

Model-Robust Designs for Split-Plot Experiments

Byran J. Smucker^{a,*}, Enrique del Castillo^b, James L. Rosenberger^c

^a*Department of Statistics, Miami University, Oxford, OH USA*

^b*Department of Industrial and Manufacturing Engineering, The Pennsylvania State University, State College, PA USA*

^c*Department of Statistics, The Pennsylvania State University, State College, PA USA*

Abstract

Split-plot experiments are appropriate when some factors are more difficult and/or expensive to change than others. They require two levels of randomization resulting in a non-independent error structure. The design of such experiments has garnered much recent attention, including work on exact \mathcal{D} -optimal split-plot designs. However, many of these procedures rely on the *a priori* assumption that the form of the regression function is known. We relax this assumption by allowing a set of model forms to be specified, and use a scaled product criterion along with an exchange algorithm to produce designs that account for all models in the set. We include also a generalization which allows weights to be assigned to each model, though they appear to have only a slight effect. We present two examples from the literature, and compare the scaled product designs with designs optimal for a single model. We also discuss a maximin alternative.

Keywords:

D-optimality, exact experimental design, model-robust, split-plot, maximin

1. Introduction

Though dating back to Yates (1935), split-plot experiments have undergone a renaissance of sorts in the experimental design literature over the last twenty years. Research by Lucas and coauthors (e.g. Lucas and Ju, 1992; Anbari and Lucas, 1994) and Letsinger, Myers, and Lentner (1996), as well as an abundance of later work (e.g. Bingham, Schoen, and Sitter, 2004; Goos, 2006; Anbari and Lucas, 2008), suggests a rising level of awareness of these experiments among researchers

and practitioners. In 2002, Goos produced a monograph on the subject, and recently Jones and Nachtsheim (2009) gave a thorough review of both the design and analysis of split-plot experiments.

Split-plot experiments arise when certain factors have levels that are significantly harder and/or more expensive to change (whole plot factors) than others (subplot factors). To reduce the number of times the whole plot factor levels must be reset, the design is assembled using two levels of randomization. This results in fewer whole plot factor level changes than required in a completely randomized design (CRD) which in turn saves the experimenter time and/or money. The next section describes split-plot experiments in more detail.

*Corresponding author: Department of Statistics, Miami University, Oxford OH 45056, USA; Tel: +1 513 529 0955; Fax: +1 513 529 0989; doi: <http://dx.doi.org/10.1016/j.csda.2012.03.010>

Email address: smuckerb@muohio.edu (Byran J. Smucker)

As an example, suppose an experiment is to be conducted to learn about factors affecting the strength of a type of ceramic pipe (Vining, Kowalski, and Montgomery, 2005). Two of the factors are concerned with the temperature of different parts of a furnace, and have levels that are difficult to change from run to run. The other two factors are the amount of binder and the grinding speed, and have levels that are easy to change. A desirable experiment requires relatively few changes to the temperature factor levels; thus, a split-plot design with a relatively small number of whole plots is appropriate. We return to this example in §4.1.

For most of their history, split-plot experiments have been viewed as categorical designs (that is, designs with qualitative factors). However, the recent split-plot resurgence has seen their reexamination as designed experiments with continuous factors. This approach has spawned relatively recent work in the optimal design of split-plot experiments, much of which has utilized \mathcal{D} -optimality (e.g. Goos and Vandebroek, 2001, 2003, 2004; Jones and Goos, 2007). This criterion is used to produce exact designs that estimate model parameters as precisely as possible. As with CRDs, these optimal designs require the specification of a particular relationship between the response and factors, i.e. the model form. If the assumed model is a poor approximation, the design obtained will not be optimal at all. Model-robust design, in this case for split-plot experiments, attenuates the dependence on such a single assumed model.

