Approximate Model Spaces for Model-Robust Experiment Design*

Byran J. Smucker[1†] and Nathan M. Drew[2]

[1]Department of Statistics

Miami University, Oxford, OH 45056

[2]Analytic Consulting Group

Epsilon, Cincinnati, OH 45242

Abstract

Optimal designs depend upon a prespecified model form. A popular and effective model-robust alternative is to design with respect to a set of models instead of just one. However, model spaces associated with experiments of interest are often prohibitively large and so algorithmically-generated designs are infeasible. Here, we present a simple method which largely eliminates this problem by choosing a small set of models which approximates the full set and finding designs that are explicitly robust for this small set. We build our procedure on a restricted columnwise-pairwise algorithm, and explore its effectiveness for two model spaces in the literature. For smaller full model spaces, we find that the designs constructed with the new method compare favorably to robust designs which utilize the full model space, with construction times smaller by orders of magnitude. We also construct designs that heretofore have been unobtainable due to the size of their model spaces. Supplementary material (available online) includes code, designs, and additional results.

Keywords: $\mathcal{D}$-optimality; supersaturated; two-level designs.

*This is an Author's Accepted Manuscript of an article to be published in *Technometrics* (©Taylor & Francis), and will be available online at `http://www.tandfonline.com`

†corresponding author, `smuckerb@miamioh.edu`

1 Introduction

Optimal design was initially advanced via elegant mathematical theory (Kiefer 1959), but has more recently grown in popularity among practitioners as effective computational methods have been developed (e.g. Meyer and Nachtsheim 1995) and commercial software has added its functionality (e.g. DesignExpert, JMP, Minitab). This popularity is largely due to optimal design's flexibility in handling nonstandard experimental conditions, such as design space constraints, sample size requirements, continuous and categorical factors, and/or nonstandard models.

Along with optimal design comes a serious limitation: an optimal design depends upon an *a priori* assumed model form. This requires the specification of the precise functional relationship between the response and the experimental factors before the experiment is conducted. Since the model is typically unknown, the design constructed will likely be suboptimal for the true model and in some cases may not even allow the estimation of all parameters of interest.

To mitigate this problem, an intuitive, popular, and effective approach is to design for a range of models that the experimenter imagines might be fit. For instance, consider a five-factor experiment for which only 12 runs can be afforded. Classical, fractional factorial designs cannot be used since 12 is not a power of 2. If all main effects and two-factor interactions are active, the model has 16 parameters and is not estimable. However, under the effect sparsity principle (see, for instance, Wu and Hamada 2009), the experimenter may reasonably assume that only a few (say, $g = 4$) of the $c = \binom{5}{2} = 10$ possible two-factor interactions, along with the main effects, will be active. Since it is unknown *which* two-factor interactions might be present, this induces a model space consisting of $\binom{10}{4} = 210$ models. Li and Nachtsheim (2000), restricting themselves to column-balanced designs, found a design that could estimate all but one of the models, while Smucker et al. (2012) relaxed the column-balance requirement and found a design that could estimate all 210.

Though the so-called "main effect plus interaction" (MEPI) model space, as well as others described later, can be used to effectively design experiments when two-factor interactions are suspected but unspecified, the model space itself possesses a crippling handicap: it is often too large. The empirical study of Li et al. (2006) suggests that between 6 and 10 two-factor interactions will be active when studying $k = 12$ factors. (This is an extrapolation of the results of Li et al. (2006) since they only considered full factorial studies of up to seven factors; however, it still may

serve as a rough rule of thumb.) If $g = 6$, the MEPI model space has more than 90 million models; if $g = 10$, it includes more than 210 *billion*. Table 1 illustrates how the size of this model space increases as k and g increase. Other model spaces, such as the projective model space (Loeppky et al. 2007), the supersaturated model space (Jones et al. 2009), or the space including all possible submodels of a specified maximal model, grow intractably large as well.

The size of these model spaces is a limiting factor in their use as model-robust design criteria. Li and Nachtsheim (2000) present no designs for model spaces larger than 400,000 models and do not report design construction times for model spaces with more than 1,000 models. For the supersaturated model space, Jones et al. (2009) indicate that they have limited the size of designs they have produced "because the required computing time can become prohibitively large."

Table 1: MEPI model space sizes for k factors and g active two-factor interactions.

k	g	number of models
6	2	105
6	4	1,365
8	4	20,475
8	6	376,740
10	6	8,145,060
10	8	215,553,195

Here, we effectively overcome the difficulties imposed by large model spaces by developing the idea of an approximate model space, which is a small subset of models taken from the full set. Once this approximate model space is selected we construct designs using only this subset, then evaluate the designs with respect to the full model space. This methodology allows construction of designs associated with large model spaces—such as the aforementioned space which includes more than 210 billion elements—while reducing the construction time in rough commensuration with the factor of reduction between the full and approximate sets.

Using a set of models to construct model-robust design goes back to Läuter (1974). Early work focused primarily on asymptotic designs (e.g. Cook and Nachtsheim 1982; Dette 1990), but more recent work has focused on model spaces for two level designs and small n (Li and Nachtsheim 2000; Loeppky et al. 2007; Jones et al. 2009; Smucker et al. 2012). Bingham and Chipman (2007) use the set of models idea in the context of a Bayesian model-discrimination criterion which places prior probabilities on each model in a given model space. For quadratic-type models, Heredia-Langner et al. (2004), Smucker et al. (2011), and Tsai and Gilmour (2010) have presented ideas, though

only the latter effectively deals with large model spaces. Tsai and Gilmour consider model spaces consisting of specified submodels of a maximal model, using a fast approximation to $\mathcal{A}_s$ optimality, and present an approach which can be used to construct two-level designs, including the model spaces considered in this paper.

This article unfolds as follows. In Section 2, we discuss the problem in more detail, along with the model spaces we consider. In Section 3 we present our procedure, which includes a method for selecting the approximate model space. We present a range of examples in Section 4, along with a discussion of the necessary size of the approximate model space. We conclude with a discussion in Section 5.

2 Problem Definition

Consider an n-run design $\xi_n = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n]'$ in k factors, so that $\mathbf{x}_i$ is $1 \times k$ and ξ_n is $n \times k$. We assume the typical linear regression model, represented as $y_i = f'(\mathbf{x}_i)\boldsymbol{\beta} + \epsilon_i$ where $\boldsymbol{\beta}$ is a p-vector of parameters, f is a $1 \times p$ model function specifying which terms are included in the model (e.g. for $k = 2$, $f(\mathbf{x}_i) = (1, x_{i1}, x_{i2}, x_{i1}x_{i2})$ represents the two-factor interaction model), and the ϵ_i are uncorrelated errors each with mean 0 and variance σ^2. In matrix notation $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$, where $\mathbf{y}$ and $\boldsymbol{\epsilon}$ are n-vectors and $\mathbf{X} = [f(\mathbf{x}_1), f(\mathbf{x}_2), \ldots, f(\mathbf{x}_n)]'$ is an $n \times p$ matrix that expands ξ_n according to f. The least squares estimates are $\hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$ which lead to $Cov(\hat{\boldsymbol{\beta}}) = \sigma^2(\mathbf{X}'\mathbf{X})^{-1}$ and its inverse, the information matrix $\mathbf{M} = \mathbf{X}'\mathbf{X}/\sigma^2$. For the purposes of experiment design, we assume without loss of generality that $\sigma^2 = 1$.

