Optimization Problem

We reformulate our mathematical model with the goal of converting it into a linear optimization problem. The advantage of linear functions is that they are proven to be convex and thus have one or more extreme points that correspond to maximum or minimum objective values (Bazaraa et al., 2005).

Optimization frequently centers on solving convex problems, of which linear types are most easily determined. The general problem of convex optimization is to find the minimum of a convex or quasiconvex function on a finite-dimensional convex space, specified by a set of extreme points and extreme rays or vectors (Bazaraa et al., 2005). As so defined, a maximum or minimum value of a convex function may be found by systematically examining extreme points from the convex space (Bertsimas & Tsitsiklis, 1997; Nash & Sofer, 1998).

The constraints of the mathematical model define the feasible region of the objective function. A set of convex or quasiconvex constraints result in a convex or quasiconvex feasible region, which contains extreme points and an optimal solution. Reciprocally, a feasible region resulting from one or more nonconvex constraints results in a nonconvex feasible region, which cannot guarantee an optimal solution. Unfortunately, we find the design of experiment problem to be nonlinear and nonconvex.

Our first step to linearize the problem is to reformulate A3 by introducing binary variables. We represent one run (design point)  $i$  of one factor  $l$  with a set of  $n$  binary variables, one for each possible level  $j$  of the factor on that run. For a binary variable,  $Y_{i,j}^l$ , if the factor  $l$  is set to the  $j^{\text{th}}$  level in the  $i^{\text{th}}$  run, we let the variable equal one, and zero otherwise. Therefore,  $Y_{3,7}^4 = 1$  means that for the 4<sup>th</sup> factor, the level is set at 7 in the 3<sup>rd</sup> run of the experiment. The addition of these variables requires new constraints to control the values that they may assume. Our revised model follows.

**Model 2 ( $n, k$ ):****VARIABLES**

$Y_{i,j}^l$  equals 1 (0 otherwise) if factor  $l$  is set to level  $j$  in run  $i$

$V$  variable for  $v_{map}$

**FORMULATION**

$$\min \quad V \quad (\text{B0})$$

$$\text{s.t.} \quad V \geq \sum_{i=1}^n \left( \sum_{j=1}^n j Y_{i,j}^l - \bar{x} \right) \left( \sum_{j=1}^n j Y_{i,j}^m - \bar{x} \right) \quad \forall l, m; l \neq m \quad (\text{B1})$$

$$V \geq - \sum_{i=1}^n \left( \sum_{j=1}^n j Y_{i,j}^l - \bar{x} \right) \left( \sum_{j=1}^n j Y_{i,j}^m - \bar{x} \right) \quad \forall l, m; l \neq m \quad (\text{B2})$$

$$\sum_{j=1}^n Y_{i,j}^l = 1 \quad \forall i, l \quad (\text{B3})$$

$$\sum_{i=1}^n Y_{i,j}^l = 1 \quad \forall j, l \quad (\text{B4})$$

$$Y_{i,j}^l \in \{0, 1\} \quad \forall i, j, l \quad (\text{B5})$$

Constraints B1 and B2 are reformulations of A1 and A2. Constraints for B3 require that exactly one level be chosen for each factor, for each run. Constraints B4 ensure that each level is chosen exactly one time for each factor. These constraints are linear since they are nothing more than a summation of simple variables. However, constraints B1 and B2 are still nonlinear because they involve the product of  $Y_{i,j}^l$  and  $Y_{i,j}^m$ .

**3. A Complete Linear Formulation of the Optimization Problem**

There are several ways to deal with the nonlinearity in B1 and B2. One is to develop derivative-free heuristic algorithms (including the many varieties of local search) that generate many design matrices, and evaluate  $V$  for each one. Evaluating  $V$  is much easier than optimizing it, but we recognize that heuristics do not guarantee an optimal solution, although they sometimes result in one.

Our approach to the difficulties associated with constraints B1 and B2 is to optimize one column (factor) of the design matrix at a time. We find level settings for a

given factor,  $m$ , that minimizes  $v_{map}$  between  $m$  and every other (fixed) column in the design. Therefore, variables corresponding to factor  $m$  are the only ones that appear in the model, and we treat all other factors in the design as having “fixed” values, denoted  $\hat{X}_i^l$ . The resulting levels for the targeted factor  $m$  guarantees the lowest value for  $v_{map}$ , given that all other factor levels remain constant. This final formulation is a single-column optimization program for a specified column,  $m$ , in a design with  $n$  runs and a total of  $k$  factors.

**Model 3 ( $m, k, n$ ):**

**DATA**

$\hat{X}_i^l$  level of factor  $l$  for run  $i$  in design  $X$

**FORMULATION**

$$\min \quad V \quad (C0)$$

$$\text{s.t.} \quad V \geq \sum_{i=1}^n (\hat{X}_i^l - \bar{x}) \left( \sum_{j=1}^n jY_{i,j}^m - \bar{x} \right) \quad \forall l \neq m \quad (C1)$$

$$V \geq -\sum_{i=1}^n (\hat{X}_i^l - \bar{x}) \left( \sum_{j=1}^n jY_{i,j}^m - \bar{x} \right) \quad \forall l \neq m \quad (C2)$$

$$\sum_{j=1}^n Y_{i,j}^l = 1 \quad \forall i, l \quad (C3)$$

$$\sum_{i=1}^n Y_{i,j}^l = 1 \quad \forall j, l \quad (C4)$$

$$Y_{i,j}^l \in \{0, 1\} \quad \forall i, j, l \quad (C5)$$

Constraints C1 and C2 are now linear in  $Y^m$ . Several of the terms in these constraints are constant, and can be further simplified:

$$\begin{aligned} \sum_{i=1}^n (\hat{X}_i^l - \bar{x}) \left( \sum_{j=1}^n jY_{i,j}^m - \bar{x} \right) &= \sum_{i=1}^n (\hat{X}_i^l - \bar{x}) \sum_{j=1}^n jY_{i,j}^m - (\bar{x}) \sum_{i=1}^n (\hat{X}_i^l - \bar{x}) \\ &= \sum_{i=1}^n (\hat{X}_i^l - \bar{x}) \sum_{j=1}^n jY_{i,j}^m \end{aligned} \quad (4.3)$$

This simplification yields our final model.

**Final model 3 ( $m, n, k$ ):**

**INDICES**

$i$  runs (alias  $j$ )  $i = 1, \dots, n$   
 $l$  factors  $l = 1, \dots, k$

**DATA**

$\hat{X}_i^l$  level value of factor  $l$  for run  $i$  in the given design  $X$

**VARIABLES**

$Y_{i,j}^m$  equals 1 (0 otherwise) if factor  $m$  is set to level  $j$  in run  $i$   
 $V$  maximum absolute value of  $v_{map}$  for any two columns in  $X$

**FORMULATION**

$$\min V \quad (C0)$$

$$\text{s.t. } V \geq \sum_{i=1}^n (\hat{X}_i^l - \bar{x}) \sum_{j=1}^n j Y_{i,j}^m \quad \forall l \neq m \quad (C1)$$

$$V \geq -\sum_{i=1}^n (\hat{X}_i^l - \bar{x}) \sum_{j=1}^n j Y_{i,j}^m \quad \forall l \neq m \quad (C2)$$

$$\sum_{j=1}^n Y_{i,j}^m = 1 \quad \forall i \quad (C3)$$

$$\sum_{i=1}^n Y_{i,j}^m = 1 \quad \forall j \quad (C4)$$

$$Y_{i,j}^m \in \{0,1\} \quad \forall i, j \quad (C5)$$

For the remainder of this document, we refer to the optimization program based on the final model as **VMIN**: a *minimization* model for the variable  $V$ .

**C. OPERATIONALIZING THE LINEAR OPTIMIZATION PROBLEM**

This final formulation is key to our search for new designs with orthogonal or nearly orthogonal properties. We encode VMIN into GAMS (General Algebraic Modeling System, 2008), a powerful high-level modeling system for mathematical programming and optimization, and solve using the commercially available solver Cplex (2008). GAMS consists of a language compiler and a suite of integrated high-

performance solvers that are specifically designed for modeling large linear, nonlinear, and mixed integer optimization problems. The result is a robust programming language that can solve problems with thousands of linear constraints and thousands of variables.

### 1. The Size and Complexity of VMIN

Because the complexity of even relatively small problems has the potential of taxing the model to a point that prevents it from converging to a solution, we deliberately choose to focus on one factor column to solve. Equations from C3 in VMIN correspond to a total of  $n$  equations—one for each run in the design. Because  $m$  is the targeted column, it contains the only set of variables in the problem—one for each level of the factor  $m$ . We illustrate with a linear equation for C3, where the targeted (shaded) column is  $m = 3$  and run number is  $i = 2$ . If we set  $n = 5$  runs, there will be five value levels, which the variables  $Y_{i,j}^m$ ,  $j = 1, 2, 3, 4, 5$  represent in the shaded column of Table 13. Column (k#) and row (n#) labels in the table are for ease of reference.

Table 13. We present how the parameters of a design and the design’s initial values translate into the constraints in VMIN to optimize the values of the targeted variable column ( $m = 3, k = 4, n = 5$ ), where value levels for column 3 remain as variables and all other values are fixed.

	k1	k2	k3	k4
n1	3	4	$Y_{1,j}^3$	4
n2	4	2	$Y_{2,j}^3$	5
n3	1	3	$Y_{3,j}^3$	2
n4	2	5	$Y_{4,j}^3$	1
n5	5	1	$Y_{5,j}^3$	3

The constraint for this instance of C3 is  $Y_{2,1}^3 + Y_{2,2}^3 + Y_{2,3}^3 + Y_{2,4}^3 + Y_{2,5}^3 = 1$ . We see in this equation that only one of the variables may equal one—that is, only one value level from factor 3 can be assigned to the second run of the experiment.

The following discussion explains the constraints involved in the linear formulation of the problem. For constraints C1 and C2, we consider

$$\bar{x} = \frac{n+1}{2} = \frac{5+1}{2} = 3 \text{ and the fixed values for columns } k1, k2, \text{ and } k4 \text{ of the initial design}$$

to construct the following pseudo expression for  $v_{map}$ .

$$V \geq \sum_{i=1}^n (\hat{X}_i^l - \bar{x}) \sum_{j=1}^n jY_{i,j}^m \quad \forall l \neq m$$

$$V \geq (3-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3) + (4-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3) + (1-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3) + (2-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3) + (5-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3)$$

$$V \geq (4-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3) + (2-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3) + (3-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3) + (5-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3) + (1-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3)$$

$$V \geq (4-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3) + (5-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3) + (2-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3) + (1-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3) + (3-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3)$$

$$V \geq - \sum_{i=1}^n (\hat{X}_i^l - \bar{x}) \sum_{j=1}^n jY_{i,j}^m \quad \forall l \neq m$$

$$V \geq - \left( \begin{aligned} &(3-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3) + (4-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3) + \\ &(1-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3) + (2-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3) + \\ &(5-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3) \end{aligned} \right)$$

$$V \geq - \left( \begin{aligned} &(4-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3) + (2-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3) + \\ &(3-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3) + (5-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3) + \\ &(1-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3) \end{aligned} \right)$$

$$V \geq - \left( \begin{aligned} &(4-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3) + (5-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3) + \\ &(2-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3) + (1-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3) + \\ &(3-3)(1Y_{1,1}^3 + 2Y_{1,2}^3 + 3Y_{1,3}^3 + 4Y_{1,4}^3 + 5Y_{1,5}^3) \end{aligned} \right)$$

Constraints for C3 ensure that each run has exactly one value level assigned.

$$\sum_{j=1}^n Y_{i,j}^m = 1 \quad \forall i$$

$$Y_{1,1}^3 + Y_{1,2}^3 + Y_{1,3}^3 + Y_{1,4}^3 + Y_{1,5}^3 = 1$$

$$Y_{2,1}^3 + Y_{2,2}^3 + Y_{2,3}^3 + Y_{2,4}^3 + Y_{2,5}^3 = 1$$

$$Y_{3,1}^3 + Y_{3,2}^3 + Y_{3,3}^3 + Y_{3,4}^3 + Y_{3,5}^3 = 1$$

$$Y_{4,1}^3 + Y_{4,2}^3 + Y_{4,3}^3 + Y_{4,4}^3 + Y_{4,5}^3 = 1$$

$$Y_{5,1}^3 + Y_{5,2}^3 + Y_{5,3}^3 + Y_{5,4}^3 + Y_{5,5}^3 = 1$$

Constraints for C4 ensure that each value level is assigned exactly one time.

$$\sum_{i=1}^n Y_{i,j}^m = 1 \quad \forall j$$

$$Y_{1,1}^3 + Y_{2,1}^3 + Y_{3,1}^3 + Y_{4,1}^3 + Y_{5,1}^3 = 1$$

$$Y_{1,2}^3 + Y_{2,2}^3 + Y_{3,2}^3 + Y_{4,2}^3 + Y_{5,2}^3 = 1$$

$$Y_{1,3}^3 + Y_{2,3}^3 + Y_{3,3}^3 + Y_{4,3}^3 + Y_{5,3}^3 = 1$$

$$Y_{1,4}^3 + Y_{2,4}^3 + Y_{3,4}^3 + Y_{4,4}^3 + Y_{5,4}^3 = 1$$

$$Y_{1,5}^3 + Y_{2,5}^3 + Y_{3,5}^3 + Y_{4,5}^3 + Y_{5,5}^3 = 1$$

Finally, constraints for C5 ensure that each binary variable assumes only the value of 1 or

0. It requires a set of two constraints for each variable.

$$Y_{i,j}^m \in \{0,1\} \quad \forall i, j$$

$$\begin{array}{ll}
Y_{1,1}^3 \leq 1, Y_{1,1}^3 \geq 0 & Y_{3,4}^3 \leq 1, Y_{3,4}^3 \geq 0 \\
Y_{1,2}^3 \leq 1, Y_{1,2}^3 \geq 0 & Y_{3,5}^3 \leq 1, Y_{3,5}^3 \geq 0 \\
Y_{1,3}^3 \leq 1, Y_{1,3}^3 \geq 0 & Y_{4,1}^3 \leq 1, Y_{4,1}^3 \geq 0 \\
Y_{1,4}^3 \leq 1, Y_{1,4}^3 \geq 0 & Y_{4,2}^3 \leq 1, Y_{4,2}^3 \geq 0 \\
Y_{1,5}^3 \leq 1, Y_{1,5}^3 \geq 0 & Y_{4,3}^3 \leq 1, Y_{4,3}^3 \geq 0 \\
Y_{2,1}^3 \leq 1, Y_{2,1}^3 \geq 0 & Y_{4,4}^3 \leq 1, Y_{4,4}^3 \geq 0 \\
Y_{2,2}^3 \leq 1, Y_{2,2}^3 \geq 0 & Y_{4,5}^3 \leq 1, Y_{4,5}^3 \geq 0 \\
Y_{2,3}^3 \leq 1, Y_{2,3}^3 \geq 0 & Y_{5,1}^3 \leq 1, Y_{5,1}^3 \geq 0 \\
Y_{2,4}^3 \leq 1, Y_{2,4}^3 \geq 0 & Y_{5,2}^3 \leq 1, Y_{5,2}^3 \geq 0 \\
Y_{2,5}^3 \leq 1, Y_{2,5}^3 \geq 0 & Y_{5,3}^3 \leq 1, Y_{5,3}^3 \geq 0 \\
Y_{3,1}^3 \leq 1, Y_{3,1}^3 \geq 0 & Y_{5,4}^3 \leq 1, Y_{5,4}^3 \geq 0 \\
Y_{3,2}^3 \leq 1, Y_{3,2}^3 \geq 0 & Y_{5,5}^3 \leq 1, Y_{5,5}^3 \geq 0 \\
Y_{3,3}^3 \leq 1, Y_{3,3}^3 \geq 0 &
\end{array}$$

VMIN constraints become more complex as  $n$  and  $k$  increase. There are 25 binary variables, 1 continuous (objective) variable, and 66 constraints for this relatively small design. The size of the problem increases quadratically in the dimensions. Consider a problem  $n = 33$  and  $k = 11$ . The problem now involves nearly 1,100 equations and almost 1,000 binary variables to determine an optimal arrangement of value levels for just **one** factor,  $m$ . The solution is a subset of variables of size  $n$ , each equaling one, and the remaining  $n(n - 1)$  variables being zero.

## 2. Applying RLH and Florian's Method to Initiate VMIN

Random Latin hypercube (RLH) generation and Florian's correlation reduction method provide an excellent initial solution to VMIN. Preliminary computations to support an optimization process are categorized as heuristic algorithms—methods to reach suboptimal solutions (Bertsimas & Tsitsiklis, 1997).

A good initial solution aids the optimization routine to solve the problem more quickly and with a greater chance of reaching a low  $v_{map}$ . In our process, we obtain a good initial solution by quickly generating many RLHs—through  $k$  random permutations



of the first  $n$  natural numbers—and selecting the one with minimum  $\rho_{map}$ . Then, we iteratively apply Florian’s (1992) proven correlation reduction method to create an FRLH that serves as the starting solution for VMIN. In practice, for designs of the scale in this paper (or a little larger), it takes only seconds on a modern desktop processor to generate hundreds, or even thousands, of RLHs for designs with scores of variables. The FRLH improvement on the best RLH usually takes a few minutes. It typically takes on the order of a few hours to generate the final design—which likely requires several iterations of VMIN. Additional columns can be added by appending a new random column and then iteratively applying VMIN. For the size of the experiments we discuss, it usually takes on the order of tens of minutes to optimize the new column.

Table 14 shows a design with  $n = 17$  runs and  $k = 7$  factors. The associated objective value,  $V$ , for the initial design corresponds to  $\rho_{map} = 0.066$ . Table contents are the actual levels for the column factor for each run. There are  $17^2 = 289$  variables for each specific factor,  $m$ , in this problem:  $Y_{1,1}^6, Y_{1,2}^6, \dots, Y_{12,4}^6, Y_{12,5}^6, \dots, Y_{17,16}^6, Y_{17,17}^6$ . Table 15 illustrates the initial binary value for  $Y_{2,4}^6 = 1$ —meaning that the actual level value for factor 6 on the 2<sup>nd</sup> run is 4, and reciprocally  $Y_{2,j}^6 = 0$  for all  $j \neq 4$ .

Table 14. This  $n = 17, k = 7$  design,  $\rho_{map} = 0.066$ , is a result from generating numerous RLHs, selecting the design with the least nonorthogonality, and applying Florian's method iteratively. It is the initial design before using VMIN.

	<b>k1</b>	<b>k2</b>	<b>k3</b>	<b>k4</b>	<b>k5</b>	<b>k6</b>	<b>k7</b>
<b>n1</b>	15	2	7	5	8	9	15
<b>n2</b>	17	15	11	11	6	4	14
<b>n3</b>	5	7	8	14	2	16	4
<b>n4</b>	13	10	1	9	15	13	5
<b>n5</b>	11	6	17	3	12	15	1
<b>n6</b>	3	14	14	13	10	6	10
<b>n7</b>	7	16	2	6	11	14	12
<b>n8</b>	8	9	13	1	16	7	8
<b>n9</b>	1	11	5	2	7	8	6
<b>n10</b>	14	13	3	15	5	11	3
<b>n11</b>	10	5	12	10	4	1	2
<b>n12</b>	2	1	4	8	3	2	11
<b>n13</b>	4	3	6	16	17	12	16
<b>n14</b>	6	12	16	17	13	10	9
<b>n15</b>	9	17	10	4	9	3	13
<b>n16</b>	16	4	9	12	14	5	7
<b>n17</b>	12	8	15	7	1	17	17

For a more expansive look into this example, we continue to focus on column  $m = 6$  of the initial design. The initial  $Y_{i,j}^m$  solution set for  $m = 6$  is more easily seen in Table 15 as a two-dimensional display of variable values that show exactly one factor level ( $j$ ) assigned to each run ( $n$ ). This matrix of zeroes and ones corresponds to the values in column vector (k6) from Table 14.

Table 15. Binary variable values for factor  $m = 6$  in the initial design show the sparseness of this single column optimization problem. A set of these variables corresponds to each factor.

$m = 6$	j1	j2	j3	j4	j5	j6	j7	j8	j9	j10	j11	j12	j13	j14	j15	j16	j17
n1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
n2	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
n3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
n4	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
n5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
n6	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
n7	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
n8	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
n9	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
n10	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
n11	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
n12	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
n13	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
n14	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
n15	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
n16	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
n17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

We apply VMIN to target factor  $m = 6$  to improve  $V$ . In the following paragraphs we discuss the root mean square (Owen, 1994) to explain the rationale for selecting  $m = 6$  as the column to optimize. VMIN focuses on the problem of populating the above matrix with ones in the appropriate cells, and zero elsewhere. VMIN considers each of the 289 variables in the problem, and through the Cplex solver, determines a solution (i.e., an arrangement of values of “1”s in the matrix) that minimizes  $V$  to within a user-specified tolerance of the optimal solution.

Tolerance is the difference between the notional optimal value of the objective function and the actual computed value, divided by the notional value, and multiplied by 100%. Common practice sets tolerance at 10%, but we use 1% in VMIN to approach orthogonality. We adjust tolerance as design dimensions near saturation.

### 3. Iterative Application of VMIN

Iterative application of VMIN constructs the entire design. Solving one instance of VMIN guarantees an optimal solution for the current factor of interest. However, our simplification of the optimization problem to one column leaves us with a nonoptimal solution for the overall DOE problem. To reach orthogonality or near orthogonality requires a deliberate process. Therefore, after determining the best value levels for one column, we select a new column to optimize. We repeat this process until we have an NOLH, or there is no further progress.