The CRD version of the model-robustness problem has a long history, beginning with Box and Draper (1959) and exhaustively developed for the case of infinite run sizes (e.g. Stigler, 1971; Läuter, 1974; Studen, 1982; Cook and Nachtsheim, 1982; Dette, 1990; Dette and Franke, 2001; Fang and Wiens, 2003). The

version of this problem which assumes finite run sizes (exact designs) has received somewhat less attention, though algorithms have been developed to address the problem in various ways (Welch, 1983; DuMouchel and Jones, 1994; Li and Nachtsheim, 2000; Heredia-Langner, Montgomery, Carlyle, and Borror, 2004; Tsai and Gilmour, 2010; Smucker, del Castillo, and Rosenberger, 2011, 2012). In particular, Li and Nachtsheim (2000), Heredia-Langner et al. (2004), and Smucker et al. (2011) study exact CRDs while using criteria similar to that used in this paper.

For split-plot experiments, model-robust design methods do not exist. In this paper, we provide a framework which allows the model form assumption to be relaxed, and present an exchange algorithm to construct model-robust split-plot designs. We allow the user to specify a set of models, instead of a single model, and rather than using the \mathcal{D} -optimality criterion to construct the design, we maximize a scaled version of the product of the determinants of the information matrices with respect to each model in the set. A generalization allows the experimenter to specify a level of interest in each model, giving the experimenter recourse if certain models are preferred over others.

Besides \mathcal{D} -optimal design, there are other approaches to the design of split-plot experiments. An algorithmic alternative is that of Trinca and Gilmour (2001). Their procedure applies to the more general setting in which there are any number of randomization strata (as opposed to the split-plot case in which there are only two), and they use orthogonality as their criterion so that they can avoid the need to specify any prior information about the variance components. Recently, Jones and Goos (2012) have advocated \mathcal{I} -optimal split-plot designs which minimize the average prediction variance

across the region of experimentation. Fractional factorial split-plot designs have also been extensively studied (Huang, Chen, and Voelkel, 1998; Bingham and Sitter, 1999; Bisgaard, 2000; Bingham and Sitter, 2001), often using the minimum aberration criteria to differentiate competing designs. Other authors have generalized second-order response surface designs, such as central composite and Box-Behnken, to the split-plot case (e.g. Draper and John, 1998; Vining et al., 2005; Parker, Kowalski, and Vining, 2006, 2007). The latter two papers exploit an equivalent estimation property that allows the split-plot experiment to be analyzed via ordinary least squares.

2. The Split-Plot Design and Model

A split-plot experiment consists of two separate sets of factors: the n_w difficult to change whole plot variables, denoted $\mathbf{w} = (w_1, w_2, \dots, w_{n_w})$, and the remaining n_s easy to change subplot factors, denoted $\mathbf{s} = (s_1, s_2, \dots, s_{n_s})$. A split-plot design specifies as many factor level combinations of \mathbf{w} as there are whole plots, and within each whole plot it specifies as many factor level combinations of \mathbf{s} as there are experimental runs in the whole plot. When the design is executed, the whole plot factor level combinations are randomly assigned to the whole plots, and within each whole plot, the subplot factor level combinations specified in the design are randomly assigned to the experimental runs.

The regression model for the j^{th} observation within the i^{th} whole plot is

$$y_{ij} = \mathbf{f}'(\mathbf{w}_i, \mathbf{s}_{ij})\boldsymbol{\tau} + \delta_i + \epsilon_{ij}, \quad (1)$$

where $i = 1, \dots, b$, $j = 1, \dots, u_i$, \mathbf{f} is the model expansion in terms of the whole plot and subplot factors, $\boldsymbol{\tau}$ is

a p -vector of model parameters, δ_i is the whole plot error term, and ϵ_{ij} is the subplot error term. Note that b is the number of whole plots and u_i is the number of runs in whole plot i . Both δ_i and ϵ_{ij} are assumed to be *iid* normal with mean 0 and variances σ_δ^2 and σ_ϵ^2 , respectively. The parameter vector $\boldsymbol{\tau}$ includes the intercept, p_w parameters for whole plot factors only, p_s parameters for subplot factors only, and p_{ws} parameters for whole plot-by-subplot interactions. We can write the model in matrix notation as