Since $\mathbf{M}$ depends upon the model f through $\mathbf{X}$, the design chosen to optimize a function of $\mathbf{M}$ will depend upon this model. Since f is generally not known when the design is constructed, we consider a set of r models, $\mathcal{F} = (f_1, f_2, \ldots, f_r)$, and ideally would choose a design which optimizes a function of this set. Such model spaces are experiment-specific, and we consider two in Section 2.2 that are relevant for main effects or two-factor interaction models.

2.1 Measures of Model Robustness

For general $\mathcal{F}$, there is no guarantee that an arbitrary design can estimate all models. *Estimation capacity* (Sun 1993; Li and Nachtsheim 2000) quantifies a design's ability to estimate the models

in $\mathcal{F}$:

$$EC = \frac{\text{Number of estimable models}}{\text{Number of models in } \mathcal{F}}.$$

For many model spaces, a design can be found for which $EC = 1$. In this case, a secondary criterion may be employed to improve the efficiency of the design. Here, we follow the literature (Li and Nachtsheim 2000; Loeppky et al. 2007; Jones et al. 2009; Smucker et al. 2012) and use the $\mathcal{D}$-criterion, though others could be used. The $\mathcal{D}$-criterion, which minimizes $|Cov(\hat{\boldsymbol{\beta}})|$, has computational advantages while also being invariant to linear transformations of $\mathbf{X}$. Define $\mathcal{D}$-efficiency for model f as $\left(\frac{|\mathbf{M}_f(\xi_n)|}{|\mathbf{M}_f(\xi_n^*)|}\right)^{1/p_f}$, where $\mathbf{M}_f(\xi_n)$ is the information matrix for model f and design ξ_n, ξ_n^* is the n-run design optimal for model f, and p_f is the number of parameters for model f. Instead of finding ξ_n^* for every $f \in \mathcal{F}$, we use a lower bound approximation to the $\mathcal{D}$-efficiency

$$E_f(\xi_n) = \left(\frac{|\mathbf{M}_f(\xi_n)|}{n^{p_f}}\right)^{1/p_f} = \frac{|\mathbf{M}_f(\xi_n)|^{1/p_f}}{n}, \tag{1}$$

which compares $|\mathbf{M}_f(\xi_n)|$ to a hypothetical orthogonal design whose determinant is n^{p_f}. This leads to what Li and Nachtsheim (2000) call *information capacity* (IC), which is simply the average $\mathcal{D}$-efficiency over all models in $\mathcal{F}$. Note that the IC can be computed even if some models are not estimable, in which case the inestimable models have a $\mathcal{D}$-efficiency of 0.

2.2 Model Spaces

We now briefly describe, for a given model space, the notion of *changeable elements*. A changeable element for a model space $\mathcal{F}$ is a model term whose presence may vary from model to model. The number of changeable elements for a particular model space will be denoted by c. We will consider two model spaces below.

If an experimenter wishes to design a two-level experiment that will allow two-factor interactions to be estimated, standard fractional factorial designs can be used. However, if a resolution IV design is performed and nature decides that two aliased two-factor interactions are simultaneously active, the model will not be estimable. Resolution V designs resolve estimation problems for two-factor interactions but come at the price of many runs. As an alternative, Sun (1993) and

Li and Nachtsheim (2000) framed the "main effects plus interactions" (MEPI) model space that consisted of all models with k main effects and g two-factor interactions. Since there are $c = \binom{k}{2}$ possible two-factor interactions, the space includes $r = \binom{c}{g}$ models. Thus, a two-level design which maximizes EC with respect to this model space tends to have better estimability properties than resolution IV designs with the same number of runs. Often, such designs are not orthogonal and, depending on the algorithm, lack column balance. Note that g might be chosen as an upper bound on the number of two-factor interactions that are expected, since there is evidence that these designs perform well if in fact fewer than g terms are active (Li and Nachtsheim 2000). For this model space, the c two-factor interactions are the changeable elements, whereas the main effects appear in each model and thus are not changeable.

Supersaturated designs are those for which $k \geq n$, and thus these designs cannot estimate all main effects simultaneously. We also consider the supersaturated (SS) model space (Jones et al. 2009), which is the set of all models which include $g < n$ out of the k main effects. This implies that the number of models in the space is $r = \binom{k}{g}$. These designs are two-level and effect sparsity is a key assumption. The goal is to construct a design that maximizes the EC, and if $EC = 1$ can be reached, find the design that maximizes the information capacity. In this case, $c = k$; that is, all main effects are changeable elements.

Though we have not done extensive testing for other model spaces, the general framework presented in the following section could be more broadly applied. For instance, the projective model space (Loeppky et al. 2007) includes all models for which a subset of the main effects, and all associated two-factor interactions are active. One might also consider the model space consisting of all possible submodels of a defined maximal model. Variations of this could include only submodels of the maximal model that respect effect heredity or strong heredity (see Wu and Hamada 2009).

3 Methodology

To avoid the computational bottleneck accompanying large model spaces, we propose the following three-step procedure:

1. Select a small sample of s_1 models from the full model space, $\mathcal{F}$. This is the *approximate*

model space, denoted by $\mathcal{S}_1$, and is chosen to be close to balanced in a way that will be made specific in Section 3.1.

2. Construct a number of designs that are robust for the models in $\mathcal{S}_1$.

3. Evaluate these designs with respect to the models in $\mathcal{F}$. If the size of $\mathcal{F}$ precludes quick evaluation of all models, take a sample, $\mathcal{S}_2$, of s_2 models and evaluate the designs with respect to this set. The design that performs the best with respect to $\mathcal{F}$ (or $\mathcal{S}_2$) is chosen.

The rest of this section gives more details regarding these three steps.

3.1 Selecting the Approximate Model Space

To implement this procedure, we must select a relatively small subset of models, $\mathcal{S}_1$, from the full model space. We will call the selection of $\mathcal{S}_1$ the meta-design problem. Intuitively, we would like to do this in a balanced way so that the total number of times each changeable element appears in all the models in $\mathcal{S}_1$ is about the same. Furthermore, we might wish to ensure that each pair of changeable elements appears together in a model in $\mathcal{S}_1$ roughly the same number of times.

Viewing the selection of models in this way calls to mind balanced incomplete block designs (BIBDs) in which the levels of a single categorical factor must be assigned to several blocks so that each pair of treatments appears together in a block the same number of times (see, for instance, Dean and Voss 1999). Consider:

1. The number of changeable elements, c (e.g. the number of 2fi's in the MEPI model space), is analogous to the number of treatments in a BIBD.