Selecting the next column to optimize can become complex. There are other methods for selecting the next column to optimize, such as selecting a column that is associated with  $\rho_{map}$ , but this value exists for at least one other column. It is possible that even more than two columns are associated with  $\rho_{map}$ . Instead, following Owen (1994), we use root mean square (*rms*) as the measure for selecting the next column to optimize. Computing *rms* for a column  $m$  involves summing the squared correlation coefficients between  $m$  and all other columns in the design, and dividing the total with the number of unique column pairs, which yields

$$\rho_{rms}^2(m) = \frac{\sum_{l \neq m}^k \rho_{lm}^2}{k-1},$$

where  $\rho_{lm}$  is the correlation coefficient in (1.1), and  $k$  is the number of columns in the design. We select the column with the largest *rms* as the one to optimize. The greatest *rms* identifies the column that has, on average, the largest squared correlation with any other column, and is therefore the most problematic, relative to all other columns. Selecting this column for optimization increases the chances for reducing the design's overall nonorthogonality. In practice, this approach has proven straightforward, tractable, and successful. Although iterative column optimization may not terminate at a solution that equates to the lowest possible  $\rho_{map}$ , it provides a near optimal solution that has a very good chance of meeting NOLH criteria.

Numerous difficulties emerge when using optimization to solve the complete DOE problem. It is a primary reason that scientists have sought other methods for

developing experimentation schemes. Our presentation of a solution for  $m = 6$  in Table 15 offers a glimpse of the issues that an optimization approach faces. The two-dimensional matrix presents the VMIN attempt to position “1”s where *no two entries are in the same row or column*. Taking this same problem vertically for each factor further complicates the arrangement of “1”s. A new restriction that no two-factor ( $m$ ) matrices should have the same run ( $n$ ) and level ( $j$ ) cell filled complicates the problem sufficiently to challenge the most robust optimization schemes. Such a problem requires  $m$  sets of  $n^2$  variables moving simultaneously. The resulting problem is nonlinear and nonconvex.

Restricting the optimization problem to one factor at a time allows us to transform it into a linear one. Our formulation offers a means for reaching a focused solution that guarantees an optimal solution for any given column,  $m$ . The method that we employ to generate an initial solution is grounded in optimization theory and practical experience. Our iterative application of VMIN methodically reduces  $\rho_{map}$  to an acceptable level. Short of optimality, creation of new designs that meet NOLH criteria satisfies our study goals. This method breaks free from previous constraints to produce NOLHs in design dimensions that other approaches do not.

#### **D. NEW DESIGNS FROM VMIN**

This section describes the full impact of combining FRLH and optimization into a new methodology. Their synthesis produces new NOLH designs. We quantify our method’s effectiveness and present excerpts of new NOLH designs that are not possible to construct using earlier techniques. The relative ease and flexibility of this technique make it an attractive option for constructing new and efficient computer experiments.

As we previously stated, small NOLH designs ( $n \leq 50$ ) are difficult to generate if  $k \geq \frac{n}{3}$ . Cioppa’s (2002) method generates millions of LHs before selecting one that warrants application of Florian’s method. In practice, Florian’s method successfully

produces NOLH designs for  $n > 50$  and  $k \leq \frac{n}{3}$  with a single RLH generation,<sup>8</sup> but cannot guarantee an NOLH for dimensions that violate these combinatorial conditions.

Our optimization program is a complement to FRLH. It not only provides a means to create NOLH designs with dimensions  $n \geq 50$ ,  $\frac{n}{3} \leq k < n$ , it covers design dimensions for any  $n$ , and  $k \rightarrow n$ . Designs in Appendix A demonstrate our technique's utility.

Cioppa and Lucas (2007) discuss a number of OLH and NOLH designs and compare their orthogonal properties with the average correlation measure of simple LHs. An excerpt from their article shown in Table 16 shows their dimensions and corresponding orthogonal characteristics.

Table 16. An excerpt from Cioppa and Lucas (2007) compares RLHs and Cioppa's (2002) OLH and NOLH designs with respect to their orthogonal characteristics.

Design	Max Pairwise Correlation
$O_{11}^{33}$	0
Best $N_{11}^{33}$	0.0234
Mean $R_{11}^{33}$	0.4015
$O_{16}^{65}$	0
Best $N_{16}^{65}$	0.0219
Mean $R_{16}^{65}$	0.3194
$O_{22}^{129}$	0
Best $N_{22}^{129}$	0.0015
Mean $R_{22}^{129}$	0.2332

Gaps in the current OLH and NOLH library exist because of the rigidity of dimensional constraints. To achieve orthogonality, Cioppa's existing convention for design dimensions requires an increase in the number of runs from 33 to 65 when the

---

<sup>8</sup> In practice, one RLH generation is nearly always sufficient to produce a NOLH using Florian's method. However, choosing the RLH with the least maximum absolute pairwise correlation, from up to 1,000 RLHs, helps increase the chances of producing an NOLH.

number of factors increases from 11 to 12. In this example, there is a breach in the OLH and NOLH library for designs that have less than 17 runs and  $1 \leq k \leq 7$ , or for designs  $18 \leq n \leq 32$  and  $8 \leq k \leq 11$ .

Efforts to produce smaller designs benefit from starting with a “good” initial design (Tang, 1998). We consider a starting design for an  $8 \times 3$  RLH, which violates our rule of thumb;  $n = 8 < 50$  and  $k = 3 > \frac{8}{3} = 2.67$ . There are no known catalogued OLH or NOLH designs with these dimensions. We generate an FRLH design, which has a  $\rho_{map}$  equal to 0.0476, shown on the left side of Table 17. Although the FRLH meets Ang’s near-orthogonality criteria, we employ VMIN. We use the FRLH design as a heuristic solution to initiate VMIN. VMIN optimizes the third column resulting in a fully orthogonal design,  $O_3^8$ :  $\rho_{map} = 0$  and a condition number of 1.0. We add the final design on the right-hand side of Table 17 to the OLH library.

Table 17. Results from a small design after iterative application of Florian’s method on the left-hand is not an OLH design. It can be used as the initial design for VMIN. The final design is orthogonal, as shown on the right-hand design.

<b>Initial FRLH</b>			<b>Final Orthogonal</b>		
<b>Design: <math>\rho_{map} = 0.0476</math></b>			<b>Design: <math>\rho_{map} = 0.0</math></b>		
<b>k1</b>	<b>k2</b>	<b>k3</b>	<b>k1</b>	<b>k2</b>	<b>k3</b>
3	8	7	3	8	2
1	4	6	1	4	4
6	7	3	6	7	6
4	2	2	4	2	1
8	6	5	8	6	5
5	1	8	5	1	7
7	3	4	7	3	3
2	5	1	2	5	8

We summarize our methodology to construct new designs in the following steps and present it graphically in Figure 9.

**Step 1:** Consider the dimensions,  $n$  and  $k$ , for the desired design.

**Step 2:** Determine if OLH or NOLH designs exist for desired dimension:

If OLHs or NOLHs with the same design dimensions are available, use them. As an alternative, determine if it is possible to construct a new design based on previous methods.

If the designs do not exist in the library, or design dimensions do not meet OLH and NOLH conventions, go to Step 3.

**Step 3:** Determine if it is necessary to apply the combined technique or only Florian's method.

If  $n > 50$  and  $k < \frac{n}{3}$ , generate one or more RLHs and iteratively apply Florian's method.

Upon termination, the resulting design will likely possess NOLH properties—if not, try a few more. If unsuccessful after a few iterations, go to Step 6.

If  $n > 50$  and  $\frac{n}{3} \leq k < n$ , or  $n \leq 50$ , go to Step 4.

**Step 4:** Generate up to 1,000 RLHs and select the design with the minimum  $\rho_{map}$  and designate it  $X_{initial}$ .

**Step 5:** Iteratively apply Florian's method on  $X_{initial}$  until  $\rho_{map}$  does not improve. Designate this improved design as  $X_H$ ; the result of a heuristic approach.

**Step 6:** With  $X_H$  as a start point, implement VMIN iteratively.

**Step 7:** Record the resulting design. If the design is unsatisfactory, return to Step 4 and repeat with new RLHs.

A departure from previous techniques, we find that our methodology usually uncovers more than one design from which to choose. Time permitting, an experimenter can generate a number of candidate NOLHs, and then use criteria other than  $\rho_{map}$ , such as space-filling properties, to select the NOLH that best meets the experiment's requirements. Some methodologies are deterministic in their final designs, such as Steinberg and Lin (2006) and Ang (2006), leaving the experimenter with only a single choice at their limits. The opportunity to quickly produce different NOLH designs, with few dimensional constraints, is an important strength of our method. Figure 9 outlines the steps to achieve these designs.



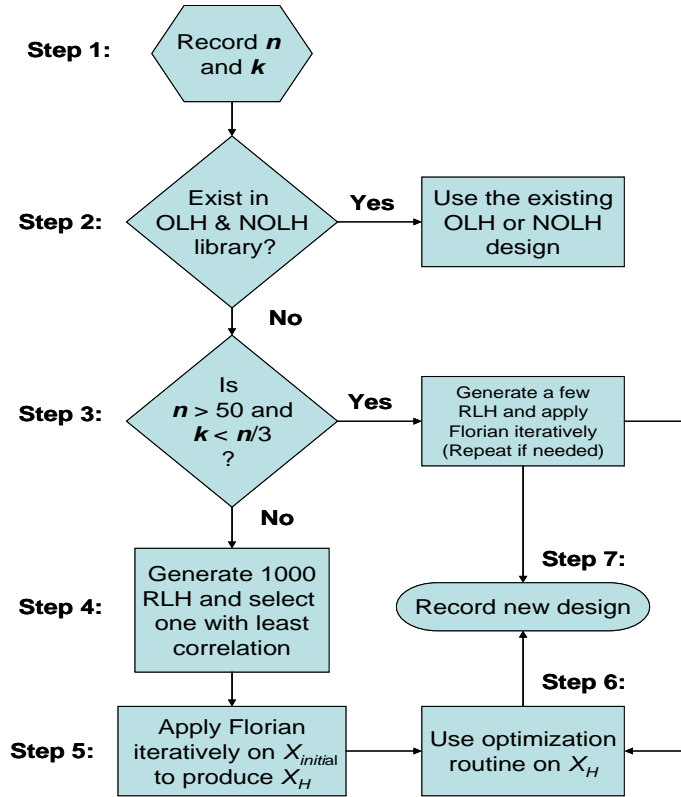


Figure 9. This flow chart presents the combined process for matrix transformation and optimization of an MIP to construct an OLH or NOLH design.

We demonstrate our methodology with a design for 14 runs and 7 factors. The resulting designs are in Table 18.

**Step 1:** The desired design has dimensions  $n = 14$  and  $k = 7$ .

**Step 2:** These dimensions do not conform to OLH or NOLH convention, which prompts the experimenter to go to Step 3.

**Step 3:** Because  $n = 14 < 50$  and  $k = 7 \Rightarrow \frac{k}{n} = \frac{7}{14} = 0.50 > 0.33$ , Florian's method alone will not likely result in an NOLH design. Go to Step 4.

**Step 4:** Using scripts that we developed in R software, we generate 1,000 RLHs and select the design with the minimum  $\rho_{map}$  and designate it  $X_{initial}$ . The resulting design has  $\rho_{map}$  equal to 0.367 and condition number 4.489.

**Step 5:** Iteratively applying Florian's method on  $X_{initial}$  results in a design with a  $\rho_{map}$  of 0.099. We designate this heuristic as  $X_H$  and present it on the left-hand side of Table 18.

**Step 6:** Iterative implementation of VMIN on  $X_H$  results in an NOLH design with  $\rho_{map}$  of 0.033 and a condition number of 1.134. We show the final design on the right-hand side of Table 18.

**Step 7:** We record this NOLH,  $N_7^{14}$ , as an entry into the NOLH library.

Table 18. The result from iterative application of Florian’s method on a randomly generated  $n = 14, k = 7$  design is the initial design for VMIN, as shown on the left-hand side of the table. It does not meet NOLH orthogonality criteria. The final design for  $n = 14, k = 7$  after VMIN is a new NOLH design, presented on the right-hand side.

Initial FRLH Design: $\rho_{map} = 0.099, \text{condition}(X'X) = 1.46$							Final Nearly Orthogonal Design: $\rho_{map} = 0.033, \text{condition}(X'X) = 1.13$						
K1	K2	K3	K4	K5	K6	K7	K1	K2	K3	K4	K5	K6	K7
1	8	1	10	9	13	13	8	4	5	3	9	13	13
6	9	13	13	14	8	6	7	5	1	9	14	8	6
8	12	3	7	3	7	2	14	3	9	7	3	7	2
4	14	10	11	1	5	8	4	7	3	12	1	5	8
13	4	2	4	11	2	10	3	1	11	14	11	2	10
10	7	14	2	8	11	7	10	2	14	4	8	11	7
7	1	9	14	2	3	9	13	8	4	6	2	3	9
5	5	12	5	5	14	5	2	6	8	10	5	14	5
12	6	7	9	10	4	3	5	9	2	1	10	4	3
9	10	5	12	13	12	4	11	12	6	11	13	12	4
11	2	8	8	6	10	14	12	13	7	13	6	10	14
3	3	4	1	7	6	1	6	14	13	8	7	6	1
14	13	6	6	4	9	11	1	11	10	2	4	9	11
2	11	11	3	12	1	12	9	10	12	5	12	1	12

We reiterate that this process is readily repeatable—using new random number streams to yield a new starting RLH—and develop many different NOLHs with our required dimensions, perhaps using other criteria, such as space-filling properties or consideration for higher order terms (interactions, quadratic, etc.) to select the best design.

An extension of this example demonstrates the power and flexibility of our combined methodologies. To develop  $N_{12}^{14}$  we use a new random stream to build a new  $n = 14, k = 10$  RLH design, and iteratively apply Florian’s method to produce a heuristic solution to initiate VMIN. An  $N_{10}^{14}$  design was possible with this method. Adding one

new column at a time from a random permutation of the value levels and applying VMIN, we construct the nearly saturated NOLH design,  $N_{12}^{14}$ , shown in Table 19.

Table 19. The  $N_{12}^{14}$  developed by RLH, FRLH, and VMIN emphasizes the flexibility of our methodology to construct any number of different NOLH designs (to include saturated ones). Note that none of the columns of this design are identical to the design on the right-hand side of Table 19. It shows that this new design is not merely an extension of the  $n = 14, k = 7$  design, although our technique can certainly extend smaller designs.

<b>Nearly Saturated NOLH Design:</b>											
$\rho_{map} = 0.046, \text{condition}(X^t X) = 1.33$											
<b>K1</b>	<b>K2</b>	<b>K3</b>	<b>K4</b>	<b>K5</b>	<b>K6</b>	<b>K7</b>	<b>K8</b>	<b>K9</b>	<b>K10</b>	<b>K11</b>	<b>K12</b>
3	6	1	6	14	5	5	6	10	6	14	6
1	7	9	10	5	14	2	8	8	14	4	4
12	8	3	14	4	7	13	2	11	13	8	11
9	5	4	3	3	2	3	12	6	9	2	14
10	1	7	1	10	8	14	9	7	12	6	2
11	12	10	13	12	1	6	11	9	7	3	1
6	3	14	11	11	11	12	10	4	4	7	13
4	11	11	7	7	6	9	14	12	11	13	12
2	4	5	12	9	4	7	4	3	5	5	8
14	2	12	8	6	9	1	5	13	3	11	7
7	10	13	4	2	3	8	1	2	8	12	5
5	13	8	2	8	10	11	3	14	2	1	9
8	9	2	9	1	12	10	13	5	1	10	3
13	14	6	5	13	13	4	7	1	10	9	10

Certainly another approach would be to use the initial  $n = 14, k = 7$  design from Table 18 and add one column at a time, but starting at this low number of variables is cumbersome. We chose to be aggressive and began with an  $k:n$  ratio equal to two-thirds, violating our “one-third” rule of thumb. The flexibility of saturated designs permits experimenters to take any subset of columns to develop a new design, when for example, more degrees of freedom for error in a linear fit is desirable. We complete our development of this design with a final application of Florian’s method to determine if an improvement is possible. There is no improvement in  $\rho_{map}$  and we enter  $N_{12}^{14}$  into the NOLH catalogue.

Our method requires relatively little time and resources. We construct the design using R software and GAMS. Generating 1,000 RLHs in R requires only seconds for a  $14 \times 10$  design matrix. Applying Florian’s method iteratively takes a similar amount of time. In practice, generation of 1,000 small RLHs, selection, and administration of Florian’s method are immediately sequential and involves less than two minutes. The GAMS program requires the most time, but current applications for small designs show that it is not uncommon for VMIN to reach completion within five minutes.

We use the largest of the “small” catalogued OLH and NOLH designs— $n = 33$  and  $k = 11$ —to illustrate the minimal resource investment for our methodology. Generating 1,000 RLHs, we create an FRLH with  $\rho_{map}$  equal to 0.0307 and a condition number of 1.156, which meets our NOLH criteria. We complete this process in less than 30 seconds on a standard desktop processor. Applying VMIN is not necessary to meet NOLH criteria for a  $33 \times 11$  design matrix.

As a test case, we develop a saturated design that does not adhere to OLH or NOLH dimensional constraints:  $n = 33$  and  $k = 32$  (see Appendix A). Approaches from Ang (2006), Cioppa (2002), and Ye (1998) do not develop designs to explore 32 variables in so few runs. Ang (2006) requires 65 runs to address 32 variables. Cioppa (2002) requires a sample size of 513, while Ye (1998) requires an increase in the number of runs to 131,073, for  $m = 17$ . Steinberg and Lin (2006) recognize that their methodology faces limitations in design dimensions. A comparison of the methods to construct a design to explore 32 factors is in Table 20. The comparison shows the advantages of our new technique over these other construction methods.

Table 20. A comparison of different approaches to develop a design that can explore 32 variables shows that our new S-NOLH design requires the least number of runs, and is therefore most efficient. Other approaches can theoretically develop experimental designs to explore 32 or more variables, but to the author's knowledge, none are currently in the catalogue of OLH and NOLH designs.

$k = 32$	<b>Ye</b>	<b>Cioppa</b>	<b>Ang</b>	<b>Steinberg and Lin</b>	<b>New</b>
Required $n$	131,073	513	65	64	<b>33</b>
$\rho_{map}$	0	0	0	0	<b>0.0435</b>

In the process of developing an  $N_{32}^{33}$  design, we also generate three new, individually unique designs for the same dimensions, each meeting orthogonality criteria. The best design has  $\rho_{map} = 0.0435$ . Two other designs have  $\rho_{map}$  values of 0.0465 and 0.0451. There are nine other new and different designs that do not meet NOLH criteria; they possess values of  $\rho_{map}$  between 0.05 and 0.06. Although other techniques can theoretically develop a design to explore 32 or more variables, such efforts (e.g., Ang, 2006 and Steinberg & Lin, 2006) result in exactly one such design.

Our recent discussion traces known design dimensions, but the flexibility of our technique results in previously unknown OLH and NOLH designs. We begin with Cioppa's  $O_7^{17}$  design as a start point. Adding a new column consisting of an ordered vector of 1, 2, ..., 17, we apply VMIN. The result is a new orthogonal design,  $O_8^{17}$ . A repetition of this process produces another completely orthogonal design,  $O_9^{17}$ . The author knows of no other method that produces an OLH of this dimension. Our extension of this process does not produce completely orthogonal designs, but meets nearly orthogonality criteria for an  $N_{13}^{17}$  design. Beyond 13 factors, we apply our full methodology to produce an  $N_{14}^{17}$  design with  $\rho_{map} = 0.0466$ . We also initiate a new FRLH, discarding Cioppa's  $O_7^{17}$ , and create a fully saturated NOLH,  $N_{16}^{17}$ , with  $\rho_{map} = 0.0491$  as shown in Table 21. Although we do not use Cioppa's original  $O_7^{17}$  as a basis, the  $N_{16}^{17}$  design still retains good properties, such as space filling (see Appendix B).

Table 21. This is a new design from an original FRLH for 17 runs. We extend the number of factors to explore until it is fully saturated. It allows an experimenter to explore 16 factors within 17 runs. The  $k:n$  ratio of 0.94 is large and makes the design a good screening plan.

	k1	k2	k3	k4	k5	k6	k7	k8	k9	k10	k11	k12	k13	k14	k15	k16
n1	5	9	13	7	15	1	15	6	5	2	7	11	1	9	14	3
n2	13	14	1	16	12	5	6	3	11	5	6	1	6	7	3	13
n3	14	2	6	9	9	2	16	8	9	17	11	17	10	3	4	12
n4	12	13	16	10	11	17	9	1	17	15	12	13	2	13	8	7
n5	4	15	11	15	6	12	7	16	6	6	15	14	4	1	10	15
n6	9	4	15	17	14	8	13	15	10	7	17	3	15	14	6	9
n7	10	7	14	3	10	4	4	17	13	14	3	4	3	11	11	17
n8	8	12	7	1	13	13	14	4	2	12	14	2	12	5	13	14
n9	17	6	8	2	3	15	11	13	7	1	10	7	5	10	2	5
n10	1	3	2	6	17	14	3	10	8	9	9	15	9	15	5	11
n11	2	17	3	8	1	6	17	11	12	11	8	9	11	17	7	8
n12	16	11	4	12	7	7	2	12	3	16	13	8	7	12	16	1
n13	15	10	12	13	8	11	10	7	4	4	1	16	16	16	15	16
n14	6	1	5	14	5	16	12	9	15	10	4	5	8	4	17	6
n15	3	8	17	11	4	10	5	5	1	13	2	6	13	6	1	4
n16	7	5	10	4	2	3	1	2	14	3	16	10	14	8	12	10
n17	11	16	9	5	16	9	8	14	16	8	5	12	17	2	9	2

We complete our examples with a design that does not follow current size conventions. We seek a design that explores 20 factors within 25 runs. The  $k:n$  ratio is 0.80. There are no similar designs to use as starting templates. Neither  $n$  nor  $k$  is a factor of two. After initiating a new FRLH, we produce a new NOLH design,  $N_{20}^{25}$ ,  $\rho_{map} = 0.0439$  within 10 minutes using VMIN. After iteratively adding columns we produce a final S-NOLH ( $N_{24}^{25}$ ), shown in Appendix A. An alternative for extending an  $N_{23}^{24}$  to become an  $N_{24}^{25}$  is to add a new row (for the new level 24<sup>th</sup>) to  $N_{23}^{24}$  and apply Florian’s method and then VMIN to produce an NOLH. Taking the new NOLH (now  $N_{23}^{25}$ ) we add a new column and again apply Florian’s method and VMIN, respectively. The result is a new saturated NOLH for  $N_{24}^{25}$ . The flexibility of this alternate process is an important attribute of this method.