$$\mathbf{Y} = X\boldsymbol{\tau} + Z\boldsymbol{\delta} + \boldsymbol{\epsilon}, \quad (2)$$

where X is an $n \times p$ matrix ($n = \sum_{i=1}^b u_i$ and $p = 1 + p_w + p_s + p_{ws}$), $\boldsymbol{\tau}$ is a p -vector, Z is an $n \times b$ matrix assigning runs to whole plots, $\boldsymbol{\delta}$ is a b -vector, and \mathbf{Y} and $\boldsymbol{\epsilon}$ are n -vectors.

Each observation y_{ij} has two variance components, one contributed by the whole plot, σ_δ^2 , and one contributed by the subplot, σ_ϵ^2 . Thus, $\text{Var}(y_{ij}) = \sigma_\delta^2 + \sigma_\epsilon^2$. Within whole plot i , observations are correlated. This, along with the assumption that δ_i and ϵ_{ij} are independent, implies that $\text{Cov}(y_{ij}, y_{i'j'}) = \sigma_\delta^2$, where $j \neq j'$. Thus, the variance-covariance matrix for whole plot i is

$$\begin{aligned} V_i &= \sigma_\epsilon^2 I_{u_i} + \sigma_\delta^2 \mathbf{1}_{u_i} \mathbf{1}'_{u_i} \\ &= \sigma_\epsilon^2 (I_{u_i} + d \mathbf{1}_{u_i} \mathbf{1}'_{u_i}), \end{aligned} \quad (3)$$

where $d = \sigma_\delta^2 / \sigma_\epsilon^2$, u_i is the number of runs in whole plot i , I_{u_i} is the identity matrix of dimension u_i , and $\mathbf{1}_{u_i}$ is an u_i -vector of ones. Since the whole plots are assumed to be independent of one another, the covariance matrix

for \mathbf{Y} in (2) is block diagonal:

$$V = \begin{pmatrix} V_1 & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & V_2 & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & V_b \end{pmatrix}.$$

We can represent an exact split-plot design as

$$\xi_{b\mathbf{u}} = [(\mathbf{w}_1, \mathbf{s}_{11}), (\mathbf{w}_1, \mathbf{s}_{12}), \dots, (\mathbf{w}_1, \mathbf{s}_{1u_1}), (\mathbf{w}_2, \mathbf{s}_{21}), \dots, (\mathbf{w}_b, \mathbf{s}_{bu_b})], \quad (4)$$

where b is the number of whole plots, $\mathbf{u} = (u_1, u_2, \dots, u_b)$ is a vector representing the number of runs in each whole plot, and d is the ratio of the variance components, $\sigma_\delta^2/\sigma_\epsilon^2$.

To estimate the parameters of the split-plot model, we use Generalized Least Squares (GLS), which gives $\hat{\boldsymbol{\tau}} = (X'V^{-1}X)^{-1}X'V^{-1}\mathbf{Y}$ so that $\text{Var}(\hat{\boldsymbol{\tau}}) = (X'V^{-1}X)^{-1}$. Note that these estimators are a function of the unknown variance parameters σ_δ^2 and σ_ϵ^2 . The information matrix is

$$M(\xi_{b\mathbf{u}}, d) = X'V^{-1}X.$$

The \mathcal{D} -criterion is given by

$$\mathcal{D} = |M(\xi_{b\mathbf{u}}, d)| = |X'V^{-1}X|. \quad (5)$$

and maximizing this criterion is equivalent to minimizing the generalized variance of the parameter estimates. We note that since V_i can be written as the product of the scalar σ_ϵ^2 and a matrix that depends on d (as in Equation 3), we can without loss of generality assume $\sigma_\epsilon^2 = 1$ and calculate the \mathcal{D} -criterion by specifying d .