2. The number of models in $\mathcal{S}_1$, s_1, is analogous to the number of blocks in a BIBD.

3. The number of changeable elements in each model, g, is analogous to the size of the blocks in a BIBD.

Clearly, most combinations of c, s_1 and g will not admit a BIBD. However, BIBDs have been shown to be $\mathcal{D}$-optimal (Kiefer 1975), so $\mathcal{D}$-optimal designs for experiments with blocks (see Cook and Nachtsheim 1989) will find designs that are close to BIBDs, if possible.

Let N_d be the incidence matrix for the meta-design. That is, N_d is $c \times s_1$ with a 1 for entry ij if changeable element i appears in model j, and 0 otherwise. Then, $N_d N_d^{'}$ is called the concurrence matrix and its i^{th} diagonal gives the number of times changeable element i appears in the approximate model space, and the ij^{th} off-diagonal element gives the number of time changeable element i appears together in a model with changeable element j. If the meta-design were a BIBD, each diagonal would be $r = \frac{s_1 g}{c}$ and all of the off-diagonals would be exactly $\lambda = \frac{s_1 \binom{g}{2}}{\binom{c}{2}}$ where r and λ are both integers. Even if these quantities are not integers, we can still calculate them and construct the information matrix for the treatments. The form can be inferred from Cheng and Bailey (1991) to be $M_{\text{trt,BIBD}} = rI - C/g$ where $C = rI + \lambda K$ is the concurrence matrix and K is a matrix with 0's on diagonals and 1's off-diagonals. To make this matrix invertible, we eliminate the last row and column so that $M_{\text{trt,BIBD}}$ is a $c-1 \times c-1$ matrix.

Let X_1 be the expanded design matrix for the treatments/changeable elements, and X_2 be the expanded design matrix for the blocks/models in the approximate model space, where these matrices can be parameterized using the standard reference coding. For any meta-design, its treatment information matrix can be computed as $M_{trt} = X_1' Q_2 X_1$, where Q_2 is the orthogonal projector onto the subspace spanned by the columns of X_2. Then for any meta-design, we can calculate a measure of block $\mathcal{D}$-efficiency (SAS 2013), compared to a theoretical BIBD:

$$B = \left(\frac{|M_{\text{trt}}|}{|M_{\text{trt,BIBD}}|} \right)^{1/c}.$$

We call meta-designs constructed to be block $\mathcal{D}$-optimal nearly balanced incomplete block designs (NBIBDs), and used SAS software's PROC OPTEX to generate them. Table 2 gives the results from several NBIBD meta-designs for the MEPI model space, and compares them to meta-designs generated by randomly assigning changeable elements to models. See also Figure 1, and note how close the NBIBD is to a BIBD, and how far the random meta-design is from this ideal. In Section 4 we explore results for $s_1 = \{16, 32, 64, 128, 256\}$, for the NBIBD meta-designs.

An operational detail bearing mention: when dealing with very large model spaces, even constructing the full model space is computationally prohibitive, so in the Appendix we outline a straightforward procedure that maps a number between 1 and r to a model in $\mathcal{F}$, without explicitly

Table 2: Block $\mathcal{D}$-efficiency for MEPI meta-designs.

k	g	s_1	Meta-design	Block $\mathcal{D}$-eff.
5	4	16	NBIBD	0.9962
			Random	0.9549
8	5	64	NBIBD	0.9978
			Random	0.9693
10	7	64	NBIBD	0.9982
			Random	0.9495
12	10	128	NBIBD	0.9974
			Random	0.9370

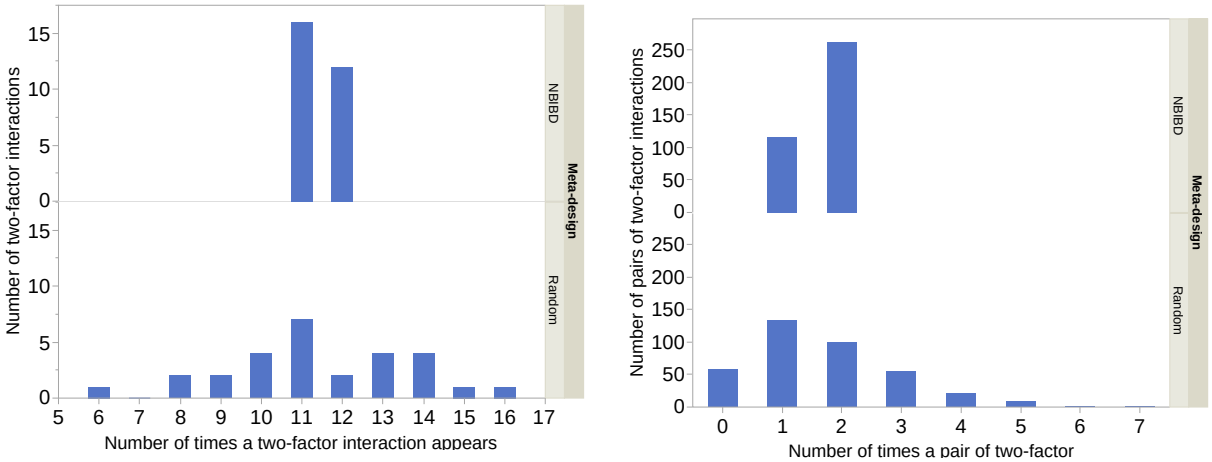

Figure 1: NBIBD vs. random meta-designs, for the MEPI $\{k = 8; g = 5; s_1 = 64\}$ experiment, in terms of the number of times each two-factor interaction appears in the approximate model space (left) and in terms of the number of times each pair of two-factor interactions appear together in the same model (right).

constructing the full model space. Though the Appendix focuses on the MEPI model space, a similar approach could be used for the supersaturated model space and others whose elements can be counted systematically.

3.2 Constructing a Design Based on the Approximate Model Space

Once $\mathcal{S}_1$ has been selected standard algorithms can be used to construct a design that is robust to this approximate model space. In this article, we focus primarily on the restricted columnwise-pairwise (RCP) algorithm of Li and Nachtsheim (2000), to make comparisons with designs given in the literature. This algorithm ensures that the resulting design will be column-balanced, though this restriction limits its performance in terms of estimation and information capacity. We have also extensively explored another algorithm, similar to that of Smucker et al. (2012), which is based on coordinate exchange (Meyer and Nachtsheim 1995). This procedure removes the balance require-

ment and thus allows for better estimation and information properties. We provide an overview of the RCP algorithm, but relegate a description of the coordinate exchange-based algorithm to the supplementary material.