Table 22 compares a few design dimensions from different construction methodologies. It is evident that our new method of combining FRLH and mixed integer programming (FRLH-MIP) has the flexibility to create unique designs that cater to the

experimenter’s needs. As previously mentioned, Steinberg and Lin (2006) require 64 runs to explore 32 factors, while FRLH-MIP uses 33 runs (Appendix A). This short discussion encompasses three unique design combinations for each of the given techniques. The new designs are saturated and further fill the OLH and NOLH library.

Table 22. A summary of techniques from Cioppa, Ang, Steinberg and Lin, and our new method shows that the FRLH-MIP method is a viable option to extend the library of OLH and NOLH designs. A label of “NA” designates dimensions not currently available from the technique.

	Cioppa	Ang	Steinberg and Lin	New
$n$	17	17	16	17
$k$	7	8	12	16
$\rho_{map}$	0	0	0	0.0490
$n$	33	33	NA	33
$k$	11	16	NA	32
$\rho_{map}$	0.0234	0	NA	0.0451
$n$	65	65	64	64
$k$	16	32	56	63
$\rho_{map}$	0.0219	0	0	0.0443

## E. SUMMARY

Combining FRLHs with an optimization routine to construct new designs provides the freedom to create the variety of orthogonal and nearly orthogonal design dimensions that we have discussed. The capability to produce many different NOLH designs with the same dimensions introduces great possibilities in stacking and crossed designs. With a choice of different designs, possessing acceptable nonorthogonality traits, scientists can use other measures to discriminate among them. The  $N_{30}^{33}$ ,  $N_{31}^{33}$ , and  $N_{32}^{33}$  designs in Appendix A are examples of this flexibility. Examining each design shows that they are not dependent on each other (i.e., the  $N_{31}^{33}$  design is not just a one-column extension of  $N_{30}^{33}$ ). Eliminating one column from  $N_{31}^{33}$ , or two columns from  $N_{32}^{33}$ , creates three different  $N_{30}^{33}$  designs. In comparison with the effort that other

methods require to produce one design, the undertaking for our technique is relatively trivial. Coupled with its flexibility in design dimensions, our approach provides scientists with a new tool to fill much of the OLH and NOLH library.



THIS PAGE INTENTIONALLY LEFT BLANK

## V. MANAGING MIXED-FACTOR, MIXED-LEVEL EXPERIMENTS WITH A NEW DESIGN METHODOLOGY

In accordance with the LHS construct (McKay et al., 1979), OLH and NOLH designs assume that all variables are continuous. Frequently this assumption is false. Many experiments include a number of discrete variables that do not necessarily have the same number of value levels. Converting (by rounding) the actual values of these discrete variables onto a raw OLH or NOLH design, especially when there are a small number of design points, often results in an overall design with poor orthogonality properties. We designate these types of designs as mixed-factor, mixed-level (MFML) experiments.

MFML experiments diminish the advantages of OLH and NOLH designs. In this chapter, we explore means to mitigate these disadvantages and restore near orthogonal properties. The foundation for creating a nearly orthogonal MFML design is the set of continuous variables from which we produce an NOLH using our new techniques. Our study exploits the dimensional flexibility of our new designs, by combining them with stacking methods and intelligent application of proven design techniques. The resulting MFML designs retain much of the orthogonality properties of the basic NOLH, thereby maintaining their utility. In practice, this approach has worked well for creating an efficient design for a problematic set of factors.

### A. THE BASE CONTINUOUS DESIGN

The set of continuous variables in the experiment determines the *base design* and is the foundation of the MFML methodology. Integer variables with a large number of value levels may also be part of the base design. The complete set of variables in the base design provides the flexibility in our methodology. Because the number of runs for this subset of variables can usually be set to any number  $n$ , the base design can match the design for the set of discrete variables as needed.

Previous chapters describe the ease of generating an OLH or NOLH from a set of continuous variables with a confluence of one or more techniques. The lone “hard” constraint with this methodology is that  $n > k$ . However, some analysts may choose to

avoid saturated designs. Space-filling characteristics may be an important feature for the experiment. The analyst may also decide that a design where  $\frac{k}{n} \leq 0.33$  is more appropriate, or desire a greater number of degrees of freedom. No matter the method for selecting  $n$ , it presents an initial restriction. We designate this initial restriction in the number of runs as  $n_I$ , where  $n_I > k$ , and use it to guide constructing a separate design for the set of discrete variables.

## **B. DISCRETE INTEGER VARIABLES AND STACKING METHODOLOGY**

The set of discrete variables presents difficulties for creating a LH design that possesses an acceptable  $\rho_{map}$ . Often, discrete integer variables do not have the same number of value levels. In such cases where sample size is a constraint, there may be little chance for creating a complete balanced design for them (Box et al., 1978; Satterthwaite, 1959). We first consider cases when there is a single discrete integer variable mixed with an experiment that predominantly contains continuous variables. Our method also handles subsets of discrete variables, each with differing numbers of value levels. A part of this process uses stacking methods—i.e., permuting the columns and appending additional rows or another design (Cioppa & Lucas, 2007). The full methodology includes stacking two or more designs, adding columns for binary factors through random permutations of  $\{0, 1\}$ , and crossing quantitative designs with designs for qualitative variables. The goal for systematically constructing an MFML experiment is to create a nearly orthogonal random balanced design (Satterthwaite, 1959).

### **1. One Subset of Discrete Variables with a Common Value Level**

We examine the set of discrete variables and identify the value level, which we denote as  $\ell_i$ , for each variable,  $i = 1, 2, \dots, w$ . If it is reasonable to use a common value level  $\ell$  (i.e., all discrete variables have the same number of value levels), it is much easier to generate a total MFML design with nearly orthogonal properties. Given that  $\ell$  is greater than the number of discrete variables,  $d$ , we apply similar techniques for creating an NOLH from continuous variables. From empirical study, cases in which

$\ell \geq 10 \geq d$  show that our methods often result in a design with an acceptable  $\rho_{map}$ . For cases in which  $\ell < d$ , we split the set of discrete variables into smaller subsets such that  $\ell > d$  is true for each new subset, permitting the application of previous techniques to create an NOLH. In the next section, a small example demonstrates the significance of establishing these constraints on  $\ell$ .

## 2. Addressing More Than One Subset of Discrete Variables

For experiments containing several discrete variables with varying numbers of value levels, we group the discrete variables into subsets with corresponding common  $\ell$ 's. Subject matter expertise and the analyst's reasoning can simplify the number of subsets with which to work. The concept for developing a complete design with these disparate subsets of discrete variables is to incorporate each subset design as an LH into the overall design. Figure 10 illustrates this idea later in the section.

We identify  $w$  subsets of discrete variables and designate the discrete subset designs as  $DV_1, DV_2, \dots, DV_w$ . Each subset has a corresponding number of value levels,  $\ell_1, \ell_2, \dots, \ell_w$ , from which we determine the least common multiple (LCM) and designate it as  $n_{DV}$ , which is the minimum total number of runs for all discrete variables in the design. For each  $\ell_i$ , there exists an integer value  $b_i$ , such that  $\ell_i \bullet b_i = n_{DV}$ ,  $i = 1, 2, \dots, w$ , and determines the **initial** number of stacks for each  $DV_i$ . We adjust the number of stacks to match  $n_i$  if  $n_i < n_{DV}$ , or increase  $n_i$  if  $n_i > n_{DV}$ .

The intent of creating a balanced design LH (Box et al., 1978; Satterthwaite, 1959) drives our choice for grouping discrete variables. We contemplate  $\ell_i$  to determine whether it is possible to incorporate a smaller  $DV$  within a larger  $DV$ . For instance, we consider three sets of discrete variables with value levels 5, 8, and 10, thus  $n_{DV} = LCM = 40$ . We denote each design as  $DV_1, DV_2, DV_3$ , respectively. A savvy analyst could reasonably, if resources permit, decide to increase the number of value levels for  $DV_2$  from 8 to 10, thereby creating a larger group ( $DV_3$ ) based on 10 value levels. This simplification makes a balanced design easier to obtain for a reasonable  $n$ .

The experimenter matches two stacks of  $DV_1$  within every one  $DV_3$ . We update LCM:  $n_{DV} = LCM = 10$ .

To further clarify the MFML process, we establish that the base design of continuous variables requires  $n = 40$  design points, based on the original LCM. We determine the final number of stacks for  $DV_1$  and  $DV_3$  based on  $n$  to be 8 and 4, respectively. The four stacks of  $DV_3$  act as the *base stack* for the eight stacks of  $DV_1$ . A short program in R performs iterative random stacking of  $DV_1$ . For each iteration of stacking, we append the new stack to the stacks of  $DV_3$  and compute  $\rho_{map}$ . The experimenter can limit the number of iterations to a number of trials deemed reasonable or a threshold for an acceptable  $\rho_{map}$ .

A final generation of a base design from the remaining continuous variables results in an MFML design with the structure shown in Figure 10. We note that the flexibility of our new NOLH designs allows the base design to adjust to the discrete variable designs. For clarification, we rename  $DV_1$  as  $DV_5$  and  $DV_3$  as  $DV_{10}$  in the illustration.

DV <sub>5</sub>	DV <sub>10</sub>	Base
DV <sub>5</sub>		
DV <sub>5</sub>	DV <sub>10</sub>	
DV <sub>5</sub>		
DV <sub>5</sub>	DV <sub>10</sub>	
DV <sub>5</sub>		
DV <sub>5</sub>	DV <sub>10</sub>	
DV <sub>5</sub>		

Figure 10. The structure of the stacked MFML design.

### 3. Formalized Stacking Methodologies

In this section, we discuss our stacking approach as part of the MFML methodology. Whereas stacking is sometimes applied to increase  $n$ , we use stacking as

an opportunity to improve  $\rho_{map}$  for the subsets and overall set of discrete integer variables. A stacked design in which each design is based on randomly permuted columns of the original design can improve the  $\rho_{map}$  of the complete design, such that  $(\rho_{map})_{stacked} \leq (\rho_{map})_{original}$  (Cioppa & Lucas, 2007). This method of stacking helps efforts to improve the overall design that includes the base design.

Stacking also provides the freedom to mitigate assumptions that prove false. Our discussions in the previous two sections assume  $\ell_i > d_i$ ,  $i=1,2,\dots,w$ , where  $d_i$  is the number of variables in the  $i^{th}$  subset of discrete variables. Another assumption is that  $n_{DV} > k_{DV}$ , where  $k_{DV}$  is the total number of discrete variables,  $k_{DV} = d_1 + d_2 + \dots + d_w$ , and that  $n_{DV} > k$ . When these assumptions do not hold, we use stacking methods.

To apply stacking methods, we first establish the final number of runs for the experiment,  $n$ , by comparing  $n_{DV}$  and  $n_I$ : (1) If  $n_{DV} \geq n_I$ , then  $n = n_{DV}$ ; (2) If  $n_{DV} < n_I$ , then we take the ceiling<sup>9</sup> of the ratio of the  $n_I$  and  $n_{DV}$  as a modification factor, such that

$n = \left\lceil \frac{n_I}{n_{DV}} \right\rceil \times n_{DV}$ . The number of stacks for each  $DV_i$  in the overall MFML design is

$s_i = b_i \times \left\lceil \frac{n_I}{n_{DV}} \right\rceil$ , recalling that  $b_i$  is the initial number of stacks for each subset of discrete

variables. We adjust  $b_i$  based on the modification factor,  $\left\lceil \frac{n_I}{n_{DV}} \right\rceil$ , if this is possible.

We leverage the new method for stacking from Cioppa & Lucas (2007) to reduce the nonorthogonality of the overall stacked design. We generate a  $DV$  design with a reasonable  $\rho_{map}$  and permute the columns to create a “new” design—once for each required number of stacks,  $s$ , for the design. Each rearranged LH is not truly a new design since there is a one-for-one column match in the new and original design, albeit in different column order. However, the overall stacked design is different and an improvement in  $\rho_{map}$  is evident (Cioppa & Lucas, 2007). We may permute the  $k$

---

<sup>9</sup> The *ceiling* of the value  $x$  is the smallest integer value that is greater than or equal to  $x$ .

columns in any of  $k!$  ways and choose  $s$  of them to create an overall stacked design. Put another way, we can choose from any of  $\binom{k!}{s}$  stacked designs. We select the set of new designs based on a desire to reduce the  $\rho_{map}$  of the stacked design. Another simple program in R allows a search for a set of “new” designs that will stack into an overall discrete integer design with an acceptable  $\rho_{map}$ . After selecting the set of “new” designs to stack, we see that there are  $s!$  ways of stacking them. Each restacked design from the same set of “new” designs results in the same overall  $\rho_{map}$ . These findings provide a flexibility to reduce nonorthogonality inherent in a single subset of discrete variables in MFML experiments.

Laterally appending a stacked design for one subset of discrete variables to another subset’s discrete variables design may increase  $\rho_{map}$  for the *joined design*,  $DV_{Joined}$ . To minimize  $\rho_{map}$  for the joined design, we hold one set of stacked designs as the *base stack*,  $DV_{Base}$ , while randomly restacking the subset of *interest*,  $DV_{Interest}$ . During random restacking,  $DV_{Interest}$  maintains its  $\rho_{map}$ , but its correlation with  $DV_{Base}$  changes, thereby changing  $\rho_{map}$  for  $DV_{Joined}$ . We continue restacking  $DV_{Interest}$  until  $\rho_{map}$  for  $DV_{Joined}$  is acceptable or ceases to improve. Since  $\rho_{map}$  for the joined design will not be better than the worse  $\rho_{map}$  of the two *DVs*, it is reasonable to set  $\rho_{map}$  from the worse of  $DV_{Base}$  and  $DV_{Interest}$  as a threshold. The joined design is the new *base stack*. We incorporate a new  $DV_{Interest}$  in the same manner until all discrete variable subsets are in the base stack.

In the case of a single integer variable, a simplistic approach reaps significant benefits. We randomly permute a vector of the possible values of the variable; the vector consists of each value repeated exactly the same number of times that there would have been stacks. For instance, a single variable with  $\ell = 4$  levels and  $n = 20$ , results in a vector consisting of the values 1, 2, 3, and 4, each repeated  $\frac{n}{\ell} = \frac{20}{4} = 5$  times,

$\underline{x} = (1, 1, 1, 1, \dots, 4, 4, 4, 4, 4)$ . The analyst generates a new random permutation of the vector, appends it to  $DV_{Base}$ , measures  $\rho_{map}$ , and compares it with a specified correlation criterion. A maximum number of iterations may also be a criterion. A modification of the R program that handles restacking helps in this process. We find the single integer variable approach useful when there are few value levels ( $\ell \leq 10$ ) and applicable when discrete variables cannot be grouped.

Our overall design strategy includes known design techniques. Centered designs and two-level factorial designs are often part of MFML experiments. Additionally, a library of orthogonal arrays (Sloane, 2006) offers a number of catalogued orthogonal arrays (OAs) that may match the requirements for an MFML experiment. For readers interested in more details about OAs, Hedayat et al. (1999) have a comprehensive discussion of theory and applications. A group of these OAs already contains designs with factors that have varying numbers of value levels. For instance, the library lists an MA.12.2.4.3.1 design. The nomenclature represents a mixed orthogonal array (MA) that has 12 runs, 4 two-level factors, and 1 three-level factor. These designs are very specific to the types and mixture of factors they can address.

Many cases may occur for subsets of discrete variables. We do not attempt to enumerate them and leave it to the analyst to best simplify the experiment, while using our guidance for combining design methods. The goal for this methodology is to develop a design that can explore the main effects of each factor.

Once the final *base stack* design is complete, the experimenter can apply our new techniques or other known methods to create the *base design* from the set of continuous variables. The total number of runs is still  $n$ . The number of variables for the base design is  $k_{CV} = k - k_{DV}$ . We expect to benefit from the flexibility of our new designs if known OLH and NOLH designs cannot accommodate  $k_{CV}$ .

### C. COMBINED AND STACKING METHODOLOGIES

This section describes our procedure for logically combining stacking methodologies and proven design schemes to produce designs with acceptable nonorthogonality measures. Because there are many possible types of variables and



different numbers of value levels that can occur, we discuss our general methodology for a fairly complex case. A complex case involves a design that has continuous variables, discrete variables—each set with a different number of value levels and/or single DV that has a number of value levels that matches neither of the DVs’ sets. We include a flow chart as a template for the MFML methodology. Description of a student thesis that employed this process illustrates its utility.

### 1. Steps for Creating a Mixed Factor, Mixed Level Design

This section lists the general steps to guide the creation of an MFML design with acceptable nonorthogonality. In their present form, these steps are helpful, but require more specificity. Experience and reason are required to determine how to best employ the methodology for each new case.

**Step 1:** Determine the sets of discrete and continuous variables. Set  $n_I$  to meet requirements for all variables, such that  $n_I \geq k + 1$ .

**Step 2:** Group the set of DVs into subsets that have common numbers of value levels ( $\ell_i$ ). Simplify subsets and  $\ell_i$  as logic dictates.

**Step 3:** Split subsets of DVs if  $\ell_i < d_i$  in the subset.

**Step 4:** Determine  $LCM = n_{DV}$  from the set of  $\ell_i, i = 1, \dots, w$ . Determine  $b_i, \forall i$ .

**Step 5:** Adjust the total number of runs to be  $n = \left\lceil \frac{n_I}{n_{DV}} \right\rceil \times n_{DV}$ .

**Step 6:** Use new and past techniques to create the base design with an acceptable  $\rho_{map}$ .

**Step 7:** Determine the number of stacks required for each set of DVs:  $s_i = \frac{n}{\ell_i}, \forall i$  or

$$s_i = b_i \times \left\lceil \frac{n_I}{n_{DV}} \right\rceil, \forall i.$$

**Step 8:** Create a stacked design for each set of DVs, restack, and append until the overall design meets the threshold for  $\rho_{map}$ .

It is common for  $\rho_{map}$  of the stacked DVs to dictate  $\rho_{map}$  for the overall design. At times, the experimenter may wish to stack the base design.

## **2. Design Best Practices within the Mixed Factor, Mixed Level Process**

Intertwining proven design schemes with the MFML methodology is integral to our overall process. Full and fractional factorial designs, centered designs, and simple LHs are a few of the proven techniques that also play a role in creating a full MFML design. Although we have concentrated on numeric variables, many experiments include qualitative variables. Binary and categorical variables are often essential to the scientific method.

Recall that a major objective in the MFML process is for a balanced design. Understanding that nonnumeric variables play no part in computing correlation, it is still important for our technique to incorporate these variables in the MFML. We can cross orthogonal designs derived from our techniques with other designs. The flexibility that we offer in design dimensions eases the ability to match the dimensions of a balanced nonnumeric design. It is not uncommon to have five binary variables in an experiment. A full factorial design requires  $2^5 = 32$  runs. In the absence of a catalogued orthogonal (or nearly orthogonal) design, we can offer a 32-run design from our technique with the set of numeric variables. We can also incorporate larger resolution III or resolution V fractional factorial designs, such as those in Sanchez & Sanchez (2005).

### **D. EXPLORING FIRST RESPONSE TO A BOMB ATTACK USING A MIXED-FACTOR, MIXED-LEVEL DESIGN**

This section demonstrates the utility of the MFML. We discuss the general methodology in a fairly complex case involving a graduate student's thesis. The study is ideal for illustration because it explores a mixture of qualitative and quantitative variables, which include integer and continuous variables.

#### **1. The Thesis Problem**

Major Jon Roginski's thesis (2006), "Simulating Emergency Response Using Multi-Agent Simulation," studies first response to a bomb attack in Baltimore's Inner Harbor area (see Figure 11). It leverages simulation techniques to determine the efficient use of resources for the Department of Homeland Security (DHS) within the National

Exercise Program. DHS's previous efforts to employ simulations have cost tens of millions of dollars, with an unsatisfactory return on investment (Roginski, 2006).

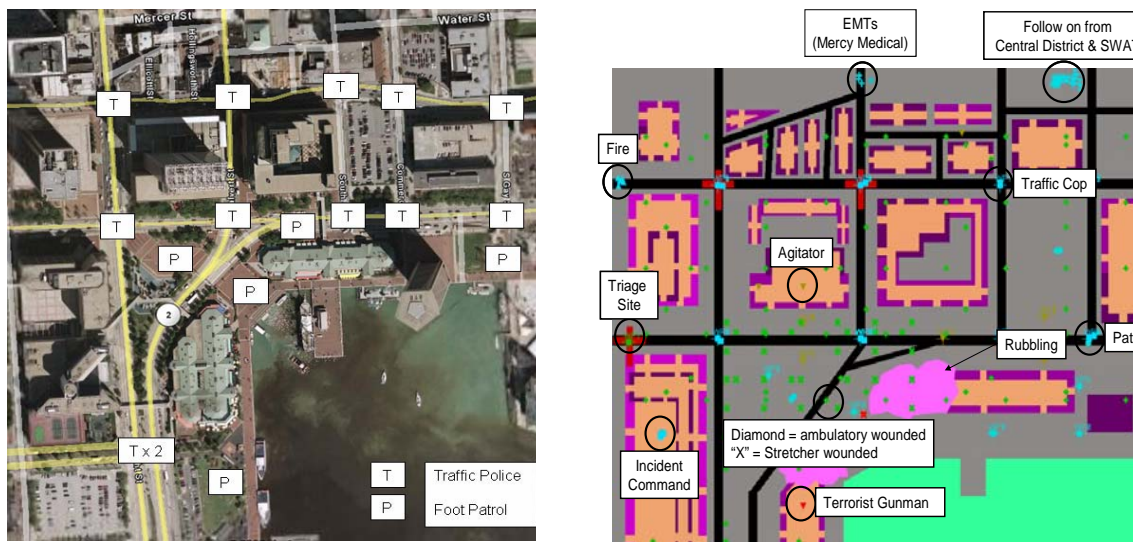


Figure 11. Baltimore harbor vignette terrain: actual photo and within multiagent simulation (Roginski, 2006).