The model-robust criteria used in this paper are based upon the \mathcal{D} -criterion, which is particularly appropriate

when variances of and covariances between parameter estimates should be made small. However, though it does not focus directly on minimizing a function of the prediction variance, there is some empirical evidence (Lucas, 1974; Donev and Atkinson, 1988) suggesting that \mathcal{D} -optimal designs perform solidly with respect to the prediction variance-based criteria, \mathcal{G} and \mathcal{V} . Because the \mathcal{D} -criterion is computationally expedient and widely used in practice, we adopt it as the basis of our algorithm here.

3. A Model-Robust Exchange Algorithm for Split-Plot Experiments

In this section, we first review the basic idea of the exchange algorithm for \mathcal{D} -optimal split-plot designs, given by Goos and Vandebroek (2003), the goal of which is to find a design that maximizes the \mathcal{D} -criterion, (5). We then develop a generalization of the algorithm which is robust for a set of user-specified models, achieving this robustness by seeking the design which maximizes the product of the determinants of the information matrices for each model in the set, where each determinant is scaled by the number of parameters in the model. We also give, briefly, a description and discussion of a maximin model-robust criterion.

3.1. \mathcal{D} -Optimal Split-Plot Exchange Algorithm

The exchange algorithm of Goos and Vandebroek (2003) requires specification of the number of whole plots as well as the number of runs in each whole plot (i.e. total number of runs). Goos and Vandebroek (2004) give an algorithm for which these requirements are relaxed. However, when split-plot experiments are called for, it is most often because certain factors are

difficult or expensive to change. In that case, the experimenter will most often want to predetermine the number of times the whole plot factor levels are changed (i.e. the number of whole plots). This, along with the typical total sample size constraint, will suggest the size of the whole plots as well (the procedure allows whole plots to have equal or unequal sizes). Thus, the algorithm of interest is the one given in Goos and Vandebroek (2003) which requires the specification of these values. In addition, this \mathcal{D} -optimal split-plot exchange algorithm requires the construction of a candidate list and a given, assumed model form.

First, the algorithm stochastically selects an initial design, and then iteratively improves it by considering three types of exchanges: 1) Swaps of current design points with candidate points having the same whole plot factor settings; 2) Swaps of current design points with other design points having the same whole plot factor settings; and 3) Swaps of current whole plot factor settings with other possible whole plot factor settings. Exchanges of the first and second type are complicated by the requirement that design points may only be swapped if they share the same levels of the whole plot factor(s). The third type of exchange is quite powerful, because it allows all design points in a particular whole plot to be replaced by a whole plot with new levels of the whole plot factor.

For each mechanism in turn, all possible exchanges are considered for each appropriate design point (or whole plot). The algorithm iterates until convergence. Many algorithm tries are used to avoid getting stuck in local optima. The model-robust algorithm developed below is similar to this procedure in structure and approach.

A key feature of exchange algorithms for CRDs,

which lessens the computational requirements imposed by such a brute force method, is the use of updating formulae for the determinant and inverse of the information matrix. Arnouts and Goos (2010) have developed analogous updating formulae for split-plot information matrices, which are reproduced in Appendix A and referred to as necessary.

3.2. Model-Robust Split-Plot Exchange Algorithm

Our model-robust approach for the split-plot optimal design problem is based upon \mathcal{F} , a user-specified set of m models, instead of the classical optimal design assumption that a single \mathbf{f} is known in advance of the experiment. Our goal is to find a design that maximizes the product of determinants of the information matrices for each model, where each determinant is scaled by the number of parameters in the associated model. The criterion of interest is

$$\phi(\xi_{bu}, d, \mathcal{F}, \mathbf{v}) = \prod_{i=1}^m |M_{\mathbf{f}_i}(\xi_{bu}, d)|^{v_{\mathbf{f}_i}/p_i}, \quad (6)$$

where we maximize over all $\xi_{bu} \in \Xi_{bu}$ and Ξ_{bu} is the space of all possible split-plot designs with b whole plots of sizes $\mathbf{u} = (u_1, u_2, \dots, u_b)$ and with d , the ratio of the variance components. The quantity $v_{\mathbf{f}_i}$ is the weight for model i and p_i is the number of parameters in model i .