We assume that $\mathcal{F}$ (whether MEPI, supersaturated or another model space for two-level designs) has been determined by the experimenter, and that s_1 models have been selected from $\mathcal{F}$ constituting $\mathcal{S}_1$ (see Section 3.1). Note also that the designs constructed via these algorithms are only explicitly robust for the models in $\mathcal{S}_1$. Their efficacy with respect to the models in $\mathcal{F}$ is considered in the third step of the procedure (Section 3.3) and more specifically in Section 4. For numerical stability, determinants are calculated on the log scale.

The restricted columnwise-pairwise algorithm of Li and Nachtsheim (2000) proceeds column by column through the design, exchanging coordinates within a given column. Assuming a column-balanced design—that is, each column has equal numbers of 1s and -1s—considering all $\binom{n}{n/2}$ possible balanced orderings within a column, or even all $(n/2)^2$ possible swaps becomes computationally prohibitive. Instead, for each column, an element is randomly selected and swaps with all other elements of opposite sign are considered. The exchange that most increases the estimation capacity is executed. If two or more exchanges tie in terms of estimation capacity, the tie is broken using information capacity. Though the algorithm follows Li and Nachtsheim (2000) fairly closely, there may be some differences in implementation.

Algorithm 3.1 The algorithm is as follows, and is performed n_t times.

1. Initialize the algorithm by constructing a randomly chosen, balanced initial design.
2. Improve the design by maximizing $EC_{\mathcal{S}_1}$ and $IC_{\mathcal{S}_1}$.
   (a) Iteratively for each column, randomly select an element within the column.
   (b) Consider exchanges with all other column elements of opposite sign, computing $EC_{\mathcal{S}_1}$ and $IC_{\mathcal{S}_1}$ for each possible exchange.
   (c) If an exchange results in a higher $EC_{\mathcal{S}_1}$, or the same $EC_{\mathcal{S}_1}$ with a higher $IC_{\mathcal{S}_1}$, execute the exchange and update the design, its information matrix for all models in $\mathcal{S}_1$, the current $EC_{\mathcal{S}_1}$, and the current $IC_{\mathcal{S}_1}$.
   (d) Continue until an iteration has elapsed without any improvement.

This process is repeated an experimenter-specified number of times, n_t, each from a randomly generated initial design.

3.3 Relating the Design to the Full Model Space

If the number of models r is relatively small—say a few thousand—the n_t designs generated in Section 3.2 can each be evaluated with respect to all models in $\mathcal{F}$, and the design that is best in terms of estimation capacity and/or information capacity with respect to $\mathcal{F}$ can be selected as the design of choice. This is the typical approach employed in previous research (Li and Nachtsheim 2000; Loeppky et al. 2007; Jones et al. 2009; Smucker et al. 2012).

However, if r is large—say a million or a billion—evaluating even one design with respect to all models in $\mathcal{F}$ is computationally prohibitive. Thus, we randomly sample from $\mathcal{F}$, calling this sampled set of models $\mathcal{S}_2$ and evaluate each design with respect to $\mathcal{S}_2$ ($EC_{\mathcal{S}_2}$ and/or $IC_{\mathcal{S}_2}$). We can also utilize standard statistical procedures for inference on $EC_{\mathcal{F}}$ and/or $IC_{\mathcal{F}}$, but we consign those details and associated results to Supplementary Material A.1. The number of models in $\mathcal{S}_2$, s_2, should be large enough to produce narrow confidence intervals (CIs), but small enough to pose few computational demands. We have satisfactorily used $s_2 = 2{,}000$ throughout.

4 Empirical Results

In this section we test the approximate model spaces methodology using the RCP procedure given in Algorithm 3.1, using both the SS and MEPI model spaces. For smaller model spaces we compare the designs based on our procedure (call them $\mathcal{S}$-designs) to those based upon $\mathcal{F}$, and show that the $\mathcal{S}$-designs are competitive, and sometimes superior, in terms of $\mathcal{F}$-robustness. We also provide designs for large $\mathcal{F}$ that have previously been unobtainable, and make recommendations regarding the necessary size of the approximate model space.

4.1 Examples

As stated in the previous section, we give results for $s_1 = \{16, 32, 64, 128, 256\}$ to examine the effects of the size of $\mathcal{S}_1$ on the effectiveness of the procedure. In all cases we measure estimation capacity and information capacity with respect to the post-sample of models $\mathcal{S}_2$ (i.e. $EC_{\mathcal{S}_2}$ and $IC_{\mathcal{S}_2}$). For the $\mathcal{S}$ designs, we use $s_2 = 2{,}000$ when $r > 10{,}000$; $\mathcal{S}_2 = \mathcal{F}$ otherwise. $\mathcal{F}$ designs whose results are displayed use $\mathcal{S}_2 = \mathcal{F}$, in keeping with previous work. Unless otherwise noted, all designs in this section were constructed using our implementation of the RCP algorithm and use $n_t = 100$.

When $\mathcal{F}$ is large and $s_2 < r$, we can calculate appropriate CIs for $EC_{\mathcal{F}}$ and $IC_{\mathcal{F}}$. For the sake of space they are omitted from the results in this section, but are included in Supplementary Material A.1. All approximate model spaces are chosen to be near balanced incomplete block designs (NBIBD) in the sense described in Section 3.1.

Note that instead of taking the best meta-design and using it to perform n_t algorithm tries, for the same amount of computational effort we could have taken the top b meta-designs and used them each to perform n_t/b algorithm tries. We explored this approach with the experiments detailed in this section, using $n_t = 100$ and $b = 5$, and the results were not statistically significantly different from using a single meta-design.

4.1.1 Supersaturated Model Space

We first explore results for the supersaturated (SS) model space (see Section 2.2), applying the methodology of Section 3 directly. The SS model space was originally presented by Jones et al. (2009), who computed so-called "Model-Robust Supersaturated" (MRSS) designs using the RCP algorithm and the full model space.

For the experiments with smaller $\mathcal{F}$, we focus primarily on three MRSS experiments from Jones et al. (2009): $\{n = 6; k = 10; g = 3\}$, $\{n = 8; k = 12; g = 5\}$, and $\{n = 10; k = 15; g = 6\}$; see Table 3(a). For the given combinations of n and k, Jones et al. (2009) found these values of g to be the largest for which $EC_{\mathcal{F}} = 1$.

Edwards and Mee (2011) warn that in order for a supersaturated design to be successful, there must be extreme effect sparsity in which most of the variability in the data is due to just a few, additive effects. Marley and Woods (2010) give a specific recommendation that there should be at least three times as many runs as active main effects. We give results in Table 3 for some supersaturated experiments in which g is close to this recommendation ($\{n = 14; k = 23; g = 5\}$; $\{n = 16; k = 30; g = 6\}$), as well as experiments of the same sizes with larger g. The largest experiment has a ratio of runs to active effects of about 2.5 instead of 3, but note an experimenter is well-advised to take g as an upper bound on the number of active main effects expected, because based upon results in similar studies (Li and Nachtsheim 2000; Loeppky et al. 2007), the designs should perform well even if fewer than g are active.