Major Roginski adapts an existing organizational learning process and integrates low- and high-resolution simulation to provide decision support. His study presents the potential benefits of low-resolution simulation, *using efficient experimental design and high-performance computing*. He examines a 48-dimensional response surface. These factors are a mixture of integer and DVs, as well as three two-level variables and two three-level variables.

This is quite a design challenge. Without the use of an efficient design, Major Roginski calculates that a traditional gridded design for 48 factors and 30 replications requires  $1.60 \times 10^{24}$  CPU hours, or approximately 116 trillion times the age of the universe, even with the help of the Maui High Performance Computer Center (MHPCC). Using an efficient MFML design and the MHPCC, all experiments are completed in less than three weeks or 156 CPU centuries (Roginski, 2006).

## **2. Applying the Mixed-Factor, Mixed-Level Methodology to a Thesis Question**

Distilling this problem requires an examination of the variables in the experiment. The analyst considers 48 variables, three of which are color or pattern indicators in the simulation. These three variables are not part of the design. Five other variables are discrete: three have two-levels and two have three-levels. Sloane (2006) does offer some alternatives for this design, but the designs in the orthogonal array library are fairly restrictive in the number of runs and number of mixed factors they can handle.

Table 23 shows 40 of the variables that are the focus of design development. The table shows that many of the variables have only eight levels. Our previous discussions in this chapter state that variables with less than ten levels are problematic. A number of the variables have 144 or more levels, which may be considered as continuous. Lastly, four variables have 11 levels. Table 23 separates the different number of value levels.

Table 23. Forty of the variables in Major Roginski's experiment are displayed. The variables are all integers, but some have enough value levels to be considered as continuous, as the shaded rows indicate.

<b>Variable Name</b>	<b>Min Value</b>	<b>Max Value</b>	<b># Value Levels</b>	<b>Type</b>
Num Ag1	0	7	8	Integer
Num Ag2	0	7	8	Integer
Num Ag3	0	7	8	Integer
Num Ag4	0	7	8	Integer
Num Gun	0	7	8	Integer
Num EMT SJ	0	7	8	Integer
Num EMT MM	0	7	8	Integer
Num ESU1	0	7	8	Integer
Num ESU2	0	7	8	Integer
Num ESU3	0	7	8	Integer
Num ESU4	0	7	8	Integer
Num ESU5	0	7	8	Integer
Num Foll1	0	7	8	Integer
Num Foll2	0	7	8	Integer
Num Foll3	0	7	8	Integer
Num Traff1	0	7	8	Integer
Num Traff2	0	7	8	Integer
Num Traff3	0	7	8	Integer
Num Traff4	0	7	8	Integer
Num Traff5	0	7	8	Integer
Num Traff6	0	7	8	Integer
Num Traff7	0	7	8	Integer
Num Traff8	0	7	8	Integer
Num Traff9	0	7	8	Integer
Num Traff10	0	7	8	Integer
Num Traff11	0	7	8	Integer
Num Traff12	0	7	8	Integer
<b>Desire Civ</b>	<b>1</b>	<b>144</b>	<b>144</b>	<b>Integer~Continuous</b>
<b>Color</b>	<b>1</b>	<b>144</b>	<b>144</b>	<b>Integer~Continuous</b>
<b>Marksman Pol</b>	<b>1</b>	<b>144</b>	<b>144</b>	<b>Integer~Continuous</b>
<b>Prob Com</b>	<b>1</b>	<b>144</b>	<b>144</b>	<b>Integer~Continuous</b>
<b>Vul Ag</b>	<b>1</b>	<b>144</b>	<b>144</b>	<b>Integer~Continuous</b>
<b>Vul Gun</b>	<b>1</b>	<b>144</b>	<b>144</b>	<b>Integer~Continuous</b>
<b>Marksman Ag</b>	<b>1</b>	<b>144</b>	<b>144</b>	<b>Integer~Continuous</b>
<b>Marksman Gun</b>	<b>1</b>	<b>144</b>	<b>144</b>	<b>Integer~Continuous</b>
<b>Num Civ</b>	<b>0</b>	<b>150</b>	<b>151</b>	<b>Integer~Continuous</b>
Num SWAT	5	15	11	Integer
Attr in Orders	0	10	11	Integer
Attr in Ag	0	10	11	Integer
Eff Bomb	0	10	11	Integer

There are a few approaches that the experiment may use as a modification of the steps listed previously. We acknowledge that, at the time of creating this design, VMIN had not yet been fully developed. Because 27 of the variables have only eight value levels, use of the single integer method is a useful scheme. Nine of the variables have 144 or more value levels. The experimenter may consider these variables to have a common  $\ell$  of 144. We easily develop an NOLH with previous techniques.

The remaining four variables have 11 value levels. Because  $\ell > d$  for this subset, creation of a  $4 \times 11$  NOLH design is possible, followed with a stacking approach. The overall design will require at least 144 design points. Therefore, it is possible to develop an FRLH design for all these numeric variables. A large number of design points mitigates round off issues and converts the actual values of DVs into a nearly orthogonal design matrix.

The decision to treat 40 of the integer variables as continuous variables, and develop a base design from them, involves the LCM in the two-level and three-level variables in the experiment. Three two-level variables require an eight-run design, which can be orthogonal. A full-factorial design for two three-level variables requires nine runs. The LCM for these two sets of DVs is 72. Because a number of the integer variables have 144 levels, two stacks of the combined DV sets matches the base design.

To illustrate, Table 24 has the eight design points of a full-factorial design for the two-level variables. They match with one design point of the two, three-level variables (diagonal shading). Since there are nine design points for the two, three-level variables, these eight design points for the two-level, full-factorial design is repeated for each unique design point, which results in 72 runs.

Table 24. Excerpt of the DV subset design for the MFML, containing 24 of the 144 total design points, shows how a full, two-level factorial design for three binary variables aligns with three design points of two discrete variables, each with three value levels. We shade sections of the table to emphasize the match between a single design point for the discrete, non-binary variables and a full two-level factorial for the binary variables.

ESU	FO Police	Traffic	Time OP	Panic
0	0	0	1	1
0	0	1	1	1
0	1	0	1	1
0	1	1	1	1
1	0	0	1	1
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1
0	0	0	1	2
0	0	1	1	2
0	1	0	1	2
0	1	1	1	2
1	0	0	1	2
1	0	1	1	2
1	1	0	1	2
1	1	1	1	2
0	0	0	1	3
0	0	1	1	3
0	1	0	1	3
0	1	1	1	3
1	0	0	1	3
1	0	1	1	3
1	1	0	1	3
1	1	1	1	3

Stacking this set of 72 design points atop one another results in 144 total design points, which correspond with the number of runs in the base design. Using an iterative application of Florian’s method with the remaining 40 variables (assuming all are continuous) and  $n = 144$  produces a base design with  $\rho_{map} = 0.00766$ . When the actual values of the integer variables treated as continuous are converted (by rounding) into the design, the resulting nonorthogonality measure is  $\rho_{map} = 0.0593$ . Despite the many

variables with small numbers of value levels, the nonorthogonality measure remains relatively small. The complete  $n = 144$ ,  $k = 45$  design is in Appendix C.

Examination of this design reveals advantages of the MFML approach. An NOLH design using Cioppa's (2002) method to explore 45 factors requires  $m = 9$ , which corresponds with  $n = 1,025$  and  $k = 46$ . This is if all variables are considered continuous. If the five qualitative variables are separated, an NOLH design to examine the remaining 40 variables still requires 1,025 runs since  $m = 8$  only results in  $k = 37$  and  $n = 513$ . A conventional crossed design approach to combine the qualitative variables with the NOLH could lead to 73,800 runs. Furthermore, we are unaware of the existence of either a 40- or 45-variable NOLH design with so few design points. Meanwhile, our methods have developed an FRLH design that explores up to 172 variables within 193 runs ( $N_{172}^{193}$ ) and possesses a nonorthogonality measure of 0.0275. The MFML design reduces the number of runs in the NOLH by more than 85%, while preserving enough of the orthogonality properties of the base design to gain advantages in the analysis of the simulation output.

Major Roginski's study successfully examines all 48 factors using this design. His analysis of the data from the experimental design yields major findings, undiscovered from previous studies:

- The most important factor in achieving success in crisis mitigation is the effectiveness of the police.
- If a police force is not well trained, they may achieve greater success by being less persistent with individuals.
- Well established, well executed standard operating procedures (SOPs) may play a more important role in first response operations than interagency communication.
- There is a diminishing return after a certain level of first responder training. It may be more effective to leverage resources elsewhere after reaching that level.

The ability to examine factors in isolation using the MFML design contributes to the validity and significance of these findings (Roginski, 2006).



## E. SUMMARY

MFML experiments are difficult propositions for analysts. They are typically problematic, even when using orthogonal or nearly orthogonal designs. Conversion of actual values onto the raw OLH and NOLH designs often result in poor orthogonality properties, reducing the advantages of these efficient designs.

Our methodology produces a balanced MFML design that exhibits near orthogonal properties that benefit analysts. Using guidelines to partition and shape the subsets of DVs and stacking techniques, we apply proven design schemes to develop an overall balanced design. The base design consists of continuous variables, or integer variables with many value levels. The flexibility of our previous techniques permits manipulation of the base design to conform to the needs of set of DVs.

A new methodology for stacking experimental designs is an important feature of the overall process for creating MFML designs. Random column permutation of an original NOLH design does not create a new design, but randomly restacking these column-permuted designs generally results in a completely new overall design. Experience and communication with the customer, as well as knowledge about the experiment of interest, directs the analyst to logically apply this methodology. A flowchart or list of steps cannot fully capture this art.

A use-case demonstrates the utility of the MFML methodology. Major Jon Roginski's study (2006) involves 30 replications of a design consisting of 45 continuous and integer DVs with varying numbers of value levels, as well as simulation-specific factors. Roginski (2006) estimates that if a full factorial design were used, the required time to complete these experimental runs approximately equals 116 trillion times the age of the universe. Previous NOLH designs by Cioppa (2002) to explore just 45 factors require  $m = 9$ , which corresponds to  $n = 1,025$ , assuming all continuous variables. A conventional crossed design approach to combine the qualitative variables with the NOLH could lead to 73,800 runs. Our new methodology creates an MFML design with 144 runs to explore 45 variables. By using the computing power of the MHPCC along with the efficiency of the nearly orthogonal MFML design, Major Roginski's experiment requires less than three weeks or 156 CPU centuries (Roginski, 2006).

## **VI. THE COMPLETE FLEXIBLE OLH/NOLH METHODOLOGY: EXPLORING SOLDIER-LEVEL NETWORKS**

We implement our new designs to support the United States Army Training and Doctrine Command Analysis Center in a study of network-centric warfare. Our design helps assess the impact that Soldier-level, network-enabled capabilities have on a truck terminal's cargo operations as part of a Joint Force sustainment base. The research studies current joint distribution operations and the role that transportation operations play in the joint context. The system of interest is the operational concept of the Centralized Receiving and Shipping Point (CRSP) and its organization.

### **A. PROBLEM STATEMENT**

Major Francisco Baez, United States Army, (2008) examines up to 20 factors of mixed types with mixed numbers of value levels to answer the following research questions:

- Which network-enabled capability gaps exist in the execution of transportation Soldiers performing terminal cargo operations tasks, under the identified conditions, to the identified performance standards?
- Which distribution structures and types of network-enabled capabilities allow transportation Soldiers to accomplish their task to specified standards, under given conditions?
- Are the network-enabled capabilities currently available to individual transportation Soldiers?
- What is the operational impact of leaving the gap unfilled?

This study focuses on three dissimilar communications network topologies that include In-Transit Visibility (ITV), networked communications, and information systems that provide network-wide visibility of node and mode status in a shared Logistics Common Operating Picture (LCOP). Operational scenarios from subject matter experts will employ these networks in a simulation.

Factors come from concept specific attributes documents (JCS, 2005). The study considers ITV-availability, ITV-accuracy, LCOP-update rates, probability of

communications, latency, and communication relay capability as parameters that influence network capabilities. Noise factors such as resources available, convoys per hour, and convoy mix cases examine aspects of network-enabled capabilities on a broader basis.

The results from this effort can support or refute the findings from the functional needs analysis in the Joint Capabilities Integration and Development System. Recommendations from this research may also serve to shape future capabilities requirements. Most importantly, it will benefit Soldiers who operate from fixed-based facilities in theaters of operation.

## **B. DESIGN OF EXPERIMENTS AND SIMULATION RUNS**

This simulation study uses modeling and simulation and an efficient experimental design. Major Baez' study requires an MFML design for this computer experiment. His simulation includes 1 categorical variable, 5 binary variables, and 14 numeric variables (integer and continuous) with different numbers of value levels. Converting the actual values of discrete variables onto a raw NOLH, results in an overall design with poor orthogonality properties. We create a base design using the methods of Chapter V (see also Hernandez, 2008b) for one integer variable with 16 levels and 12 continuous variables, using a subset of the columns from the  $n = 16, k = 15$  saturated design shown in Table 25. We use an MFML design that consists of this new S-NOLH design, combining them with stacking methods, and intelligently applying proven design techniques.

Table 25. A saturated design for  $n = 16$  and  $k = 15$  is the foundation for creating a base design for an MFML design. Its correlation measure is 0.0471, with a condition number of 1.319.

	k1	k2	k3	k4	k5	k6	k7	k8	k9	k10	k11	k12	k13	k14	k15
n1	10	8	13	6	16	8	15	16	14	3	12	2	13	7	9
n2	13	12	4	8	13	15	3	4	9	4	3	3	4	2	13
n3	7	6	12	1	2	11	4	5	11	5	16	4	3	12	5
n4	9	16	6	4	12	5	8	14	5	6	10	15	2	15	12
n5	11	10	8	5	15	1	7	2	15	16	7	9	9	9	2
n6	14	4	1	10	3	7	12	6	16	7	11	14	12	10	16
n7	6	9	14	16	11	3	9	1	3	8	15	6	7	8	15
n8	1	13	16	7	5	10	5	7	12	9	1	10	16	11	14
n9	12	7	10	2	1	2	13	12	1	13	4	5	8	3	11
n10	16	14	11	13	6	12	2	13	6	11	14	13	14	4	4
n11	2	3	5	3	14	16	10	8	4	15	13	12	11	5	10
n12	3	2	9	11	9	4	6	11	10	1	5	16	6	1	3
n13	15	1	15	12	10	14	11	9	7	10	2	11	5	16	7
n14	5	5	2	14	8	6	1	15	8	12	8	1	10	14	8
n15	4	15	7	15	4	13	16	10	13	14	9	8	1	6	6
n16	8	11	3	9	7	9	14	3	2	2	6	7	15	13	1

We select any subset of 13 columns and stack it, building a 32-run by 13-factor base design to match the full factorial design ( $2^5 = 32$  runs) of five binary variables. Next, we cross the resulting design with the three-level categorical variable to create a 96-run design. Randomly assigning value levels for the three-level integer variable to match the overall design creates an NOLH. The final MFML design contains 96 runs to address 20 factors in one operational scenario (see Appendix D). It retains the raw design's nonorthogonality measure of  $\rho_{map} = 0.0471$ .

We remark that this design was not available when Major Baez did the production runs for his thesis. As an alternative design, Major Baez assigns values for the binary and categorical variables within the 257-design point design, since it can handle up to 29 variables (Sanchez, 2006). The complete number of design points to examine three different scenarios is 771. The experiment involves 10 replications for each scenario for each of the 257 design points, for a total of 7,710 computational experiments. This is far less than the computational effort that would be required for 10 replications of a 257-design point NOLH crossed with the full, two-level factorial design, but less efficient than the MFML design. We note that after rounding to the actual values, the overall

nonorthogonality measure from the catalogued NOLH design is  $\rho_{map} = 0.0962$ —more than twice that of the MFML, while still requiring nearly three times the number of runs.

Baez (2008) executes the MFML experiment with 10 replications and 96 design points (Hernandez et al., 2008) for each of the three communications scenarios, totaling 2,880 computational experiments to serve as a validation experiment.

### C. ANALYSIS

Analysis of the simulation output data was conducted primarily through the JMP Statistical Discovery Software (2004), a product of the Statistical Analysis Software institute. Major Baez chose JMP (2008) for its data visualization features that allow the user to interactively investigate data, and to refine and understand the analysis results in a dynamically-linked spreadsheet and graphical environment (Baez, 2008).

One analysis approach was construction partition trees, which we will use to illustrate the utility of our new designs. Partition trees recursively split the data from the simulation into homogeneous subsets in accordance with the relationship between the response variable and the predictors. Each split considers all possible cuts or groupings, given the current state of the tree to select a partition with the largest likelihood-ratio Chi-square statistic. A threshold of 5% change in  $R^2$  from the previous split determines when to split the data set (Baez, 2008). This method provides insights about the most significant factors influencing the response variables.

Major Baez (2008) focuses his study on comparing the three different network structures and identifying those factors that have the greatest impact on each structure's performance, as well as differences in overall performance for the three structures. An analysis of the mean time in the CRSP is only interpreted as the average time to process the specified number of convoys in the context of the scenario, and not necessarily the “typical” time for a convoy.

The following plots from the larger design are confirmed with our more efficient design. They demonstrate interesting results that confirm the complexity of the system and point to the need for further analysis.

The plots from JMP in Figure 12 show the utility of LH designs (Baez, 2008). The graphics present the time in the CRSP as the  $i^{\text{th}}$  convoy appears. They reveal interesting results from the different scenario settings. Design Points A and B plots show two scenarios with an arrival rate of one convoy per hour. Design Point A has low traffic intensity and Design Point B has somewhat high traffic intensity. The plot for Design Point A indicates that after a certain point the time in CRSP begins decreasing slightly, but steadily until the terminating event. The plot for Design Point B outlines a supply system that is unable to handle the process for this given combination of factor settings. The time in CRSP continues to increase, indicating that very large queues are building and each successive convoy takes longer to process than the last. All three networks exhibit very consistent behavior (Baez, 2008).

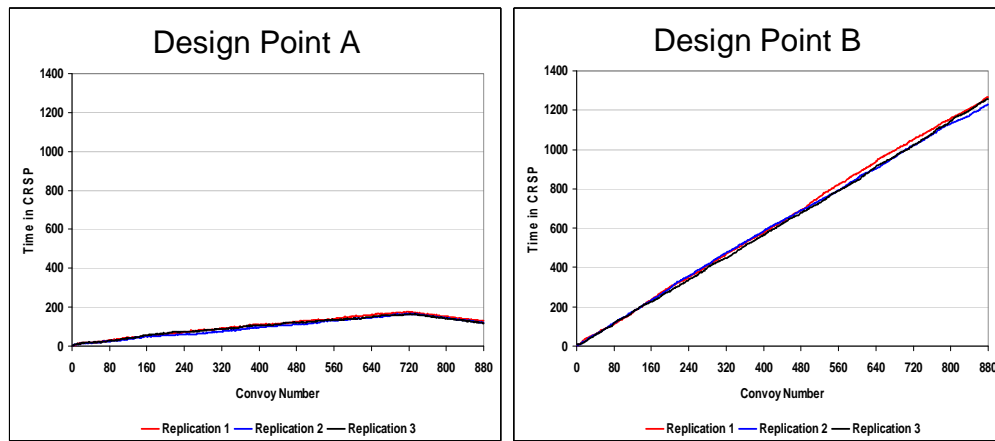


Figure 12. Time in CRSP by convoy number per network type when the rate of convoy arrival is two convoys per hour (Baez, 2008).

The first branch of the partition tree in Figure 13 reveals that traffic intensity and ITV-Available have the greatest influence on the mean time in CRSP. When traffic intensity is near zero there is little queuing in the system, while traffic intensity greater than 1.0 results in substantial queuing in the system (Baez, 2008). Furthermore, the mean time in CRSP improves with timely and reliable ITV data, even when traffic intensity nears 1.0. The partition tree clearly identifies that these two factors have the greatest influence on the mean response and is the framework for further analysis.