Since the traditional \mathcal{D} -criterion works to make the generalized variance of the parameter estimates small (i.e. their variances and covariances), this model-robust criterion encourages the same while accounting for each model in the set. The determinants are scaled by the appropriate number of parameters to, as Atkinson, Donev, and Tobias (2007, p. 368) put it, bring each quantity to the “dimensions of ... [a variance].”

The weight, or model interest, vector is $\mathbf{v} = (v_{\mathbf{f}_1}, \dots, v_{\mathbf{f}_m}) \in (0, 1]^m$ and represents the experimenter's level of interest in each of the m specified models. We scale this vector so that the *a priori* most likely model(s) has (have) a model interest level of 1. Note that the user could cast these model interest weights as prior probabilities simply by rescaling them to sum to one. Note also that typical usage will utilize equal weights for all models; however, if there are clear preferences for some models, the weights can handle this prior information.

Clearly, this model-robust procedure is more demanding computationally than the original split-plot exchange algorithm, because there are m models to consider instead of just one. The algorithm itself uses the same exchange mechanisms and much the same notation as the Goos and Vandebroek (2003) procedure, but utilizes the set of models approach via the optimality criterion (6). The details are given in Appendix B, and a Matlab implementation is available at <http://www.users.muohio.edu/smuckebj/research.html>.

3.3. A Possible Alternative Model-Robust Criterion

As an alternative to the scaled product criterion, one might consider maximizing the minimum efficiency. That is,

$$\max_{\xi_{bu} \in \Xi_{bu}} \min_{\mathbf{f} \in \mathcal{F}} E_{\mathbf{f}}(\xi_{bu}, d), \quad (7)$$

where $E_{\mathbf{f}}(\xi_{bu}, d)$ is the \mathcal{D} -efficiency for a design ξ_{bu} with respect to model \mathbf{f} , calculated as

$$E_{\mathbf{f}}(\xi_{bu}, d) = \left(\frac{|M_{\mathbf{f}}(\xi_{bu}, d)|}{|M_{\mathbf{f}}(\xi_{bu}^*)|} \right)^{1/p}. \quad (8)$$

Here, $M_{\mathbf{f}}(\xi_{bu}^*)$ is the information matrix for the \mathcal{D} -optimal split-plot design for model \mathbf{f} alone, and p is the number of parameters in model \mathbf{f} . This criterion produces designs that prepare for the worst-case by max-

imizing the smallest \mathcal{D} -efficiency across all models of interest.

However, there are at least two significant drawbacks to using this criterion. First, this model-robust procedure is computationally more demanding than the scaled-product model-robust algorithm because it requires that the optimal designs be found for each of the models individually. The severity of this requirement is blunted somewhat by the existence of at least one commercially available split-plot design algorithm (JMP).

The second and more important problem with the maximin criterion is evident when the model set includes models with widely varying numbers of parameters. In this case, maximin designs are produced that have less variable efficiencies with respect to the considered models. But this is problematic, because in general a design with a high \mathcal{D} -efficiency for a large model will still have a larger generalized variance (and larger parameter estimate variances) than for a small model with much lower \mathcal{D} -efficiency. By reducing the efficiencies for the large models and increasing them for the smaller, the maximin design results in unfortunately and avoidably large variances for the large models.

4. Examples

In this section we illustrate our model-robust procedures using two examples from the literature. The first involves two whole plot factors and two subplot factors. The second includes two process (whole plot) factors and three mixture (subplot) factors. In what follows, we reference and discuss a large number of model-robust designs. In the interest of space, they can be found at the first author's website (<http://www.users.muohio.edu/smuckebj/research.html>).