Table 3 shows that generally the larger the approximate model space the better the result with

respect to $\mathcal{F}$, and for all experiments represented in this table, $s_1 = 64$ produces a design that is within a half of a percent of the best design found. Note that for the designs in Table 3(b), designs robust to $\mathcal{F}$ are computationally infeasible.

Table 3: Results for SS designs constructed with the RCP algorithm for (a) smaller and (b) larger model spaces. n is the number of runs; k is the number of factors; g is the specified number of active main effects; and r is the number of models in $\mathcal{F}$. The tables reference $\mathcal{S}$-x designs, where x refers to s_1. All $\mathcal{S}$ designs are based upon NBIBDs and for all experiments for which $r < 10{,}000$, $s_2 = r$.

(a) Smaller model spaces

n	k	g	r	Design	$EC_{\mathcal{F}}$	$IC_{\mathcal{F}}$
6	10	3	120	MRSS	1	0.903
				$\mathcal{S}$-16	1	0.903
				$\mathcal{S}$-32	1	0.903
				$\mathcal{S}$-64	1	0.903
8	12	5	792	MRSS	1	0.817
				$\mathcal{S}$-16	1	0.817
				$\mathcal{S}$-32	1	0.817
				$\mathcal{S}$-64	1	0.817
				$\mathcal{S}$-128	1	0.817
				$\mathcal{S}$-256	1	0.817
10	15	6	5,005	MRSS	1	0.823
				$\mathcal{S}$-16	0.9998	0.827
				$\mathcal{S}$-32	0.9998	0.829
				$\mathcal{S}$-64	0.9998	0.829
				$\mathcal{S}$-128	0.9998	0.830
				$\mathcal{S}$-256	0.9998	0.830
14	23	5	33,649	$\mathcal{S}$-16	1	0.904
				$\mathcal{S}$-32	1	0.923
				$\mathcal{S}$-64	1	0.923
				$\mathcal{S}$-128	1	0.924
				$\mathcal{S}$-256	1	0.925

(b) Larger model spaces

n	k	g	r	Design	$EC_{\mathcal{S}_2}$	$IC_{\mathcal{S}_2}$
14	23	12	1,352,078	$\mathcal{S}$-16	0.9990	0.623
				$\mathcal{S}$-32	0.9980	0.608
				$\mathcal{S}$-64	0.9980	0.632
				$\mathcal{S}$-128	0.9990	0.629
				$\mathcal{S}$-256	0.9985	0.634
16	30	6	593,775	$\mathcal{S}$-16	1	0.884
				$\mathcal{S}$-32	1	0.901
				$\mathcal{S}$-64	1	0.905
				$\mathcal{S}$-128	1	0.906
				$\mathcal{S}$-256	1	0.907
16	30	14	1.454e08	$\mathcal{S}$-16	0.9995	0.581
				$\mathcal{S}$-32	1	0.590
				$\mathcal{S}$-64	1	0.593
				$\mathcal{S}$-128	1	0.592
				$\mathcal{S}$-256	1	0.594
24	45	10	3.190e09	$\mathcal{S}$-16	1	0.861
				$\mathcal{S}$-32	1	0.874
				$\mathcal{S}$-64	1	0.881
				$\mathcal{S}$-128	1	0.884
				$\mathcal{S}$-256	1	0.884

4.1.2 RCP Algorithm with MEPI Model Space

The "main effects plus interaction" (MEPI) model space was described in Section 2.2. Here we give a variety of examples using the RCP algorithm, and for designs with smaller model spaces we compare the designs constructed by the proposed methodology (i.e. $\mathcal{S}$ designs) with the Model Robust Factorial Designs (i.e. $\mathcal{F}$ designs) of Li and Nachtsheim (2000). For experiments with larger model spaces, we present results for designs that have been practically unobtainable until now. Supplementary Material A.1 includes these results with appropriate confidence intervals.

From Table 4 it is clear the $\mathcal{S}$ designs are competitive with the $\mathcal{F}$ designs and in fact that the $\mathcal{S}$ designs sometimes significantly outperform the designs based upon the full model space. We note that for the $\{n = 16, k = 8, g = 5\}$ and $\{n = 16, k = 9, g = 5\}$ experiments, results could not be obtained due to outsized computational and memory requirements. Thus the results are taken

directly from Li and Nachtsheim and the fact that they only used 20 algorithm tries may explain some of the discrepancy between these designs and the $\mathcal{S}$ designs.

Except for the first experiment, there are no benchmarks for the designs represented in Table 4(b). We simply obtain the designs and observe how the results change for various sizes of $\mathcal{S}_1$. Note that adding even a few additional runs can produce a pleasing increase in model-robustness. Consider $\{n = 18; k = 10; g = 7\}$. The designs associated with this experiment fail to estimate around 4% of the models and have an IC around 50%. Adding just two runs results in designs with 100% EC and an increase in IC of around 18 percentage points.

Unlike for the SS model space, there is often a lack of monotonicity in the results as a function of the size of $\mathcal{S}_1$. For instance, for $\{n = 18; k = 10; g = 7\}$ the $\mathcal{S}$-256 design has a lower IC than any of the other designs. This suggests that given the approximate model spaces specified in this article, it may be desirable to construct designs using several approximate model space sizes. Additional discussion is deferred to the following section.

4.2 The size and nature of the approximate model space

A natural question, after observing the results in Sections 4.1.1 and 4.1.2, is: How large should one choose $\mathcal{S}_1$ to be? For the supersaturated model space, the question is simplified by the observation that an increase in s_1 typically results in an increase in EC and IC. Based on Table 3, we recommend that $s_1 = 64$ since it appears that this produces results very close to the best observed. Even for the largest experiment, $\{n = 25; k = 45; g = 10\}$, $s_1 = 64$ produces a reduction from the best observed of only about 0.3%. For the SS results in Supplement A.2, using the algorithm based on coordinate exchange, the recommendation is similar though we find a slightly larger reduction when using $s_1 = 64$—up to about 1%.

For the MEPI model space, the question is complicated by the fact that sometimes using a larger approximate model space does not yield a better design. We emphasize that when compared to designs based upon the full model space, the $\mathcal{S}$ designs appear competitive for any reasonable size of $\mathcal{S}_1$. If one constructs designs for each of $s_1 = \{16, 32, 64\}$ for a particular experiment and takes the best of the three, the result will typically be close to the best observed by running $s_1 = \{16, 32, 64, 128, 256\}$. Note that to construct designs based on $s_1 = \{16, 32, 64\}$ takes about as long as constructing the design based upon $s_1 = 128$ alone and about twice as long as using $s_1 = 64$

Table 4: Results for MEPI designs constructed with the RCP algorithm and (a) smaller and (b) larger model spaces. n is the number of runs; k is the number of factors; g is the specified number of active main effects; and r is the number of models in $\mathcal{F}$. The tables reference $\mathcal{S}$-x designs, where x refers to s_1. All $\mathcal{S}$ designs are based upon NBIBDs and for all experiments for which $r < 10{,}000$, $s_2 = r$. For experiments with $r > 10{,}000$, $s_2 = 2{,}000$ for $\mathcal{S}$ designs and $s_2 = r$ for MRFDs.