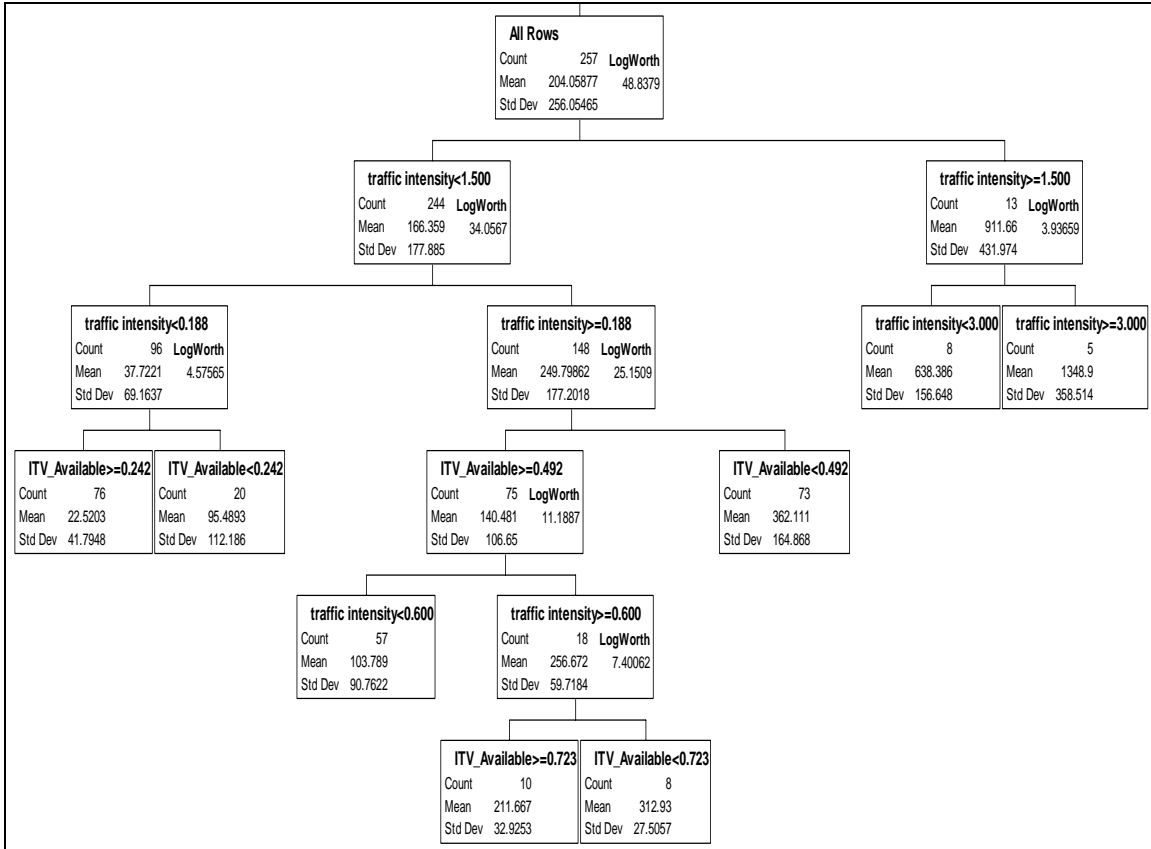


Figure 13. Partition tree for mean time in CRSP, hierarchical network structure (Baez, 2008)

Major Baez also applies multiple linear regression analysis. The multiple regression model, as the approximating function for the true functional relationship between the response and regressor variables, examines the behavior of variables of interest (Montgomery et al., 2001). Baez uses a stepwise linear regression method to fit regression metamodels for the mean time in CRSP as a function of main effects, quadratic effects, and two-way interactions. The stepwise regression control in JMP also helped identify the most influential factors.

The final regression metamodel is in Figure 14. It yields an  $R^2$  of 0.77 and contains two main effect terms and one interaction term. Other models that are considered include additional terms such as other main effects, interactions, and quadratic effects. However, these additional terms explain only 1% more of the variability. Therefore, the parsimonious model is selected. The results suggest that traffic intensity and ITV-Available are ranked as the two most influential factors. Traffic intensity is the

dominant factor, as indicated by a large |t-ratio|. These results serve to reinforce and complement those findings in the partition tree.

Whole Model					
<b>Summary of Fit</b>					
RSquare		0.768786			
RSquare Adj		0.766044			
Root Mean Square Error		123.8509			
Mean of Response		204.0588			
Observations (or Sum Wgts)		257			
<b>Analysis of Variance</b>					
		<b>Sum of</b>			
<b>Source</b>	<b>DF</b>	<b>Squares</b>	<b>Mean Square</b>	<b>F Ratio</b>	
Model	3	12903598	4301199	280.4084	
Error	253	3880781	15339	<b>Prob &gt; F</b>	
C. Total	256	16784379		<.0001*	
<b>Parameter Estimates</b>					
<b>Term</b>		<b>Estimate</b>	<b>Std Error</b>	<b>t Ratio</b>	<b>Prob&gt; t </b>
Intercept		201.08553	16.68101	12.05	<.0001*
ITV_Available		-311.4838	26.82984	-11.61	<.0001*
traffic intensity		389.47486	15.0165	25.94	<.0001*
(ITV_Available-0.50265)*(traffic intensity-0.41024)		-248.2801	48.32956	-5.14	<.0001*

Figure 14. Regression metamodel for mean time in CRSP, hierarchical network structure (Baez, 2008)

The findings from the 96-run MFML design are similar and reinforce the larger design. In validating the larger design, the MFML proves itself as a viable exploratory tool that can save the experimenter time and money. In this case, the MFML would have saved the experimenter two-thirds of the resources expended for the full design.

#### D. WIDESPREAD UTILITY OF NEW NOLH DESIGNS

In addition to Major Baez, over a dozen thesis students have used some form of FRLH, NOLH, S-NOLH, and MFML designs to complete their studies. Captain Joshua Ang (2006), Singapore Army, used FRLH designs to understand their utility and identify areas to improve the library of OLH and NOLH designs. Major Chris Michel (2006), United States Marine Corps, used earlier versions of the FRLH to find more efficient designs to evaluate the Marine Corp’s Artillery Triad. Major Jon Alt (2006), United States Army, explored tactics, techniques, and procedures that troops in small combat units could apply in response to the challenges of future combat systems and their employment. Major Earl Richardson (2006), United States Marine Corps, employed NOLHs to study sensitivities in the Infantry Warrior Simulation (IWARS).



THIS PAGE INTENTIONALLY LEFT BLANK

## VII. CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE STUDY

Immense computing power enables the simulation of increasingly complex problems, thereby allowing analysts to greatly assist decision makers in all sectors of society. Studies in human behavior and biomimetics often use computer simulations that rely on efficient designs of experiments, adopting techniques that allow analysts to “systematically examine a broader range of possible innovations . . .” (Booker, 2005). The United States Marine Corps Warfighting Lab uses agent-based models in computer experiments to understand nontraditional combat through the aggregate study of large numbers of nonlinear actors (Ilachinski, 1997).

In particular, Department of Defense faces many of the new challenges that Brown’s *Grave New World* (2003) uncovers. The impact of technology on global security, the importance of nonmilitary factors in military affairs, and the role that transnational actors play in disrupting nation-states are only the beginning in a growing list of variables that decision makers must consider—often making physical experimentation infeasible.

Analysts frequently turn to computational experimentation, such as computer simulation, to address the hundreds of input variables for the system(s) of interest. In so doing, analysts search for schemas that gain the most information from experiments. Computational and monetary costs for one experimental run push analysts to find efficient designs, especially when it is ill-advised to make assumptions about variables of interest. Earlier designs of experiments may not always be suited for computer experimentation. The recent surge of methods to construct efficient designs, most notably those with orthogonal properties (OLH and NOLH), offers analysts a new methodology for exploring extremely complex problems (Kleijnen et al., 2005). The principal shortfalls of these designs are their strict limitations in dimensionality and their scarcity. Techniques that overcome these restrictions will make orthogonal and nearly orthogonal designs an important class of design for exploring large models.

Our new techniques to construct experimental designs provide the freedom to create a variety of OLH and NOLH designs for many new design dimensions. The flexibility of this methodology enables the construction of a design with the exact dimensions that an analyst requires. It replaces the awkwardness of fitting value levels in a catalogued design, by using each value level more than once when a variable contains fewer levels than runs. Additionally, our method possesses the ability to produce new designs in a matter of hours.

Introducing a new family of S-NOLHs contributes to the discipline of design of experiments. The efficiency of examining the experimental space of  $k$  variables from a sample size,  $n = k + 1$  in a LH offers great potential for simulation experiments. The ability to select any one of  $\binom{k}{2}$  subsets of columns or exchange columns between subsets in an S-NOLH design gives analysts unprecedented flexibility.

This new construction methodology fills much of the void in the OLH and NOLH library. Catalogued designs from Cioppa's (2002) original work explore  $k = m + \binom{m-1}{2}$  factors. The gap between any two sequentially-sized designs is  $m - 1$  factors and grows as  $m$  increases. S-NOLH designs fill gaps between the number of factors, as well as the combinatorial voids created when also considering the sample size,  $n = 2^m + 1$ . For instance, a gap exists between  $O_7^{17}$  and  $N_{11}^{33}$  when there is a need to address between 8 and 10 factors for, at most, 17 runs. A saturated  $N_{16}^{17}$  easily fills this gap. Currently, no OLH or NOLH designs exist between  $n = 17$  and  $n = 33$ . We build a saturated  $N_{23}^{24}$  (Appendix A) that does not conform to  $n$  or  $k$  restrictions that exist in previous OLH and NOLH construction methods. We have catalogued a number of saturated designs that address from 2 to 63 factors, with no more than 64 runs. More designs are available for the asking.

One research area that requires further study is the performance of our new designs in the presence of interaction and quadratic terms. OLH and NOLH designs from Ye (1998), Cioppa (2002), and Ang (2006) retain their orthogonal properties in the presence of a single quadratic or two-factor interaction term. Our preliminary findings

show that RLH outperforms OLH and NOLH in terms of  $\rho_{map}$  as the number of quadratic or two-factor interaction terms increase, but this requires further study. The impact of higher order interactions in an experiment is also worthy of examination. An understanding of thresholds related to the number and order of nonlinear terms that may possibly be in an experiment will guide analysts for using our designs.

The success of VMIN for building new orthogonal or nearly orthogonal designs is a significant step for using optimization methods to construct experimental designs. The initial design that VMIN uses is important for quickly finding a solution, or in finding a solution at all. Development of a more comprehensive heuristic to initiate VMIN would be a major contribution to this area of research. Our current approach is to start with a new FRLH when VMIN finds difficulty in converging to a solution. A better method than *rms* for selecting the next column to optimize may also improve the algorithm. At times, VMIN remains on the same column throughout the optimization routine because one run of VMIN on the targeted column shows no improvement, thereby resulting in the same column with the greatest *rms*. A means to move the program to the next column should help VMIN's performance. A process to automatically "skip" to the next column after no improvements in the selected column will enable VMIN to continue. This forced movement can in turn affect the optimization of the unimproved column when the routine eventually returns to it. Currently, we manually stop the program and target the column that has the next greatest *rms*. Although this process has proven effective, a more automated means would be very useful.

THIS PAGE INTENTIONALLY LEFT BLANK

## BIBLIOGRAPHY

- Alt, J.K. (2006, June). Exploring Tactics, Techniques and Procedures for a Future Force Warrior Small Combat Unit. Master's Thesis, Monterey, CA: Naval Postgraduate School.
- Ang, J.K. (2006, December). Extending Orthogonal and Nearly Orthogonal Latin Hypercube Designs for Computer Simulation and Experimentation. Master's Thesis, Monterey, CA: Naval Postgraduate School.
- Baez, F.R. (2008, June). Combat Service Support Soldier Network Enabled Operations. Master's Thesis, Monterey, CA: Naval Postgraduate School.
- Bazaraa, M.S., Jarvis, J.J., & Sherali, H.D. (2005). *Linear Programming and Network Flows* (3rd ed.). New York, NY: John Wiley & Sons.
- Bertsimas, D., & Tsitsiklis, J.N. (1997). *Introduction to Linear Optimization*. Belmont, MA: Athena Scientific.
- Booker, L. (2005). Learning from Nature: Applying Biometric Approaches to Military Tactics and Concepts. *The Edge: Biotechnology Issue*, 9(1), 10-11.
- Box, G.E.P., Hunter, W.G., & Hunter, J.S. (1978). *Statistics for Experimenters: An Introduction to Design, Data Analysis and Model Building*. New York: John Wiley & Sons, Inc.
- Brown, M.E. (2003). *Grave New World*. Washington, D.C.: Georgetown University Press.
- Buyske, S., & Trout, R. (2001, Fall). Advanced Design of Experiments. Statistics 591 Lecture Series, Rutgers University.
- Cioppa, T.M. (2002, September). Efficient Nearly Orthogonal and Space-Filling Experimental Designs for High-Dimensional Complex Models. Doctoral Dissertation, Monterey, CA: Naval Postgraduate School.
- Cioppa, T.M., & Lucas, T.W. (2007, March). Efficient Nearly Orthogonal and Space-Filling Latin Hypercubes. *Technometrics*, 49(1), 45-55.
- Conover, W.J. (1999). *Practical Nonparametric Statistics*. New York, NY: John Wiley and Sons, Inc.
- Cplex. (2008). ILOG Solver. Cplex. Retrieved on 5 June 2008, from <http://www.ilog.com/products/cplex/>.

- David, V.V., & Nagaraja, A.B. (2003). *Order Statistics* (3rd Ed.). New York, NY: John Wiley and Sons, Inc.
- Dell, R.F. (2004, October). Class Notes from Linear Programming: OA3201. Monterey, CA: Naval Postgraduate School.
- Duma, D. (2003, July). A Perspective on Life-Cycle Evaluations. Brief to National Experimentation, Testing, Training, and Technology Conference. Retrieved on 15 December 2008, from <<http://www.dtic.mil/ndia/2003NET3/Duma.ppt#2>>.
- Fedorov, V.V. (1972). *Theory of Optimal Experiments*. New York: Academic Press.
- Florian, A. (1992). An Efficient Sampling Scheme: Updated Latin Hypercube Sampling. *Probabilistic Engineering Mechanics*, 7, 123-130.
- GAMS. (2007a). GAMS Brochure. GAMS. Retrieved on 5 June 2008, from <<http://www.gams.com/>>.
- GAMS. (2007b). GAMS Solver. GAMS. Retrieved on 5 June 2008, from <<http://www.gams.com/solvers/cplex.pdf>>.
- Graham, A. (1981). *Kronecker Products and Matrix Calculus: With Applications*. New York: Halsted Press, John Wiley and Sons.
- Hernandez, A.S. (2008a). The Efficacy of Random Latin Hypercube Designs. Working Paper, Monterey, CA: Naval Postgraduate School.
- Hernandez, A.S. (2008b). Managing Mixed Factor, Mixed Level Experiments. Working Paper, Monterey, CA: Naval Postgraduate School.
- Hernandez, A.S., Carlyle, W.M, & Lucas, T.W. (2008). Using Integer Programming to Generate Flexible Nearly Orthogonal Latin Hypercubes. Working Paper, Monterey, CA: Naval Postgraduate School.
- Hedayat, A.S., Sloane, N.J.A., Stufken, J. (1999). *Orthogonal Arrays: Theory and Applications*. New York, NY: Springer-Verlag New York, Inc.
- Ilachinski, A. (1997). *Irreducible Semi-Autonomous Adaptive Combat (ISAAC): An Artificial-Life Approach to Land Warfare*. Center for Naval Analyses Research Memorandum, CRM 97-61.10. Alexandria, VA: Center for Naval Analysis.
- Iman, R.L., & Conover, W.J. (1982). A Distribution-Free Approach to Inducing Rank Correlation Among Input Variables. *Communications in Statistics – Simulation Computations*, 11(3), 311-334.

- Iman, R.L., & Davenport, J.M. (1982). Rank Correlation Plots for Use with Correlated Input Variables. *Communications in Statistics – Simulation Computations*, 11(3), 335-360.
- Iman, R.L., & Shortencarrier, M.J. (1985, August). LHS: A Program to Generate Input Samples for Multivariate Simulations. *The American Statistician*, 39(3), 212.
- Johnson, M., Moore, L., & Ylvisaker, D. (1990). Minimax and Maximin Distance Designs. *Journal of Statistical Planning and Inference*, 26, 131-148.
- Johnson, R.T. (2008). The Design and Analysis of Computer Experiments. Doctoral Dissertation, Tempe, AZ: Arizona State University.
- Johnson, R.T., Montgomery, D.C., & Jones, B. (2008). An Empirical Study of Computer Experiments – Design Comparison and Sample Size Survey. Working Paper, Tempe, AZ: Arizona State University.
- Joint Chiefs of Staff (JCS). (2005). *Net-Centric Environment Joint Integrating Concept*, Version 1.0. Retrieved on 2 December 2008, from <[http://www.dtic.mil/futurejointwarfare/concepts/netcentric\\_jic.pdf](http://www.dtic.mil/futurejointwarfare/concepts/netcentric_jic.pdf)>.
- Joint Chiefs of Staff (JCS)/J-8. (2006). *Capabilities-Based Assessment User's Guide*, Version 2.0. Retrieved on 20 November 2007, from <[http://www.dtic.mil/futurejointwarfare/strategic/cba\\_guidev2.pdf](http://www.dtic.mil/futurejointwarfare/strategic/cba_guidev2.pdf)>.
- JMP. (2008). JMP® Statistical Discovery Software. JMP 7. Retrieved on 23 June 2008, from <<http://www.jmp.com/software>>.
- Kleijnen, J.P.C., Sanchez, S.M., Lucas, T.W., & Cioppa, T.M. (2005, Summer). A User's Guide to the Brave New World of Designing Simulation Experiments. *INFORMS Journal on Computing*, 17(3).
- Kutner, M.H., Neter, J., & Wasserman, W. (1990). *Applied Linear Statistical Models*. Homewood, IL: Richard D. Irwin, Inc.
- Law, A.M., & Kelton, W.D. (1999). *Simulation Modeling and Analysis*, 3<sup>rd</sup> Edition. New York, NY: McGraw-Hill Companies, Inc.
- Leon, S.J. (2002). *Linear Algebra with Applications*. Upper Saddle River, NJ: Prentice Hall.
- Kim, L., & Loh, H. (2003, September). Classification Trees and Bivariate Linear Discriminant Node Models. *Journal of Graphical and Statistics*, 12, 512-530.
- Lucas, T.W., & Sanchez, S.M. (2006a). The Brave New World of Designing Simulation Experiments for Defense and Homeland Security Applications. *2006 Joint Statistical Meetings Proceedings*.



- Lucas, T.W., & Sanchez, S.M. (2006b, April). Design of Experiments for Analyzing Systems of Systems. The 13th US-ROK Defense Analysis Seminar.
- McKay, M.D., Beckman, R.J., & Conover, W.J. (1979, May). A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code. *Technometrics*, 2(2), 239-245.
- Michel, C. (2006, September). Evaluating the Marine Corp's Artillery Triad in STOM Operations. Master's Thesis, Monterey, CA: Naval Postgraduate School.
- Montgomery, D.C. (2005). *Design and Analysis of Experiments*. Hoboken, NJ: John Wiley & Sons, Inc.
- Montgomery, D.C., Peck, E.A., & Vining, C.G. (2001). *Introduction to Linear Regression Analysis*. New York: John Wiley & Sons, Inc.
- Nash, S.G., & Sofer, A. (1996). *Linear and Nonlinear Programming*. New York, NY: McGraw-Hill Companies, Inc.
- Owen, A.B. (1994, December). Controlling Correlations in Latin Hypercube Samples. *Journal of the American Statistical Association: Theory and Methods*, 89(428), 1517-1522.
- Owen, A.B. (1992). Orthogonal Arrays for Computer Experiments, Integration, and Visualization. *Statistica Sinica*, 2, 439-452.
- Patterson, H.D. (1954). The Errors of Lattice Sampling. *Journal of the Royal Statistical Society, Series B (Methodological)*, 16(1), 140-149.
- Richardson, E., (2006, September). Analyzing the Distributed Capabilities of the Future Force Warrior Small Combat Unit (FFW SCU). Master's Thesis, Monterey, CA: Naval Postgraduate School.
- Roginski, J.W. (2006, June). Emergency First response to a Crisis Event: A Multi-Agent Simulation Approach. Master's Thesis, Monterey, CA: Naval Postgraduate School.
- Saeger, K.J., & Hinch, J.H. (2001). Understanding Instability in a Complex Deterministic Combat Simulation. *Military Operations Research*, 6, 43-55.
- Sanchez, S.M., & Lucas, T.W. (2002). Exploring the World of Agent-Based Simulations: Simple Models, Complex Analyses. *Proceedings of the 2002 Winter Simulation Conference*, eds. J. Snowdon, J. Charnes, C-H Chen, & E.Yücesan. I Piscataway, NJ: IEEE.

- Sanchez, S.M. (2006). NOLH Designs Spreadsheet. Retrieved on 17 October 2007, from <<http://diana.cs.navy.mil/SeedLab/>>.
- Sanchez, S.M., & Sanchez, P.J. (2005). Very Large Fractional Factorial and Central Composite Designs. *ACM Transactions on Modeling and Computer Simulation*, 15(4), 362-377.
- Santner, T.J., Williams, B.J., & Notz, W. (2003). *The Design and Analysis of Computer Experiments*. New York: Springer Verlag.
- Satterthwaite, F.E. (1959, May). Random Balance Experimentation. *Technometrics*, 1(2), 111-137.
- Sloane, N.J.A. (2006). A Library of Orthogonal Arrays. Retrieved on 15 December 2008, from <<http://www.research.att.com/~njas/>>.
- Steinberg, D.M., & Lin, D.K.J. (2006). A Construction Method for Orthogonal Latin Hypercube Designs. *Biometrika*, 93(2), 279-288.
- Sugiyama, S.O., & Chow, J.W. (1997, April). @Risk, Riskview and BestFit. *OR/MS Today*, 24(2), 64-66.
- Tang, B. (1993, December). Orthogonal Array-Based Latin Hypercubes. *Journal of the American Statistical Association: Theory and Methods*, 88(424), 1392-1397.
- Tang, B. (1998). Selecting Latin Hypercubes Using Correlation Criteria. *Statistica Sinica*, 8, 965-977.
- Wolsey, L.A. (1998). *Integer Programming*. New York, NY: John Wiley and Sons, Inc.
- Wyss, G.D., & Jorgensen, K.H. (1998, February). A User's Guide to LHS: Sandia's Latin Hypercube Sampling Software. Albuquerque, NM: Sandia National Laboratories.
- Ye, K.Q. (1998, December). Orthogonal Column Latin Hypercubes and Their Application in Computer Experiments. *Journal of the American Statistical Association – Theory and Methods*, 93(444), 1430-1439.

THIS PAGE INTENTIONALLY LEFT BLANK

## APPENDIX A. SATURATED NOLH DESIGNS

Appendix A displays a number of saturated NOLH designs that we developed during the course of our study. Each design is designated with a symbol per Cioppa (2002) and the nonorthogonality measure for the design.

$$N_{12}^{14} \text{ with } \rho_{map} = 0.0462$$

	k1	k2	k3	k4	k5	k6	k7	k8	k9	k10	k11	k12
n1	3	6	1	6	14	5	5	6	10	6	14	6
n2	1	7	9	10	5	14	2	8	8	14	4	4
n3	12	8	3	14	4	7	13	2	11	13	8	11
n4	9	5	4	3	3	2	3	12	6	9	2	14
n5	10	1	7	1	10	8	14	9	7	12	6	2
n6	11	12	10	13	12	1	6	11	9	7	3	1
n7	6	3	14	11	11	11	12	10	4	4	7	13
n8	4	11	11	7	7	6	9	14	12	11	13	12
n9	2	4	5	12	9	4	7	4	3	5	5	8
n10	14	2	12	8	6	9	1	5	13	3	11	7
n11	7	10	13	4	2	3	8	1	2	8	12	5
n12	5	13	8	2	8	10	11	3	14	2	1	9
n13	8	9	2	9	1	12	10	13	5	1	10	3
n14	13	14	6	5	13	13	4	7	1	10	9	10

$$N_{15}^{16} \text{ with } \rho_{map} = 0.0471$$

	k1	k2	k3	k4	k5	k6	k7	k8	k9	k10	k11	k12	k13	k14	k15
n1	10	8	13	6	16	8	15	16	14	3	12	2	13	7	9
n2	13	12	4	8	13	15	3	4	9	4	3	3	4	2	13
n3	7	6	12	1	2	11	4	5	11	5	16	4	3	12	5
n4	9	16	6	4	12	5	8	14	5	6	10	15	2	15	12
n5	11	10	8	5	15	1	7	2	15	16	7	9	9	9	2
n6	14	4	1	10	3	7	12	6	16	7	11	14	12	10	16
n7	6	9	14	16	11	3	9	1	3	8	15	6	7	8	15
n8	1	13	16	7	5	10	5	7	12	9	1	10	16	11	14
n9	12	7	10	2	1	2	13	12	1	13	4	5	8	3	11
n10	16	14	11	13	6	12	2	13	6	11	14	13	14	4	4
n11	2	3	5	3	14	16	10	8	4	15	13	12	11	5	10
n12	3	2	9	11	9	4	6	11	10	1	5	16	6	1	3
n13	15	1	15	12	10	14	11	9	7	10	2	11	5	16	7
n14	5	5	2	14	8	6	1	15	8	12	8	1	10	14	8
n15	4	15	7	15	4	13	16	10	13	14	9	8	1	6	6
n16	8	11	3	9	7	9	14	3	2	2	6	7	15	13	1

$N_{16}^{17}$  with  $\rho_{map} = 0.0490$

	k1	k2	k3	k4	k5	k6	k7	k8	k9	k10	k11	k12	k13	k14	k15	k16
n1	5	9	13	7	15	1	15	6	5	2	7	11	1	9	14	3
n2	13	14	1	16	12	5	6	3	11	5	6	1	6	7	3	13
n3	14	2	6	9	9	2	16	8	9	17	11	17	10	3	4	12
n4	12	13	16	10	11	17	9	1	17	15	12	13	2	13	8	7
n5	4	15	11	15	6	12	7	16	6	6	15	14	4	1	10	15
n6	9	4	15	17	14	8	13	15	10	7	17	3	15	14	6	9
n7	10	7	14	3	10	4	4	17	13	14	3	4	3	11	11	17
n8	8	12	7	1	13	13	14	4	2	12	14	2	12	5	13	14
n9	17	6	8	2	3	15	11	13	7	1	10	7	5	10	2	5
n10	1	3	2	6	17	14	3	10	8	9	9	15	9	15	5	11
n11	2	17	3	8	1	6	17	11	12	11	8	9	11	17	7	8
n12	16	11	4	12	7	7	2	12	3	16	13	8	7	12	16	1
n13	15	10	12	13	8	11	10	7	4	4	1	16	16	16	15	16
n14	6	1	5	14	5	16	12	9	15	10	4	5	8	4	17	6
n15	3	8	17	11	4	10	5	5	1	13	2	6	13	6	1	4
n16	7	5	10	4	2	3	1	2	14	3	16	10	14	8	12	10
n17	11	16	9	5	16	9	8	14	16	8	5	12	17	2	9	2

$N_{18}^{19}$  with  $\rho_{map} = 0.0456$

	k1	k2	k3	k4	k5	k6	k7	k8	k9	k10	k11	k12	k13	k14	k15	k16	k17	k18
n1	16	11	2	6	13	12	11	3	14	6	3	19	9	1	11	6	17	4
n2	1	3	17	1	14	5	10	2	19	5	9	6	12	9	13	8	4	12
n3	14	6	19	3	11	15	14	18	12	19	6	11	1	7	4	9	13	16
n4	8	18	18	17	10	8	8	5	5	14	1	1	16	3	6	11	16	10
n5	17	13	15	19	15	2	18	15	16	8	13	8	7	14	15	4	7	2
n6	15	10	10	14	2	3	13	6	17	9	7	17	13	16	5	19	12	19
n7	4	15	13	11	3	13	19	8	4	11	17	18	14	5	17	2	10	15
n8	13	16	16	2	7	10	6	16	8	4	14	15	18	11	1	12	5	1
n9	18	9	5	7	12	4	9	4	2	7	19	2	6	8	3	5	9	17
n10	11	7	14	18	9	18	1	1	9	15	11	16	3	12	10	10	1	8
n11	7	4	12	10	8	7	2	14	1	2	4	13	8	19	14	3	18	11
n12	12	19	11	9	18	19	5	12	13	3	15	7	10	13	18	16	14	18
n13	6	2	9	16	4	17	16	13	10	1	12	5	5	4	7	17	11	5
n14	10	1	6	15	19	6	7	19	7	13	10	14	19	2	12	14	6	14
n15	19	5	7	4	6	16	15	7	6	16	8	3	17	17	19	13	8	6
n16	5	12	8	5	5	1	3	11	11	18	16	10	2	6	16	18	15	3
n17	2	17	3	8	17	9	17	10	3	10	2	12	4	15	8	15	2	9
n18	3	8	4	13	16	14	12	9	15	17	18	9	15	18	2	7	19	7
n19	9	14	1	12	1	11	4	17	18	12	5	4	11	10	9	1	3	13

$N_{23}^{24}$  with  $\rho_{map} = 0.0496$

	k1	k2	k3	k4	k5	k6	k7	k8	k9	k10	k11	k12	k13	k14	k15	k16	k17	k18	k19	k20	k21	k22	k23
n1	5	23	21	15	20	6	15	3	3	15	10	4	16	5	13	16	11	1	1	15	18	21	12
n2	9	5	10	13	22	14	4	24	15	5	1	17	22	1	11	13	21	8	8	16	8	10	23
n3	19	1	22	7	21	8	10	6	13	17	22	11	1	4	22	15	22	13	13	18	9	1	8
n4	20	16	17	14	18	19	8	4	23	3	18	20	14	17	24	2	9	14	7	5	23	18	20
n5	12	21	11	2	16	11	7	10	9	1	23	9	15	14	2	22	3	22	16	21	13	3	22
n6	4	22	18	4	13	1	23	18	19	14	20	24	20	12	17	11	18	12	20	9	2	19	13
n7	10	14	7	23	11	2	20	8	22	4	5	1	2	6	15	6	4	16	17	8	4	11	16
n8	22	17	15	11	3	18	22	9	2	2	9	21	8	7	5	3	20	18	6	23	7	15	6
n9	14	18	3	21	14	23	5	5	8	22	16	15	12	3	18	21	13	20	21	13	3	24	14
n10	2	4	5	3	17	10	12	1	10	12	3	16	9	22	9	19	16	23	4	2	12	16	9
n11	24	7	19	6	15	5	6	16	17	23	4	8	21	16	10	7	2	21	14	22	10	23	7
n12	3	19	12	5	10	24	2	13	21	20	14	14	6	11	3	1	7	5	9	10	6	6	3
n13	8	12	13	19	12	16	17	23	4	11	19	7	24	9	21	10	5	24	5	6	15	2	1
n14	11	2	4	8	6	17	16	19	14	9	24	3	10	21	20	14	12	4	2	20	5	22	18
n15	1	6	20	16	7	21	19	7	16	19	11	2	19	15	8	5	24	19	22	17	21	12	21
n16	15	24	6	9	5	7	11	21	5	24	6	13	4	18	23	8	17	15	10	11	20	7	24
n17	21	13	24	12	8	22	14	11	7	10	2	10	18	24	16	20	10	2	18	3	1	5	15
n18	7	8	9	10	23	20	24	17	6	13	7	22	5	13	14	12	1	6	24	19	22	13	11
n19	6	15	16	22	2	9	3	12	20	7	8	19	11	19	19	24	14	11	15	24	19	9	2
n20	17	10	8	1	1	12	9	15	12	6	12	5	13	2	12	17	15	7	23	1	24	20	5
n21	18	3	2	18	4	3	13	2	11	21	17	23	23	10	7	9	6	3	12	12	14	4	17
n22	23	20	1	17	24	15	21	14	24	16	13	6	17	20	6	18	23	9	11	14	17	8	4
n23	16	9	23	20	9	13	18	22	18	18	15	18	3	8	1	23	8	17	3	4	16	14	19
n24	13	11	14	24	19	4	1	20	1	8	21	12	7	23	4	4	19	10	19	7	11	17	10

$N_{24}^{25}$  with  $\rho_{map} = 0.0477$

	k1	k2	k3	k4	k5	k6	k7	k8	k9	k10	k11	k12	k13	k14	k15	k16	k17	k18	k19	k20	k21	k22	k23	k24
n1	20	2	5	11	6	23	9	14	23	19	20	13	8	9	25	25	11	5	10	21	3	20	15	15
n2	16	20	14	10	3	10	21	25	7	7	25	12	10	3	15	18	25	17	19	17	23	12	18	24
n3	6	25	3	18	4	14	7	3	14	11	21	3	13	25	13	10	22	25	6	15	5	17	12	17
n4	5	10	9	12	9	2	25	8	9	23	10	1	24	13	11	19	7	7	21	12	4	7	25	13
n5	13	4	16	24	8	16	24	2	15	13	11	22	1	10	2	9	5	23	22	18	12	15	10	22
n6	21	3	6	23	11	6	11	6	2	6	13	6	18	4	14	7	19	8	15	9	17	25	6	1
n7	15	11	19	2	14	5	22	22	24	2	18	10	12	24	8	6	4	15	12	22	14	24	16	2
n8	3	8	11	16	24	25	23	15	5	8	17	20	23	18	6	20	21	9	7	25	13	6	7	6
n9	11	7	24	4	12	17	3	9	10	3	23	7	9	11	7	4	14	1	17	5	2	3	11	19
n10	23	21	18	19	5	15	10	19	6	10	2	15	2	17	1	22	16	10	3	10	6	11	23	3
n11	1	19	7	21	10	8	2	23	19	12	9	17	20	5	4	12	2	2	9	16	15	21	9	23
n12	22	5	13	20	15	1	5	10	11	4	15	23	22	21	23	14	6	19	5	13	16	2	21	21
n13	17	12	12	6	13	22	14	7	18	16	19	5	19	2	3	21	3	24	2	3	25	9	13	12
n14	14	23	20	15	17	11	15	1	21	5	4	4	7	16	22	23	13	3	16	20	24	8	4	14
n15	25	18	4	8	18	9	6	11	25	22	8	18	17	7	5	3	23	13	20	23	11	1	14	10
n16	10	1	21	17	20	19	1	21	12	21	6	2	14	22	9	17	20	21	24	14	22	18	19	18
n17	4	13	1	14	19	20	16	24	16	1	3	9	6	6	24	8	9	22	18	6	9	5	17	8
n18	18	17	15	13	1	21	12	20	3	24	12	11	16	20	21	1	1	12	14	19	18	4	3	9
n19	19	9	8	3	25	4	20	17	4	20	5	8	5	14	12	11	15	14	1	11	7	16	2	25
n20	8	15	17	25	23	12	18	13	20	25	24	14	4	12	18	2	17	4	4	7	21	13	24	11
n21	7	22	23	7	22	13	4	5	1	15	14	19	11	1	20	13	8	20	13	24	8	22	20	7
n22	2	6	25	9	2	3	13	16	22	18	7	21	15	8	19	15	24	18	8	8	10	10	1	4
n23	9	16	2	5	16	7	8	12	8	17	22	25	3	23	10	24	10	11	25	2	19	14	8	5
n24	12	14	10	1	7	24	17	4	13	9	1	24	21	19	16	5	18	6	11	4	20	23	22	20
n25	24	24	22	22	21	18	19	18	17	14	16	16	25	15	17	16	12	16	23	1	1	19	5	16



$N_{32}^{33}$  with  $\rho_{map} = 0.0434$

	k1	k2	k3	k4	k5	k6	k7	k8	k9	k10	k11	k12	k13	k14	k15	k16	k17	k18	k19	k20	k21	k22	k23	k24	k25	k26	k27	k28	k29	k30	k31	k32
n1	14	23	5	18	12	26	4	7	31	32	10	8	1	1	3	10	20	28	18	17	27	24	24	12	3	24	24	23	10	8	17	32
n2	5	1	6	3	15	2	23	19	23	31	27	31	5	20	20	18	17	32	26	28	14	15	2	9	23	7	17	13	25	32	16	26
n3	26	6	8	17	32	5	2	13	29	14	32	14	21	18	23	31	29	13	21	15	26	6	22	25	5	9	29	7	1	13	22	2
n4	29	16	14	4	18	28	33	18	20	19	8	3	14	33	33	11	14	19	19	8	10	30	32	23	22	6	30	9	7	26	26	30
n5	13	29	3	6	13	30	20	32	18	3	21	33	28	19	7	15	22	11	24	2	11	5	7	14	2	8	6	27	5	15	23	27
n6	19	11	31	24	26	18	22	3	4	22	13	25	30	7	27	21	32	30	10	4	5	9	11	6	12	30	27	22	6	24	6	31
n7	2	10	28	30	21	17	13	16	22	2	15	15	11	21	16	5	24	22	29	12	13	1	31	4	25	14	15	4	32	5	33	24
n8	24	18	11	22	1	4	26	6	6	10	18	28	4	3	28	7	30	10	20	11	22	31	17	24	15	26	1	3	8	20	29	12
n9	30	26	7	23	16	11	24	28	17	7	31	13	9	13	24	13	5	33	6	1	19	20	6	11	20	18	32	15	31	1	4	15
n10	21	33	18	29	3	7	14	17	28	29	26	24	32	12	25	30	13	8	2	30	6	19	29	15	18	4	11	10	23	11	20	33
n11	15	14	21	31	14	33	1	33	11	12	6	18	10	25	29	32	33	26	12	27	21	29	1	27	10	13	19	16	26	19	24	19
n12	28	3	20	13	8	19	5	26	5	15	4	32	26	9	5	3	7	9	15	19	28	13	28	22	13	11	33	2	27	27	5	18
n13	18	24	10	21	28	21	8	4	19	33	7	29	29	32	9	8	6	27	7	9	18	16	3	26	32	20	9	5	11	10	30	7
n14	3	19	12	1	23	22	15	11	3	13	23	19	13	15	2	26	31	3	9	23	1	33	20	13	29	17	31	8	19	3	14	14
n15	23	30	33	16	33	29	17	20	25	28	22	27	2	10	30	4	8	1	30	26	4	11	8	17	6	22	21	12	20	18	15	8
n16	32	13	29	2	30	9	21	22	8	20	24	12	33	16	13	6	23	25	22	32	25	28	16	18	9	19	5	29	28	2	27	25
n17	4	21	25	8	25	1	19	27	27	5	3	30	17	8	19	33	1	23	17	14	16	25	27	29	16	32	22	25	14	17	28	16
n18	31	8	9	33	24	23	25	2	12	1	17	23	6	23	4	24	4	24	23	33	3	17	23	28	17	12	8	24	9	14	7	20
n19	16	12	30	27	4	3	31	15	30	24	1	17	18	22	6	12	27	7	25	10	12	21	5	20	19	1	28	31	15	6	13	5
n20	10	9	17	15	5	10	9	8	21	6	29	2	20	24	17	2	12	5	3	20	9	10	4	33	14	33	26	28	21	28	25	29
n21	25	17	4	20	31	8	3	23	26	17	9	6	23	17	15	17	25	6	28	3	8	27	19	16	27	25	2	19	29	31	1	22
n22	1	31	2	25	27	15	29	14	2	26	14	20	19	26	26	9	26	16	16	22	30	7	33	30	11	15	23	30	30	21	9	11
n23	9	27	26	11	17	25	32	21	32	9	16	4	25	4	11	16	28	29	14	29	23	8	13	32	26	16	10	1	12	22	2	13
n24	8	5	24	26	19	24	12	30	24	16	33	22	16	14	21	1	16	20	1	16	17	32	30	8	28	10	7	33	2	23	12	4
n25	11	28	15	14	7	13	6	9	16	4	11	16	27	31	32	14	9	14	33	31	29	22	12	1	24	23	20	17	3	9	3	17
n26	20	4	1	28	10	31	28	29	14	27	19	9	31	2	18	25	11	12	31	25	15	12	15	10	30	31	25	21	16	16	32	9
n27	12	20	22	32	29	6	30	31	7	23	20	5	8	28	1	23	10	4	8	18	31	18	10	5	8	21	14	6	4	25	18	28
n28	17	7	13	5	6	14	16	24	15	25	2	7	12	27	22	19	15	21	4	21	2	4	25	7	1	27	3	11	18	7	8	1
n29	27	22	32	10	2	20	11	25	9	30	30	21	3	29	12	28	21	18	27	6	24	3	26	31	33	29	12	26	13	12	10	21
n30	7	2	19	9	22	27	18	5	13	18	12	11	15	5	31	27	2	2	11	5	33	14	9	21	21	2	4	20	24	4	11	23
n31	22	15	23	12	11	32	27	1	33	11	25	26	22	30	8	29	18	15	13	13	32	23	21	3	7	28	16	18	33	29	19	6
n32	33	32	16	7	20	12	10	12	10	8	5	10	7	6	14	20	19	17	5	24	20	2	14	2	31	5	18	32	17	33	31	10
n33	6	25	27	19	9	16	7	10	1	21	28	1	24	11	10	22	3	31	32	7	7	26	18	19	4	3	13	14	22	30	21	3



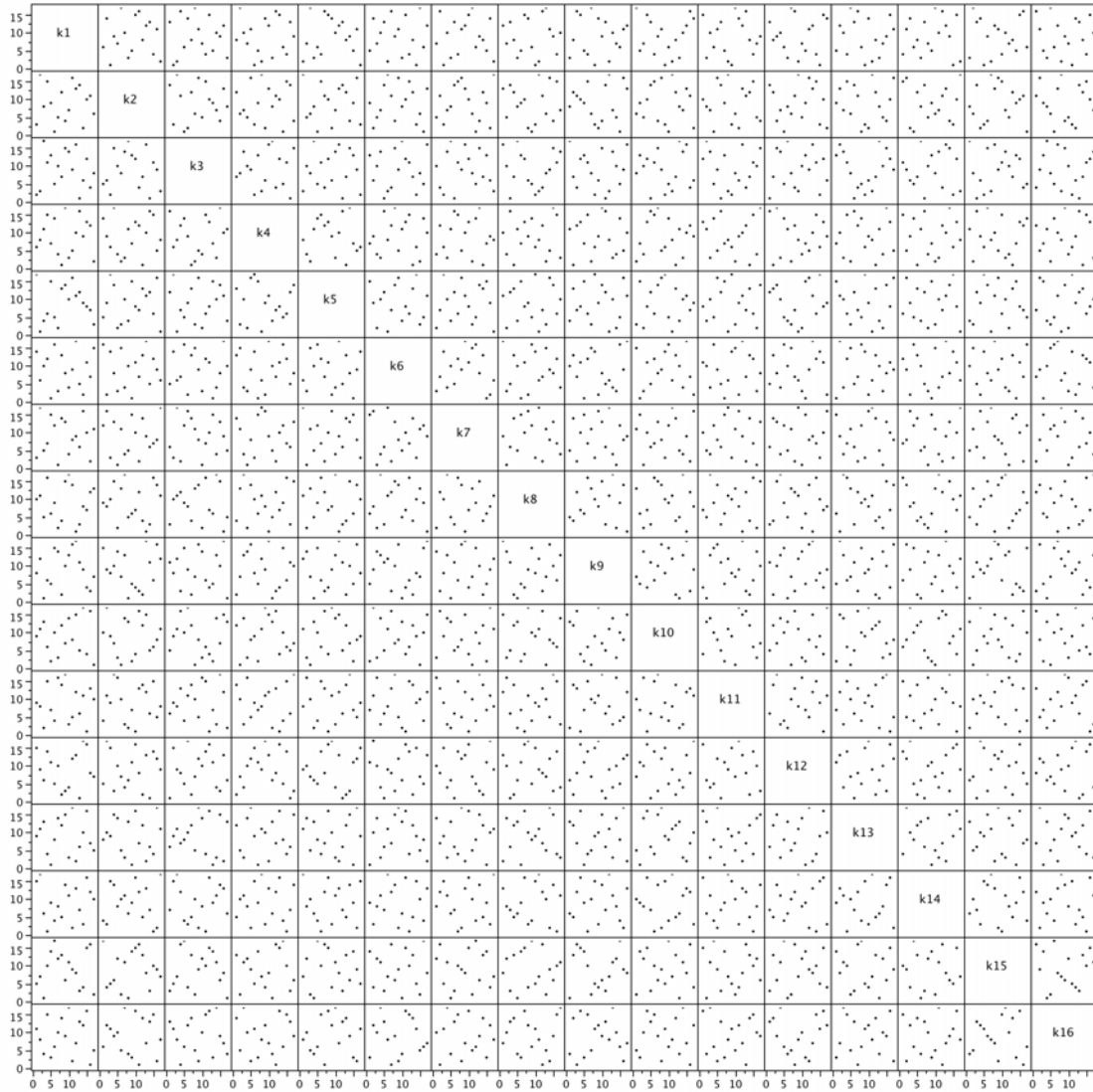






## APPENDIX B. PAIRWISE PLOT OF FACTORS IN $N_{16}^{17}$

Appendix B displays a pairwise plot of the factors in  $N_{16}^{17}$ . The dispersion of points in each pairwise plot shows reasonable space filling properties.