(a) Smaller model spaces

n	k	g	r	Design	EC_F	$IC_{\mathcal{F}}$
12	7	3	1,330	MRFD	0.9940	0.661
				$\mathcal{S}$-16	0.9910	0.663
				$\mathcal{S}$-32	0.9925	0.667
				$\mathcal{S}$-64	0.9917	0.694
				$\mathcal{S}$-128	0.9925	0.663
				$\mathcal{S}$-256	0.9940	0.662
12	7	4	5,985	MRFD	0.8989	0.517
				$\mathcal{S}$-16	0.8777	0.504
				$\mathcal{S}$-32	0.8897	0.512
				$\mathcal{S}$-64	0.8999	0.539
				$\mathcal{S}$-128	0.8989	0.517
				$\mathcal{S}$-256	0.9086	0.536
16	7	3	1,330	MRFD	1	0.888
				$\mathcal{S}$-16	1	0.886
				$\mathcal{S}$-32	1	0.888
				$\mathcal{S}$-64	1	0.888
				$\mathcal{S}$-128	1	0.888
				$\mathcal{S}$-256	1	0.888
16	10	3	14,190	MRFD	1	0.686
				$\mathcal{S}$-16	1	0.750
				$\mathcal{S}$-32	1	0.756
				$\mathcal{S}$-64	1	0.767
				$\mathcal{S}$-128	1	0.767
				$\mathcal{S}$-256	1	0.762
16	8	5	98,280	MRFD*	0.9986	0.667
				$\mathcal{S}$-16	0.9995	0.709
				$\mathcal{S}$-32	0.9985	0.747
				$\mathcal{S}$-64	0.9985	0.742
				$\mathcal{S}$-128	0.9995	0.739
				$\mathcal{S}$-256	0.9985	0.727

* This design is based upon 20 algorithm tries and taken from (Li and Nachtsheim 2000).

(b) Larger model spaces

n	k	g	r	Design	$EC_{\mathcal{S}_2}$	$IC_{\mathcal{S}_2}$
16	9	5	376,992	MRFD*	0.9801	0.597
				$\mathcal{S}$-16	0.9975	0.644
				$\mathcal{S}$-32	0.9955	0.662
				$\mathcal{S}$-64	0.9960	0.658
				$\mathcal{S}$-128	0.9950	0.632
				$\mathcal{S}$-256	0.9950	0.625
16	10	5	1,221,759	$\mathcal{S}$-16	0.9710	0.516
				$\mathcal{S}$-32	0.9760	0.519
				$\mathcal{S}$-64	0.9590	0.491
				$\mathcal{S}$-128	0.9650	0.498
				$\mathcal{S}$-256	0.9705	0.489
18	10	7	4.538e7	$\mathcal{S}$-16	0.9660	0.511
				$\mathcal{S}$-32	0.9615	0.490
				$\mathcal{S}$-64	0.9545	0.476
				$\mathcal{S}$-128	0.9580	0.488
				$\mathcal{S}$-256	0.9620	0.475
20	10	7	4.538e7	$\mathcal{S}$-16	1	0.677
				$\mathcal{S}$-32	1	0.687
				$\mathcal{S}$-64	1	0.680
				$\mathcal{S}$-128	1	0.702
				$\mathcal{S}$-256	1	0.686
24	12	10	2.110e11	$\mathcal{S}$-16	1	0.570
				$\mathcal{S}$-32	1	0.574
				$\mathcal{S}$-64	1	0.592
				$\mathcal{S}$-128	1	0.597
				$\mathcal{S}$-256	1	0.620

* This design is based upon 20 algorithm tries and taken from (Li and Nachtsheim 2000).

only. For larger designs, most notably $\{n = 24; k = 12; g = 10\}$, using $s_1 = \{16, 32, 64\}$ is inferior to the best observed by a larger margin. Thus, we recommend using, for MEPI experiments with larger model spaces, at least $s_1 = \{16, 32, 64, 128\}$ while for smaller model spaces $s_1 = \{16, 32, 64\}$ is adequate.

We limit our recommendations to the model spaces and experiment sizes explored in this paper, though they might be a good starting point for larger experiments, or experiments based upon other model spaces.

Another natural question is whether choosing the approximate model space as an NBIBD improves over the results obtained by choosing the subset of models at random. To explore this, several of the experiments from Tables 3 and 4 were chosen and designs constructed for

$s_1 = \{16, 32, 64, 128, 256\}$ with a randomly selected approximate model space. The resulting differences in EC and IC are given in the Table in Supplementary Material B. For neither the SS nor the MEPI model space are there striking differences.

The NBIBD method is cleaner in the sense that there is less concern about choosing multiple sets as a hedge against a bad random sample. We do note that there are many nearly balanced incomplete block designs, so that the NBIBD used is not necessarily the single, best sample. The approach does, however, provide a principled and intuitive approach to the meta-design problem that may serve as a foundation or baseline for other approaches. On the other hand, the NBIBD takes longer to construct, though relative to the model-robust design construction time it is not a heavy computational burden for the experiments explored here.

Note that there are other ways that models might be chosen from the approximate model space. For instance, a model might be represented as a binary string of length equal to the number of changeable elements in the model with 1s and 0s representing terms that are present and absent. Then, Euclidian or Hamming distance could measure the distance between any two models in a model space (e.g. MEPI or SS), and $\mathcal{S}_1$ could be generated to, say, maximize the minimum distance between the models. An implementation of an algorithm similar to that of Jin et al. (2005) was explored, but its results did not appear to be markedly better than those from designs obtained via random selection of $\mathcal{S}_1$.

To briefly illustrate the time savings realized by our algorithm, as well as times for large model spaces, we present some timing results in Table 5. The results are given in terms of average minutes per algorithm try. We follow the recommendations from Section 4.2: For the SS model space we give results for $s_1 = 64$ (though we include results for $s_1 = 16$ for the larger experiments for comparison purposes), and for the MEPI model space we give total times for $s_1 = \{16, 32, 64\}$ or $s_1 = \{16, 32, 64, 128\}$, depending on the size of the model space. The computation was performed on Linux cluster nodes equipped with dual quad-core Intel(R) Xeon(R) 2.26GHz E5520 CPUs. Each node has shared memory of 24GB.

This methodology allows a trade-off between quality and computational efficiency. Relatively "good" designs can be obtained, even for the largest experiments, by using a smaller approximate model space. For instance, for the MEPI experiment $\{n = 24; k = 12; g = 10\}$, there is a reduction of about 7% in IC when using $s_1 = 32$, compared to the best observed in Table 4(b). But this $\mathcal{S}$-32

design can be obtained in less than 10% of the time it takes to construct all of $s_1 = \{16, 32, 64, 128\}$.