THIS PAGE INTENTIONALLY LEFT BLANK

## **APPENDIX C. MAJOR ROGINSKI'S DESIGN**

This Appendix lists the MFML design matrix for Major Roginski's experiment. The design is split into eight sections. Each section heading describes the design points and the factors shown in the section. It also includes the nonorthogonality of the design.

Roginski MFML Section 1:  $n = 1$  thru 72,  $k = 1$  thru 11 with  $\rho_{map} = 0.0593$ .

ESU	FO Police	Traffic	Time OP	Panic	Num Ag1	Num Ag2	Num Ag3	Num Ag4	Num Gun	Num EMT SJ
0	0	0	1	1	41	45	141	103	63	64
0	0	1	1	1	82	6	117	3	103	38
0	1	0	1	1	7	18	127	53	67	107
0	1	1	1	1	87	86	111	14	24	66
1	0	0	1	1	5	118	73	80	12	48
1	0	1	1	1	132	82	63	47	86	97
1	1	0	1	1	32	109	138	66	66	18
1	1	1	1	1	110	41	86	96	21	102
0	0	0	1	2	143	137	104	120	47	88
0	0	1	1	2	25	127	122	54	16	128
0	1	0	1	2	39	32	107	62	51	65
0	1	1	1	2	8	139	67	51	7	129
1	0	0	1	2	40	54	144	7	130	21
1	0	1	1	2	56	99	109	16	141	25
1	1	0	1	2	117	26	94	94	89	42
1	1	1	1	2	37	89	78	46	90	54
0	0	0	1	3	10	1	9	63	107	31
0	0	1	1	3	130	104	75	56	116	71
0	1	0	1	3	42	62	39	134	108	92
0	1	1	1	3	89	126	85	83	50	16
1	0	0	1	3	100	21	130	68	129	33
1	0	1	1	3	46	129	14	12	52	3
1	1	0	1	3	124	92	136	114	98	67
1	1	1	1	3	58	134	17	13	118	117
0	0	0	2	1	74	74	137	87	126	137
0	0	1	2	1	90	13	84	109	32	43
0	1	0	2	1	57	73	108	55	142	40
0	1	1	2	1	33	57	116	122	78	96
1	0	0	2	1	125	24	10	88	82	76
1	0	1	2	1	65	125	81	121	65	29
1	1	0	2	1	92	63	56	4	15	115
1	1	1	2	1	20	90	4	113	48	105
0	0	0	2	2	72	2	103	44	18	94
0	0	1	2	2	69	39	93	99	54	13
0	1	0	2	2	77	12	15	82	72	87
0	1	1	2	2	119	43	123	115	95	37
1	0	0	2	2	78	84	27	26	33	47
1	0	1	2	2	54	93	139	127	41	132
1	1	0	2	2	144	96	47	110	76	104
1	1	1	2	2	38	19	92	141	131	89
0	0	0	2	3	18	120	43	59	55	100
0	0	1	2	3	102	94	90	61	59	60
0	1	0	2	3	120	35	61	36	23	27
0	1	1	2	3	126	80	70	101	94	93
1	0	0	2	3	133	77	51	21	34	144
1	0	1	2	3	31	66	6	131	2	116
1	1	0	2	3	3	107	35	102	111	45
1	1	1	2	3	95	131	95	75	125	142
0	0	0	3	1	75	4	29	116	9	62
0	0	1	3	1	79	70	60	15	68	61
0	1	0	3	1	24	58	50	49	133	127
0	1	1	3	1	43	88	24	29	45	82
1	0	0	3	1	51	115	32	144	20	17
1	0	1	3	1	6	108	99	105	123	81
1	1	0	3	1	98	10	118	97	113	85
1	1	1	3	1	88	119	68	48	140	108
0	0	0	3	2	45	31	45	140	87	103
0	0	1	3	2	36	56	40	25	97	10
0	1	0	3	2	13	34	97	60	144	99
0	1	1	3	2	1	128	131	117	73	68
1	0	0	3	2	35	117	82	9	61	41
1	0	1	3	2	115	9	129	11	14	140
1	1	0	3	2	61	52	112	130	37	28
1	1	1	3	2	68	15	102	40	26	141
0	0	0	3	3	67	51	72	57	110	118
0	0	1	3	3	113	101	135	67	60	124
0	1	0	3	3	99	44	36	108	44	26
0	1	1	3	3	52	23	21	50	127	12
1	0	0	3	3	97	17	33	73	42	131
1	0	1	3	3	129	124	66	137	119	74
1	1	0	3	3	108	69	42	107	8	23
1	1	1	3	3	34	33	26	98	81	6

Roginski MFML Section 2:  $n = 73$  thru 144,  $k = 1$  thru 11 with  $\rho_{map} = 0.0593$ .

ESU	FO Police	Traffic	Time OP	Panic	Num Ag1	Num Ag2	Num Ag3	Num Ag4	Num Gun	Num EMT SJ
0	0	0	1	1	136	46	16	125	38	51
0	0	1	1	1	9	71	124	65	17	55
0	1	0	1	1	26	95	54	39	3	2
0	1	1	1	1	116	49	28	85	100	126
1	0	0	1	1	91	47	134	32	56	86
1	0	1	1	1	27	75	5	76	70	134
1	1	0	1	1	93	143	87	106	22	39
1	1	1	1	1	123	91	44	81	102	24
0	0	0	1	2	21	11	37	138	35	114
0	0	1	1	2	94	123	46	17	84	98
0	1	0	1	2	64	64	114	86	71	5
0	1	1	1	2	127	140	1	84	120	9
1	0	0	1	2	128	112	98	78	40	14
1	0	1	1	2	140	27	3	132	83	35
1	1	0	1	2	62	130	106	41	29	8
1	1	1	1	2	30	8	100	58	121	83
0	0	0	1	3	50	138	74	129	132	58
0	0	1	1	3	134	59	30	42	136	90
0	1	0	1	3	85	114	58	93	58	56
0	1	1	1	3	86	5	55	22	117	95
1	0	0	1	3	44	53	7	30	69	7
1	0	1	1	3	80	102	132	139	128	20
1	1	0	1	3	19	76	115	10	36	91
1	1	1	1	3	15	14	91	136	39	123
0	0	0	2	1	103	113	48	128	1	22
0	0	1	2	1	106	50	34	74	62	112
0	1	0	2	1	29	25	77	23	53	84
0	1	1	2	1	84	67	69	2	114	19
1	0	0	2	1	131	22	133	123	137	44
1	0	1	2	1	70	105	126	20	57	106
1	1	0	2	1	14	3	12	77	91	57
1	1	1	2	1	122	142	89	95	74	32
0	0	0	2	2	16	7	105	135	49	52
0	0	1	2	2	59	87	121	143	28	50
0	1	0	2	2	63	42	142	38	31	75
0	1	1	2	2	76	141	88	19	139	72
1	0	0	2	2	11	121	8	112	124	101
1	0	1	2	2	12	30	52	91	138	30
1	1	0	2	2	17	133	18	24	79	36
1	1	1	2	2	114	83	120	124	106	49
0	0	0	2	3	83	97	128	52	134	4
0	0	1	2	3	137	65	53	37	96	63
0	1	0	2	3	66	110	83	31	115	70
0	1	1	2	3	48	40	2	111	112	73
1	0	0	2	3	71	122	62	28	104	139
1	0	1	2	3	104	81	41	69	6	34
1	1	0	2	3	28	132	31	119	105	121
1	1	1	2	3	109	55	125	64	80	78
0	0	0	3	1	4	135	119	118	19	113
0	0	1	3	1	112	116	65	35	101	130
0	1	0	3	1	60	20	80	5	143	119
0	1	1	3	1	53	79	11	142	93	138
1	0	0	3	1	118	38	23	27	5	15
1	0	1	3	1	135	48	38	6	30	109
1	1	0	3	1	101	68	59	72	25	59
1	1	1	3	1	138	111	76	92	92	69
0	0	0	3	2	111	61	20	33	64	143
0	0	1	3	2	121	100	113	100	75	79
0	1	0	3	2	55	103	140	79	11	133
0	1	1	3	2	105	144	19	18	88	110
1	0	0	3	2	49	136	64	45	13	77
1	0	1	3	2	107	29	25	104	77	125
1	1	0	3	2	23	36	49	34	4	11
1	1	1	3	2	142	37	101	1	43	53
0	0	0	3	3	141	78	79	90	10	111
0	0	1	3	3	47	28	143	43	27	136
0	1	0	3	3	22	85	13	89	122	122
0	1	1	3	3	96	72	71	71	135	80
1	0	0	3	3	2	60	57	8	99	1
1	0	1	3	3	139	16	96	126	85	46
1	1	0	3	3	81	106	22	70	46	120
1	1	1	3	3	73	98	110	133	109	135



Roginski MFML Section 3:  $n = 1$  thru 72,  $k = 12$  thru 22 with  $\rho_{map} = 0.0593$ .

Num EMT MM	Num ESU1	Num ESU2	Num ESU3	Num ESU4	Num ESU5	Num Foll1	Num Foll2	Num Foll3	Num Traff1	Num Traff2
66	58	113	110	36	31	106	11	89	54	134
85	118	48	113	116	96	144	53	91	100	124
99	66	130	1	93	69	99	135	127	118	99
27	16	19	21	144	93	11	125	93	73	107
53	82	142	105	64	66	9	130	2	29	73
144	22	5	42	139	51	38	25	74	67	138
118	26	31	50	114	2	128	94	38	70	1
139	52	139	72	23	75	119	13	60	51	47
143	70	84	139	121	35	44	56	138	110	116
76	112	36	124	123	101	140	29	112	104	75
84	116	9	6	87	38	96	137	9	81	130
89	109	88	112	3	26	108	139	115	55	62
138	140	1	44	24	104	27	114	114	139	128
36	51	49	101	90	4	15	1	46	123	117
68	86	2	45	72	57	80	120	141	117	3
21	43	21	94	118	5	28	133	139	124	103
3	48	42	51	20	99	41	32	102	30	120
17	142	26	141	16	54	87	110	59	72	17
121	105	123	8	119	128	138	33	35	105	57
9	101	92	65	79	133	71	16	65	5	49
95	122	32	119	5	33	36	104	5	21	110
25	59	134	34	31	7	107	119	72	43	119
55	67	4	4	96	107	42	127	85	12	66
91	83	18	129	97	137	60	132	86	102	92
19	61	116	48	82	15	53	44	56	62	141
15	130	30	16	73	88	58	93	7	15	40
10	92	40	47	4	71	94	99	131	4	14
134	98	78	69	138	92	37	62	83	115	34
59	4	109	24	135	70	47	124	140	17	33
80	75	33	23	41	14	129	37	73	39	74
123	25	25	15	55	52	25	9	16	60	18
81	115	60	11	9	84	83	48	47	82	91
18	5	121	116	14	109	32	90	25	93	98
56	108	83	106	120	82	30	118	36	53	26
117	12	115	138	74	126	110	138	69	37	123
58	56	105	12	2	132	76	105	105	74	114
93	141	43	71	80	36	122	28	19	25	93
29	29	77	122	126	115	115	116	33	42	21
67	57	70	87	63	81	65	27	144	38	53
7	106	46	35	109	49	56	5	37	103	139
41	55	95	85	110	60	141	70	22	133	58
94	64	124	67	133	43	13	109	8	66	121
77	138	135	54	105	11	121	122	137	32	41
35	94	6	142	50	64	130	134	39	40	105
136	128	122	9	54	98	86	76	84	130	135
20	62	37	84	47	1	20	111	50	31	68
96	124	57	126	88	10	64	50	106	135	5
108	30	15	81	32	118	31	126	11	45	76
62	18	131	144	104	139	69	101	130	16	28
105	71	55	33	38	83	131	8	81	10	96
116	47	62	82	39	16	75	65	32	83	30
130	36	11	134	134	131	120	112	57	76	132
4	76	41	41	70	114	135	31	126	114	109
129	46	87	25	68	22	2	107	113	89	29
13	74	81	52	142	61	85	103	109	87	12
104	17	137	86	115	37	90	36	24	121	95
97	15	10	135	46	80	82	52	49	112	82
126	44	13	78	128	110	62	20	99	71	37
64	110	54	115	37	103	112	141	116	86	65
140	103	140	137	27	29	74	4	104	34	101
82	132	136	88	99	94	97	129	88	20	69
131	32	110	74	21	20	43	6	28	99	45
107	117	129	40	49	86	88	98	76	92	39
70	38	47	79	43	32	68	15	64	11	52
34	133	103	64	122	143	102	77	94	106	24
74	144	72	18	75	105	109	74	6	127	6
57	27	28	133	84	27	104	80	78	19	125
42	90	125	140	86	142	81	55	80	131	59
98	79	58	76	89	89	72	18	110	14	61
122	63	38	19	66	129	3	23	132	2	94
83	97	90	63	7	3	100	83	42	126	50
109	88	44	95	129	8	67	40	107	22	60

Roginski MFML Section 4:  $n = 73$  thru 144,  $k = 12$  thru 22 with  $\rho_{map} = 0.0593$ .

Num EMT MM	Num ESU1	Num ESU2	Num ESU3	Num ESU4	Num ESU5	Num Foll1	Num Foll2	Num Foll3	Num Traff1	Num Traff2
102	53	63	37	26	113	103	140	29	136	86
45	45	23	58	33	117	18	60	120	132	8
50	139	66	31	76	87	39	21	44	44	15
72	41	102	98	17	21	1	87	61	61	87
69	134	141	66	35	62	63	113	41	56	112
79	31	59	14	22	63	34	85	90	75	77
1	6	52	77	28	18	6	46	63	50	84
61	13	119	43	56	140	111	49	21	119	137
125	113	144	114	112	34	48	84	117	107	143
24	69	106	59	52	144	23	7	68	95	11
23	54	97	143	131	100	59	97	17	18	111
75	102	126	75	77	48	14	100	51	141	72
112	143	74	30	132	91	52	96	1	28	129
128	20	132	56	78	111	133	121	27	108	7
133	37	3	130	57	76	116	117	103	129	104
6	1	51	57	71	40	95	75	108	68	4
111	10	35	90	19	122	78	106	23	79	90
40	78	67	97	108	23	125	57	79	111	102
65	68	143	61	117	24	29	42	34	88	136
142	42	89	36	60	25	105	59	67	125	19
132	35	34	17	67	78	4	10	143	65	71
88	107	118	27	100	55	139	14	121	26	88
92	87	85	83	113	120	21	72	10	33	51
14	104	50	127	42	47	35	54	97	134	127
63	85	24	99	34	9	77	79	136	122	126
2	123	29	121	98	121	51	19	4	69	122
37	96	64	28	48	42	70	142	119	140	97
54	49	65	89	18	44	22	108	48	84	16
103	33	14	132	1	45	84	67	53	63	79
101	91	99	13	53	73	126	88	54	138	115
100	39	127	32	107	53	73	143	87	3	142
135	28	117	55	15	134	8	131	66	137	44
120	95	107	104	51	74	12	68	14	80	22
113	40	76	46	81	90	17	78	77	49	10
78	50	111	131	59	119	50	45	124	109	54
87	9	100	96	111	39	127	66	118	6	100
26	100	133	49	61	19	49	43	52	128	36
127	3	73	29	40	68	137	38	43	13	25
44	77	80	107	65	125	26	86	12	59	20
119	89	68	120	124	79	57	81	82	144	27
71	119	108	92	10	136	55	91	100	35	106
52	81	94	93	6	30	123	136	45	23	23
11	84	114	102	103	17	89	24	71	96	35
46	114	39	108	92	127	45	39	3	97	133
43	7	101	100	45	59	142	47	95	48	64
32	65	93	62	25	12	93	73	134	113	85
114	34	8	20	83	95	132	128	20	1	131
12	24	128	60	143	50	136	71	101	27	48
33	137	27	7	44	106	118	17	26	24	78
60	111	112	38	140	77	40	102	96	47	80
86	120	56	73	130	41	16	92	15	7	9
47	131	91	70	69	138	54	144	128	91	38
137	135	16	109	30	85	98	51	125	78	56
30	129	45	2	127	28	5	123	129	41	118
49	19	17	22	106	58	134	34	31	116	2
124	125	82	125	101	65	7	2	135	46	42
39	136	12	91	8	123	91	12	111	98	13
8	11	7	53	137	102	33	64	92	143	83
16	14	53	128	85	67	143	115	62	101	70
48	73	138	111	13	46	10	69	75	57	32
51	23	69	26	102	130	113	82	142	36	43
73	121	20	5	95	6	79	89	13	120	81
28	2	104	68	141	116	19	22	55	142	113
22	8	96	103	12	141	101	35	30	94	108
110	93	61	118	94	97	66	61	70	58	31
141	72	75	80	29	124	24	41	133	9	144
90	127	71	123	62	135	61	58	122	8	89
5	99	120	10	11	108	124	26	123	85	140
106	60	79	39	91	112	114	30	18	90	67
38	126	86	117	125	56	92	3	40	64	55
115	80	22	136	136	13	117	95	58	52	46
31	21	98	3	58	72	46	63	98	77	63

Roginski MFML Section 5:  $n = 1$  thru 72,  $k = 23$  thru 33 with  $\rho_{map} = 0.0593$ .

Num Traff3	Num Traff4	Num Traff5	Num Traff6	Num Traff7	Num Traff8	Num Traff9	Num Traff10	Num Traff11	Num Traff12	Desire Civ
143	42	69	6	86	69	26	69	140	6	83
80	122	45	107	130	89	19	111	109	79	110
141	95	48	100	45	41	122	85	16	128	13
69	55	84	127	112	91	116	123	41	45	131
51	101	3	1	18	68	99	127	144	125	71
54	11	39	123	26	48	87	65	82	85	27
96	39	15	16	40	58	121	93	37	44	44
8	90	97	73	58	64	128	143	28	40	29
21	74	26	63	114	119	8	17	87	2	67
53	38	142	72	133	56	70	94	129	115	69
135	136	9	86	87	98	94	28	59	97	51
43	72	114	102	125	34	76	101	2	37	121
3	85	137	80	100	2	55	24	1	68	38
11	128	96	79	54	110	88	41	89	102	15
74	89	1	49	21	44	5	141	133	12	79
123	70	77	11	30	61	54	116	8	41	136
77	135	87	130	52	103	4	42	101	33	129
103	97	6	81	47	32	100	52	31	136	80
125	49	63	62	13	95	82	71	22	17	85
20	77	132	29	102	12	20	55	43	90	10
112	65	107	133	62	57	83	110	30	42	143
84	58	8	9	41	33	96	45	15	51	105
109	81	80	7	77	9	68	72	124	23	4
129	34	81	45	74	27	36	60	23	14	89
48	68	32	134	73	25	40	137	38	36	58
6	41	2	113	116	144	136	15	56	92	111
127	24	131	10	113	24	28	76	75	137	5
23	51	125	139	106	81	3	35	76	66	144
71	141	53	76	7	73	124	49	67	122	33
139	54	138	93	4	23	133	89	105	71	140
136	99	144	44	56	108	72	11	73	98	28
105	5	28	88	55	124	80	135	19	82	14
126	30	75	98	63	13	71	6	92	112	48
64	118	121	54	44	126	135	106	78	15	84
22	69	33	66	136	35	35	112	9	109	9
55	123	111	106	94	120	144	142	126	55	132
31	31	13	117	39	46	50	104	143	121	130
50	108	70	55	131	80	24	12	6	69	127
104	133	16	39	128	82	66	37	80	135	18
138	15	21	84	138	37	109	48	69	61	61
120	137	128	42	65	1	14	80	93	56	109
121	139	85	97	70	51	79	128	68	75	115
73	14	10	50	135	72	29	105	25	63	43
90	18	31	23	107	117	60	13	120	104	64
9	116	59	46	6	5	41	22	36	77	126
44	60	88	109	67	18	11	134	96	11	22
83	125	65	141	12	105	92	118	81	119	34
119	40	139	83	9	76	134	59	79	19	45
82	110	73	135	28	50	63	29	18	89	49
57	82	64	142	48	85	48	114	21	120	133
101	71	105	17	27	11	25	91	94	143	90
93	114	60	24	143	60	143	61	95	139	104
142	35	124	115	93	10	12	16	7	84	25
70	86	104	77	115	90	103	124	74	111	118
75	44	123	56	105	67	119	88	137	28	26
68	115	34	4	78	36	53	2	48	32	31
32	104	94	69	83	99	39	130	4	54	16
107	79	11	143	10	88	9	136	97	21	35
27	98	66	40	124	132	43	79	108	39	82
89	144	20	104	81	66	97	31	42	25	46
110	29	38	92	117	113	137	108	20	132	8
114	12	51	14	119	74	91	30	139	27	54
61	75	134	75	43	102	38	25	132	124	139
144	119	49	70	15	143	1	36	91	116	56
10	4	117	35	24	15	7	86	134	96	11
137	120	140	94	88	93	84	75	11	113	76
115	94	101	22	142	141	102	138	70	34	37
124	96	54	103	66	4	90	122	46	18	12
49	63	122	122	91	40	126	126	62	7	75
45	21	62	65	120	21	78	132	86	106	59
33	103	136	21	111	39	56	70	26	108	23
81	16	93	71	129	6	110	39	90	127	94

Roginski MFML Section 6:  $n = 73$  thru 144,  $k = 23$  thru 33 with  $\rho_{map} = 0.0593$ .

Num Traff3	Num Traff4	Num Traff5	Num Traff6	Num Traff7	Num Traff8	Num Traff9	Num Traff10	Num Traff11	Num Traff12	Desire Civ
58	113	78	59	137	43	106	144	113	58	124
30	76	5	2	60	104	104	131	13	62	98
24	142	47	87	89	20	31	81	66	4	68
76	140	29	91	122	38	57	140	117	64	102
95	91	27	112	92	75	81	27	35	49	41
37	78	108	57	132	137	141	56	121	26	81
60	27	40	116	79	16	75	20	27	100	92
86	87	129	38	20	70	30	82	61	10	73
113	43	82	74	69	115	33	120	142	99	108
63	134	95	52	123	86	89	113	14	43	106
40	59	119	13	49	83	47	119	53	93	134
122	36	98	90	80	59	139	109	130	31	7
85	143	103	8	68	131	51	90	44	30	52
36	45	42	136	46	55	49	38	115	94	112
97	138	55	118	35	84	98	10	128	117	70
56	109	14	105	84	19	114	34	111	88	50
100	57	37	108	16	116	42	99	112	72	72
47	132	74	85	141	22	118	96	98	107	24
79	56	116	128	97	30	62	64	60	20	101
92	66	79	36	139	8	74	58	12	65	125
111	33	89	18	121	114	32	53	54	138	135
62	67	12	51	8	96	120	5	34	1	141
88	10	43	96	96	3	123	133	136	130	88
39	92	110	34	11	135	115	32	40	57	21
42	112	109	37	53	14	108	97	114	142	87
128	127	91	58	38	134	37	84	10	140	95
26	22	18	138	31	87	59	62	45	24	122
13	28	113	33	72	121	10	95	3	80	60
18	17	71	99	57	63	22	78	17	105	96
12	6	25	41	61	136	13	139	141	74	1
108	62	133	78	37	139	16	67	33	129	3
133	52	35	89	95	101	15	47	125	114	17
1	26	67	15	2	65	107	74	65	126	66
118	73	56	5	85	52	67	77	47	81	86
134	107	76	61	140	62	125	7	110	60	113
4	2	19	32	32	78	45	68	32	47	97
65	64	102	95	109	129	58	1	64	131	63
91	124	58	12	144	100	46	51	83	3	78
29	100	7	31	134	112	18	14	118	8	99
17	9	86	140	127	127	142	33	5	101	39
98	111	99	137	14	125	27	26	116	118	2
7	117	126	120	29	28	64	8	122	16	117
67	84	36	121	90	111	127	100	135	70	32
25	3	4	3	34	26	129	4	138	73	103
46	37	127	27	5	140	131	129	58	144	114
140	20	90	124	71	138	65	23	100	46	93
38	19	112	119	126	123	112	21	39	59	55
87	48	141	47	17	54	101	18	99	91	138
72	83	52	101	110	7	44	98	127	52	20
130	102	46	43	51	92	73	107	55	123	120
117	61	143	144	42	49	61	63	88	13	47
131	7	23	25	36	79	86	9	84	5	119
94	32	135	48	23	31	138	44	107	9	36
16	53	120	26	108	133	21	3	119	67	62
78	93	30	131	104	142	77	40	49	29	40
132	80	68	64	103	128	2	102	85	78	142
116	50	24	19	19	130	117	121	24	53	128
28	105	57	60	1	29	105	50	51	103	100
99	13	130	126	76	109	6	125	102	35	91
34	1	92	132	82	106	95	54	63	48	53
14	130	50	129	25	94	52	87	123	76	6
41	131	22	53	64	53	23	66	57	134	65
52	8	100	82	50	71	93	73	52	50	57
102	25	17	111	118	77	113	92	77	83	77
106	23	41	125	33	47	17	83	50	110	107
5	106	72	20	3	42	111	43	71	38	19
66	126	61	114	101	17	140	19	72	87	116
59	88	115	30	75	118	132	117	104	86	74
2	46	118	68	99	97	85	57	131	133	137
35	47	83	28	59	122	69	115	29	95	42
15	129	106	67	22	107	130	46	103	22	30
19	121	44	110	98	45	34	103	106	141	123

Roginski MFML Section 7:  $n = 1$  thru 72,  $k = 34$  thru 45 with  $\rho_{map} = 0.0593$ .

Color	Marksman Pol	Prob Com	Vul Ag	Vul Gun	Marksman Ag	Marksman Gun	Num Civ	Num SWAT	Attr in Orders	Attr in Ag	Eff Bomb
47	9	37	118	140	18	82	14	101	35	95	116
9	99	68	69	80	57	128	7	143	129	23	122
61	75	72	141	131	101	108	73	27	39	119	143
127	133	84	110	97	129	112	84	122	83	65	30
141	101	22	108	109	1	107	31	4	86	53	13
13	129	141	46	118	21	7	39	43	41	69	5
56	14	82	99	126	116	113	21	125	66	4	83
124	58	75	119	107	60	141	87	95	79	90	130
62	15	98	112	89	73	137	76	133	117	111	99
35	108	131	102	59	77	130	47	100	95	52	127
114	130	92	43	108	131	139	83	39	43	139	95
57	40	114	134	7	31	89	123	1	12	138	87
97	66	130	16	106	44	28	29	88	107	134	34
4	68	25	138	52	36	54	82	32	25	105	32
73	121	58	15	67	78	96	125	41	93	121	18
22	13	17	96	34	94	115	91	129	29	39	49
11	116	117	33	99	90	80	52	66	143	141	66
69	50	78	82	24	127	97	13	130	68	26	39
7	74	107	55	21	48	61	38	55	77	88	15
94	22	133	131	53	132	11	9	115	51	55	59
2	81	32	58	136	45	36	143	142	3	46	88
52	71	46	30	35	65	40	112	70	94	125	138
105	117	40	39	46	5	100	26	53	70	129	109
144	140	11	123	36	125	35	103	25	84	44	73
132	57	99	133	62	40	26	132	121	124	102	129
74	84	61	109	83	16	19	108	79	5	122	132
25	12	63	70	17	54	136	64	85	142	135	6
80	34	24	3	65	123	98	81	29	27	107	52
10	37	143	72	71	20	81	95	48	99	99	70
86	88	106	113	105	139	29	50	16	60	104	58
67	26	15	75	95	111	111	115	77	14	1	117
26	109	50	85	122	26	76	43	126	111	82	102
119	52	67	63	22	141	73	77	114	119	137	78
96	48	73	7	13	22	32	92	141	109	84	94
109	72	132	13	128	37	56	27	14	105	36	41
55	29	83	127	29	104	60	6	109	37	110	125
33	25	62	38	40	64	103	128	76	69	50	62
6	76	124	11	129	83	102	35	63	33	33	53
75	6	129	106	63	34	2	133	120	31	24	112
122	61	26	10	45	19	22	138	82	52	7	44
43	118	104	98	111	23	42	23	73	10	41	96
99	67	59	18	120	39	74	79	144	103	127	75
118	80	41	86	132	76	52	106	124	1	117	31
111	45	34	143	26	25	6	66	20	90	63	90
81	8	45	107	9	53	134	130	47	104	59	63
30	73	140	5	77	102	109	144	92	15	103	79
66	28	116	91	124	144	12	134	10	125	115	144
120	16	69	87	138	41	138	93	56	139	32	85
65	131	33	93	103	100	23	63	57	44	108	119
134	39	14	21	96	69	67	4	74	67	133	29
98	114	81	19	74	71	94	10	36	49	136	140
42	63	7	92	139	62	17	122	61	121	126	131
51	83	1	135	86	42	33	102	19	131	22	100
19	106	52	117	56	35	15	40	58	108	2	45
110	65	108	140	68	96	24	62	12	135	86	118
91	132	4	48	93	81	27	124	127	89	124	71
46	69	96	9	5	12	91	45	51	96	81	36
41	20	79	124	8	47	75	60	50	45	98	105
104	60	42	31	19	143	1	118	96	101	42	86
82	36	2	54	54	136	58	20	21	81	66	133
18	96	10	34	44	66	34	11	78	130	61	14
20	92	144	68	115	50	10	97	72	133	106	101
101	126	57	24	90	113	127	121	17	134	20	10
71	143	64	76	14	105	5	111	87	141	78	2
8	119	21	47	137	38	114	110	128	55	75	137
87	135	139	35	58	51	14	127	136	54	76	27
102	56	142	20	49	74	53	70	34	30	58	26
130	144	54	56	33	32	132	69	140	74	17	134
93	139	30	132	73	140	116	131	83	80	13	3
48	138	31	121	78	89	119	65	112	16	97	97
29	124	44	51	15	15	106	119	105	127	62	68
115	43	94	37	23	56	133	142	42	47	101	107

Roginski MFML Section 8:  $n = 73$  thru 144,  $k = 34$  thru 45 with  $\rho_{map} = 0.0593$ .

Color	Marksman Pol	Prob Com	Vul Ag	Vul Gun	Marksman Ag	Marksman Gur	Num Civ	Num SWAT	Attr in Orders	Attr in Ag	Eff Bomb
32	111	29	74	55	68	3	74	38	123	29	115
92	128	39	114	119	122	101	107	40	122	19	139
126	21	103	89	72	10	13	24	52	38	11	21
59	113	55	52	60	95	104	8	11	72	15	89
12	97	137	61	4	85	142	104	33	87	6	50
37	5	70	60	25	33	71	139	91	126	123	47
28	122	5	4	88	97	37	15	118	113	43	69
45	11	23	8	114	121	48	98	67	8	120	55
103	62	101	120	123	112	18	22	119	132	18	20
21	93	125	64	141	52	59	126	15	63	93	61
3	105	71	111	57	88	131	117	108	73	118	56
113	78	123	49	39	99	70	86	60	65	25	37
5	7	48	62	37	107	120	90	5	88	14	111
63	38	36	122	47	108	62	100	132	102	73	33
136	87	128	53	113	27	64	137	59	20	34	74
79	94	9	42	101	80	122	129	69	21	47	54
138	82	113	128	1	7	126	17	139	57	128	82
100	10	38	29	82	124	50	49	18	4	132	28
139	136	95	103	69	115	38	88	86	71	74	12
107	24	102	137	31	61	69	2	30	78	109	7
131	54	121	44	84	4	143	94	131	76	28	110
76	142	118	50	70	14	41	67	54	114	57	93
16	2	60	6	42	43	44	101	7	98	56	84
106	53	43	22	30	29	84	34	104	50	3	114
24	137	93	144	38	98	51	48	110	13	89	16
116	59	90	142	75	55	57	89	23	137	144	104
137	1	134	83	112	2	43	75	80	19	60	92
128	49	109	79	117	70	30	3	35	140	51	72
53	44	65	129	130	120	93	141	3	82	31	25
77	115	87	66	2	110	65	99	9	11	9	48
83	41	119	57	125	119	85	53	138	40	8	51
72	30	89	1	43	135	31	96	137	2	77	142
68	77	13	77	79	117	4	19	81	9	114	67
90	32	127	126	87	82	49	140	116	118	83	57
40	107	135	88	27	92	68	51	106	7	112	22
15	125	138	45	32	126	8	136	98	116	45	124
44	47	47	130	104	46	125	68	107	18	64	17
123	91	8	105	85	106	90	109	31	46	12	23
54	104	136	90	100	137	118	58	62	23	143	77
23	35	16	25	110	59	87	56	68	112	130	42
85	134	100	139	98	30	77	135	22	22	30	24
64	19	66	104	135	63	55	25	103	136	21	128
50	70	126	65	133	134	135	33	99	138	79	43
60	31	122	115	76	128	95	78	94	120	80	8
125	110	110	12	66	84	39	105	113	62	67	60
34	120	49	26	11	103	140	30	8	128	131	120
31	112	111	78	41	142	99	18	45	42	68	136
108	46	112	23	18	6	124	55	28	59	5	91
140	23	105	2	142	130	79	71	89	144	92	40
95	27	27	101	51	86	117	32	75	91	140	11
1	85	53	136	10	93	9	46	71	53	48	121
27	90	88	71	143	11	78	42	24	17	100	4
49	100	97	40	127	58	45	28	90	92	91	81
117	102	18	81	116	114	16	5	65	58	40	123
129	89	12	100	3	75	105	1	134	100	96	46
84	79	3	14	121	3	72	61	13	97	70	80
38	51	80	36	61	138	21	57	117	6	49	65
121	55	91	27	94	109	88	44	37	110	35	64
89	17	6	97	91	8	47	116	93	56	116	9
143	123	51	73	81	24	144	85	44	75	87	106
36	18	20	80	64	49	25	120	84	115	85	19
17	86	115	125	92	67	123	54	111	106	38	108
39	4	76	67	16	91	63	59	2	64	27	103
14	64	74	116	134	9	129	41	26	36	16	35
133	33	85	28	12	118	66	12	46	24	54	141
70	95	28	84	6	72	83	36	97	48	113	1
112	42	86	59	48	79	110	113	135	85	10	76
78	3	35	17	144	133	86	80	64	34	72	98
142	141	56	95	28	28	46	72	123	61	94	113
135	127	120	94	50	87	121	114	6	26	142	126
88	103	19	41	102	17	92	37	102	28	71	38
58	98	77	32	20	13	20	16	49	32	37	135

THIS PAGE INTENTIONALLY LEFT BLANK





THIS PAGE INTENTIONALLY LEFT BLANK

## APPENDIX E. A $\rho_{map}^{\min}$ PREDICTIVE MODEL WITH QUADRATIC TERMS

The final multiple linear regression model to predict  $\rho_{map}^{\min}$  for a given  $n$  and  $k$  is a relatively simple expression:

$$\left(\overline{\rho_{map}^{\min}}\right)^E = 0.0873 + 7.859n^{-2/3} - 0.109k^{-1/3} - 11.702n^{-2/3}k^{-1/3}.$$

It is very tractable in its representation of  $n$  and  $k$ . We see that as  $n$  increases and  $k$  remains constant the first term is dominant in the expression and the mean maximum absolute pairwise correlation decreases, as Owen (1994) theorized. However, when  $k$  also grows large  $\rho_{map}^{\min}$  reduces less quickly because of the values in the second term. These results are reasonable since an increase in  $k$  increases the number of columns, thereby creating more possibility of a new column permutation that may have high correlation with one of the other columns.

Analysis of this model shows that it is adequate for predicting the best  $\rho_{map}^{\min}$  from G200 trials, given  $n$  and  $k$ . Statistics for the residuals indicate that the model is acceptable for its ability to predict. The range of residuals, in comparison with the magnitude of the predicted values, is relatively small.

Figure 15 displays curvature, suggesting that inclusion of quadratic terms may provide an improvement.

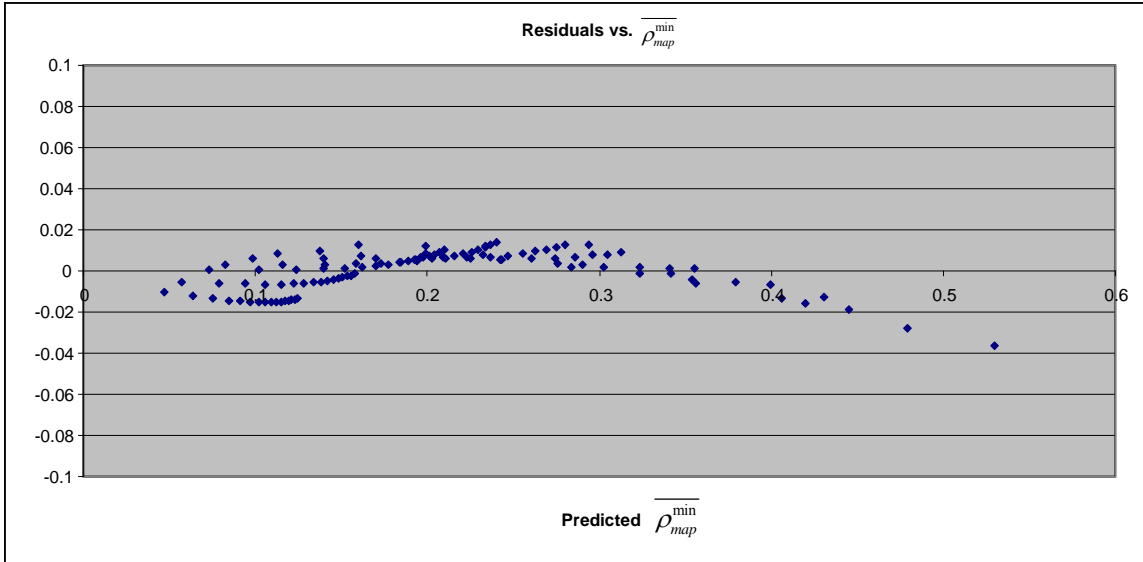


Figure 15. The plot for the residuals resulting from a G200 MLR shows curvature that indicates a need to include more complex terms in the regression model.

We explore the impact of adding more complex terms in the equation to help the model's performance. We extend the model from Chapter II to include quadratic terms for transformed  $n$  and transformed  $k$ . The result is the following eight-term equation with an  $R^2 = 0.9996$ .

$$\begin{aligned} \left(\overline{\rho_{map}^{\min}}\right)^E &= 0.03054 + 0.03208 * k^{-1/3} - 0.100806 * k^{-2/3} + 13.06842 * n^{-2/3} - 68.38078 * n^{-4/3} \\ &\quad - 30.12779 * k^{-1/3} n^{-2/3} + 254.8917 * k^{-1/3} n^{-4/3} + 17.93109 * k^{-2/3} n^{-2/3} - 254.8391 * k^{-2/3} n^{-4/3} \end{aligned}$$

The residual plot (Figure 16) for this new model continues to show curvature, but the sizes of the residuals are a full order of magnitude less than those from the more simple equation.

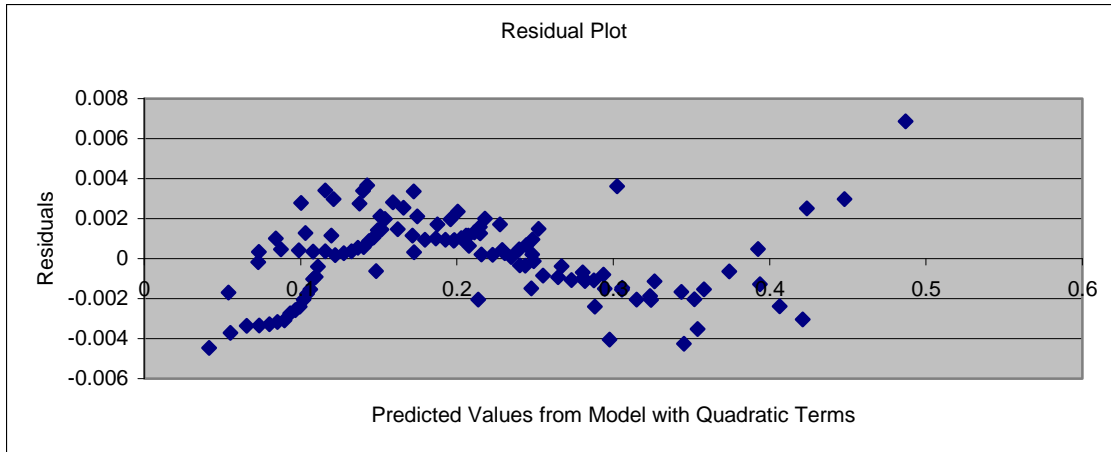


Figure 16. The residual plot for the new multiple linear regression model with quadratic terms still show curvature, but the sizes of the residuals are much smaller than from previous models.

The trade-off with this more precise formula is an increase in difficulty in explaining the impacts of  $n$  and  $k$ , as well as a reduction in the ease in which an experimenter can use it. Although the residual analysis of this multiple linear regression model reveals that there is some curvature in the relationship, the amount of error in the predicted values is relatively small. Since the majority of experimenters would use this formula mostly as a guide to find the appropriate design dimensions and not  $\rho_{map}^{\min}$  itself, the simpler model serves our purposes.

THIS PAGE INTENTIONALLY LEFT BLANK

## INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center  
Ft. Belvoir, Virginia
2. Dudley Knox Library  
Naval Postgraduate School  
Monterey, California
3. Commander  
III Corps  
Fort Hood, Texas
4. Director, TRADOC Analysis Center  
Fort Leavenworth, Kansas
5. Director, Center for Army Analysis  
Fort Belvoir, Virginia
6. Professor Matthew Carlyle  
Department of Operations Research  
Naval Postgraduate School  
Monterey, California
7. Professor Thomas Lucas  
Department of Operations Research  
Naval Postgraduate School  
Monterey, California
8. Professor Craig Rasmussen  
Department of Mathematics  
Naval Postgraduate School  
Monterey, California
9. Professor Paul Sanchez  
Department of Operations Research  
Naval Postgraduate School  
Monterey, California
10. Professor Susan Sanchez  
Department of Operations Research  
Naval Postgraduate School  
Monterey, California

11. Professor Lyn Whitaker  
Department of Operations Research  
Naval Postgraduate School  
Monterey, California
12. COL Thomas M. Cioppa  
TRADOC Analysis Center  
Fort Leavenworth, Kansas
13. Professor Ramesh Kolar  
Computer Sciences Department  
Naval Postgraduate School  
Monterey, California
14. Professor Mikhail Auguston  
Computer Sciences Department  
Naval Postgraduate School  
Monterey, California
15. Mr. Mark Sturm  
National Assessment Group  
Kirtland Air Force Base, New Mexico
16. Mr. John Davis  
National Assessment Group  
Kirtland Air Force Base, New Mexico
17. COL Alejandro S. Hernandez  
Operations Research Department  
Naval Postgraduate School  
Monterey, California