Table 5: The time to construct several model-robust designs constructed with the RCP algorithm, in terms of average minutes per algorithm try for (a) supersaturated model space and (b) the MEPI model space. For (b), the times represent the sum of the times for each of the $\mathcal{S}$ designs indicated in the Designs column.

(a) Supersaturated model space

n	k	g	r	Design	Time
8	12	5	792	$\mathcal{S}$-64	0.4
				MRSS	3.3
10	15	6	5,005	$\mathcal{S}$-64	0.8
				MRSS	47.1
14	23	12	1,352,078	$\mathcal{S}$-16	1.0
				$\mathcal{S}$-64	3.5
24	45	10	3.190e09	$\mathcal{S}$-16	7.3
				$\mathcal{S}$-64	21.1

(b) MEPI model space

n	k	g	r	Designs	Time
12	7	3	1,330	$s_1 = \{16, 32, 64\}$	0.8
				MRFD	8.5
12	7	4	5,985	$s_1 = \{16, 32, 64\}$	1.0
				MRFD	36.6
16	9	5	376,992	$s_1 = \{16, 32, 64\}$	2.4
20	10	7	4.538e07	$s_1 = \{16, 32, 64, 128\}$	9.9
24	12	10	2.110e11	$s_1 = \{16, 32, 64, 128\}$	22.0

5 Discussion

In this article we consider a generalization of optimal design in which a set of models is considered instead of just one. We present methodology to construct model-robust experiment designs much more quickly than comparable procedures in the literature. This is accomplished by choosing a relatively small sample of models from the full model space and finding a design that is explicitly robust for only these models. For large model spaces, the time savings is many orders of magnitude and this has allowed designs to be routinely produced that have been heretofore unobtainable. Furthermore, these designs are competitive in terms of model robustness to designs that are constructed while utilizing the full model space.

This design method has implications for several types of experiments and we have investigated two in particular. One is for supersaturated designs in which there are no more runs than main effects, and the model space is the set of all models for which g of the main effects are active. The other is an alternative to fractional factorial designs that requires the experimenter to specify the number of two-factor interactions that are likely to be active (or at least a reasonable upper bound on this quantity). The consequent model space consists of all models which include the main effects and g two-factor interactions. The designs produced can typically estimate a large proportion of these models, with relatively high efficiency.

We expect that the approximate model space approach will be effective for other model spaces as well. Though not included here, some preliminary work was done for the model space that includes

all possible submodels of a specified maximal model, and the results were promising. One could also consider such a model space that respects some form of effect heredity. Such model spaces have been considered in the literature by Tsai and Gilmour (2010) and Smucker et al. (2011). Another, the projective model space (Loeppky et al. 2007), is the set of all models which include g of k active main effects and all $\binom{g}{2}$ associated two-factor interactions.

Optimal design of experiments has progressed significantly since the time of Kiefer. Computational power is now great enough that experimenters need not restrict themselves to a single, assumed model. Recent work (see, for instance, Lu et al. 2011; Gilmour and Trinca 2012) has urged consideration of multiple criteria when constructing an optimal design. Instead of assuming a single model and a single optimality criterion, the idea is to simultaneously account for many different design objectives, such as model discrimination, parameter estimation, and error estimation. The work represented in this article makes the model-robust portion of this multivariate optimization problem computationally feasible.

Appendix: Model Sampling Without Explicit Construction of Complete Model Space

For model spaces with as many elements as we consider in this paper it becomes impossible to explicitly construct the full model space, so we present a procedure to quickly sample an arbitrary model within the full model space. We focus here on the MEPI model space.

To do this, we first represent a model as a vector of a length equal to the number of changeable elements in the model. In the case of the MEPI model space in which all main effects are assumed to be active, the $c = \binom{k}{2}$ two-factor interactions are the changeable elements. Since a MEPI model includes g of these two-factor interactions, the vector representing a particular model will have c elements, with g of them 1s and the rest 0s. For instance for $k = 5$ and $g = 4$, $c = \binom{5}{2} = 10$ and a model which included the x_1x_2, x_1x_3, x_1x_4, and x_4x_5 interactions would be represented as $\{1110000001\}$.

Now assume that we want to specify an arbitrary model d in this way, for arbitrary k and g which induces $c = \binom{k}{2}$ possible two-factor interactions and $r = \binom{c}{g}$ possible models. To do this,

we use the following identity, for arbitrary integers a and b where $a > b$:

$$\binom{a}{b} = \binom{a-1}{b-1} + \binom{a-2}{b-1} + \ldots + \binom{b-1}{b-1} = \sum_{i=1}^{a-(b-1)} \binom{a-i}{b-1} \tag{2}$$

We also define the associated partial sum $P_t(a, b)$ as

$$P_t(a, b) = \sum_{i=1}^{t} \binom{a-i}{b-1} \tag{3}$$

We can repeatedly apply identity (2) to map a model number (e.g. a number between 1 and r) to a c-vector with g 1s and $c - g$ 0s. Assume that we wish to map model d to its associated c-vector.

1. Determine the first included term by applying (2) to $\binom{c}{g}$ and determine t, the smallest integer for which the partial sum $P_t(c, g) \geq d$. Let $i_1 = t$.
2. Apply (2) to $\binom{c-i_1}{g-1}$ and determine t, the smallest integer for which $P_{i_1-1}(c, g) + P_t(c - i_1, g - 1) \geq d$. Let $i_2 = i_1 + t$.
3. Apply (2) to $\binom{c-i_2}{g-2}$ and determine t, the smallest integer for which $P_{i_1-1}(c, g) + P_{i_2-1}(c - i_1, g - 1) + P_t(c - i_2, g - 2) \geq d$. Let $i_3 = i_2 + t$.

...

g. Apply (2) to $\binom{c-i_{g-1}}{g-g}$ and determine t, the smallest integer for which $P_{i_1-1}(c, g) + P_{i_2-1}(c - i_1, g - 1) + P_{i_3-1}(c - i_2, g - 2) + \ldots + P_{i_{g-1}-1}(c - i_{g-2}, g - (g - 1)) + P_{i_g}(c - i_{g-1}, g - g) \geq d$. Let $i_g = i_{g-1} + t$.

Then, model d is represented by the c-vector that has 1s in the elements represented by $i_1, i_2, \ldots, i_g$, and 0s otherwise.

To clarify, assume as before that $k = 5$ and $g = 4$, which leads to $c = 10$ and $r = \binom{10}{4} = 210$. Suppose further that we wish to specify model $d = 90$. Then,

1. Determine the first active term by applying (2) to $\binom{10}{4}$. We find that the first term is $\binom{9}{3} = 84$ and the second term is $\binom{8}{3} = 56$, so that $P_1(10, 4) = 84$ but $P_2(10, 4) = 140$ which leads to $t = 2$. Let $i_1 = t = 2$.

2. The term corresponding to $t = 2$ is $\binom{c-2}{g-1} = \binom{8}{3}$ and so we apply (2) to it. Since $P_1(8,3) = \binom{7}{2} = 21$, $P_1(10,4) + P_1(8,3) = 84 + 21 > d$, so $t = 1$ and $i_2 = i_1 + t = 3$.

3. The term corresponding to $t = 1$ is $\binom{7}{2}$ and so we apply (2) to it. Since $P_1(7,2) = \binom{6}{1} = 6$ and $P_1(10,4) + P_0(8,3) + P_1(7,2) = 84 + 0 + 6 = d$, set $t = 1$ and let $i_3 = i_2 + t = 4$.

4. The term corresponding to $t = 1$ is $\binom{6}{1}$ and so we apply (2) to it: $\binom{6}{1} = \binom{5}{0} + \binom{4}{0} + \binom{3}{0} + \binom{2}{0} + \binom{1}{0} + \binom{0}{0}$. In this case, $P_6(6,1) = 6$, so $P_1(10,4) + P_0(8,3) + P_0(7,2) + P_6(6,1) = 84 + 0 + 0 + 6 = d$ which implies that $t = 6$ and thus $i_4 = i_3 + t = 10$.

Thus, we have determined that model 90 is equivalent to model $\{0111000001\}$.

This procedure can be used when sampling models from the full model space, by first sampling a number between 1 and r, and then using the above procedure to convert the model into a form that can be utilized in the algorithm.

Supplementary Materials

Supplementary material for "Approximate Model Spaces for Model-Robust Design" is included online and consists of the following:

A_Additional_Design_Results A folder which includes RCP results with CIs, design results based upon coordinate exchange, and model discrimination results.

B_Additional_Meta-Design_Results A folder containing a file comparing results using NBIBD for the approximate model space with a random selection of this set.

C_Designs A folder containing a collection of text files providing the designs described in Tables 3 and 4, and Tables 1-5 in Supplement A.2.

D_Code A folder containing several subfolders with SAS code (to generate NBIBDs) and Matlab code (to generate model-robust designs).

Acknowledgements

The authors express their thanks to John Bailer, Steve Wright, and Brad Jones, who gave feedback and ideas on an earlier version of this paper. Additionally, the reviewers, the associate editor, and the editor made suggestions that positively shaped the paper. Also, Jens Mueller provided vital computing assistance, and he is gratefully acknowledged. The first author also recognizes the support of the Committee on Faculty Research at Miami University as well as a summer research grant for new tenure-track faculty from the College of Arts and Science.

References

Bingham, D. R. and Chipman, H. A. (2007), "Incorporating Prior Information in Optimal Design for Model Selection," *Technometrics*, 49(2), 155–163.

Cheng, C.-S. and Bailey, R. (1991), "Optimality of Some Two-Associate-Class Partially Balanced Incomplete Block Designs," *The Annals of Statistics*, 19(3), 1667–1671.

Cook, R. D. and Nachtsheim, C. J. (1982), "An Analysis of the k-exchange Algorithm," Tech. rep., University of Minnesota, Dept. of Operations and Management Science.

— (1989), "Computer-Aided Blocking of Factorial and Response-Surface Designs," *Technometrics*, 31(3), 339–346.

Dean, A. and Voss, D. (1999), *Design and Analysis of Experiments*, New York: Springer-Verlag.

Dette, H. (1990), "A generalization of $\mathcal{D}$- and $\mathcal{D}_1$-optimal design in polynomial regression," *Annals of Statistics*, 18, 1784–1804.

Edwards, D. and Mee, R. (2011), "Supersaturated Designs: Are Our Results Significant?" *Computational Statistics & Data Analysis*, 55(9), 2652–2664.

Gilmour, S. G. and Trinca, L. A. (2012), "Optimum design of experiments for statistical inference," *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 61(3), 345–401.

Heredia-Langner, A., Montgomery, D. C., Carlyle, W. M., and Borror, C. M. (2004), "Model-Robust Optimal Designs: A Genetic Algorithm Approach," *Journal of Quality Technology*, 36(3), 263–279.

Jin, R., Chen, W., and Sudjianto, A. (2005), "An Efficient Algorithm for Constructing Optimal Design of Computer Experiments," *Journal of Statistical Planning and Inference*, 134(1), 268–287.

Jones, B. A., Li, W., Nachtsheim, C. J., and Ye, K. Q. (2009), "Model-robust supersaturated and partially supersaturated designs," *Journal of Statistical Planning and Inference*, 139, 45–53.

Kiefer, J. (1959), "Optimum Experimental Designs," *Journal of the Royal Statistical Society, Series B*, 21, 272–304.

— (1975), *A Survey of Statistical Designs and Linear Models*, North-Holland Amsterdam, chap. Construction and optimality of generalized Youden designs, pp. 333–353.

Läuter, E. (1974), "Experimental design in a class of models," *Mathematische Operationsforschung und Statistik*, 5, 379–398.

Li, W. and Nachtsheim, C. J. (2000), "Model-Robust Factorial Designs," *Technometrics*, 42(4), 345–352.

Li, X., Sudarsanam, N., and Frey, D. (2006), "Regularities in Data from Factorial Experiments," *Complexity*, 11(5), 32–45.

Loeppky, J. L., Sitter, R. R., and Tang, B. (2007), "Nonregular Designs with Desirable Projection Properties," *Technometrics*, 49(4), 454–467.

Lu, L., Anderson-Cook, C., and Robinson, T. (2011), "Optimization of Designed Experiments Based on Multiple Criteria Utilizing a Pareto Frontier," *Technometrics*, 53(4), 353–365.

Marley, C. and Woods, D. (2010), "A Comparison of Design and Model Selection Methods for Supersaturated Experiments," *Computational Statistics & Data Analysis*, 54(12), 3158–3167.

Meyer, R. K. and Nachtsheim, C. J. (1995), "The Coordinate-Exchange Algorithm for Constructing Exact Optimal Experimental Designs," *Technometrics*, 37(1), 60–69.

SAS (2013), *SAS/QC 9.2 User's Guide, Second Edition, The OPTEX Procedure, "Design Efficiency Measures"*, SAS Insitute Inc.

Smucker, B. J., del Castillo, E., and Rosenberger, J. L. (2011), "Exchange Algorithms for Constructing Model-Robust Experimental Designs," *Journal of Quality Technology*, 43(1), 28–42.

— (2012), "Model-Robust Two-Level Designs Using Coordinate Exchange Algorithms and a Maximin Criterion," *Technometrics*, 54(4), 367–275.

Sun, D. X. (1993), "Estimation capacity and related topics in experimental designs," Ph.D. thesis, University of Waterloo.

Tsai, P.-W. and Gilmour, S. G. (2010), "A General Criterion for Factorial Designs under Model Uncertainty," *Technometrics*, 52(2), 231–242.

Wu, C. F. J. and Hamada, M. S. (2009), *Experiments: Planning, Analysis, and Optimization*, Hoboken, NJ: Wiley, 2nd ed.