Design	Measure	$d = 1$				$d = 5.65$			
		(10)	(11)	(12)	(13)	(10)	(11)	(12)	(13)
\mathcal{F}_1 -Scaled Product	Determinant	1.37e06	1.10e15	4.33e15	0	1.36e4	2.75e12	2.60e11	0
	Scaled Determinant	16.87	23.31	11.02	0	6.71	13.52	5.91	0
	\mathcal{D} -efficiency	.923	.872	.927	0	.922	.871	.927	0
Optimal for (12)	Determinant	9.04e5	2.18e14	1.35e16	0	8.47e3	4.23e11	1.16e12	0
	Scaled Determinant	15.53	20.11	11.90	0	6.10	11.40	6.37	0
	\mathcal{D} -efficiency	.850	.752	1	0	.847	.750	1.000	0
\mathcal{F}_2 -Scaled Product	Determinant	8.26e5	1.20e14	3.86e15	8.36e24	7.80e3	2.36e11	3.18e11	1.32e18
	Scaled Determinant	15.25	19.06	10.94	5.15	6.00	10.81	5.85	3.29
	\mathcal{D} -efficiency	.835	.713	.920	.948	.834	.711	.917	.945
(.8,.8,1,.5)- \mathcal{F}_2 -Scaled Product	Determinant	8.72e5	1.49e14	4.03e15	2.59e24	8.24e3	2.94e11	3.41e11	4.03e17
	Scaled Determinant	15.42	19.43	10.97	4.98	6.07	11.03	5.87	3.18
	\mathcal{D} -efficiency	.844	.727	.922	.917	.843	.725	.921	.914
Optimal for (13)	Determinant	5.13e5	1.81e13	2.06e15	5.41e25	4.88e3	3.64e10	1.77e11	9.52e18
	Scaled Determinant	13.87	16.04	10.49	5.44	5.47	9.12	5.62	3.49
	\mathcal{D} -efficiency	.759	.600	.882	1	.758	.599	.882	.999
Vining et al. (2005)	Determinant	7.17e4	2.40e11	9.09e11	0	6.82e2	4.83e8	7.81e7	0
	Scaled Determinant	9.36	10.83	6.27	0	3.69	6.16	3.36	0
	\mathcal{D} -efficiency	.512	.405	.582	0	.512	.405	.527	0

Table 1: For the ceramic pipe experiment, the determinant of the information matrix calculated as in (5), the determinant scaled by the number of parameters, and the \mathcal{D} -efficiencies, for various designs. These quantities are calculated with respect to models (10)-(13), and results are given assuming $d = 1$ as well as $d = 5.65$.

4.1. Strength of Ceramic Pipe Experiment

Often, experimenters have a particular model in mind when approaching an experiment. However, it is usually difficult to know whether this preconceived notion will be borne out by the data. Consequently, model-robust designs allow the experimenter additional flexibility because there is no need to commit to a design for a particular model. Rather, a (possibly large) group of plausible models can be chosen, each weighted by the *a priori* likelihood of it being the true model, and a design crafted that accounts for this model uncertainty.

In a reprise of the example at the outset (Vining et al., 2005), we return to the strength of ceramic pipe experiment. Recall that there were two hard-to-change factors (zone 1 temperature of furnace, w_1 ; zone 2 temperature of furnace, w_2) and two easy-to-change factors (amount of binder, s_1 ; grinding speed, s_2). The experiment used 12 whole plots each of size 4 and after running the experiment, it was found that $\hat{d} = \hat{\sigma}_\delta^2 / \hat{\sigma}_\epsilon^2 =$

$.52828 / .09348 = 5.65$. In Vining et al. (2005), the model was assumed to be full quadratic in all factors, ostensibly because the experimenters wished to allow for the modeling of curvature:

$$\mathbf{f}'(\mathbf{w}, \mathbf{s}) = \left(1, w_1, w_2, w_1 w_2, w_1^2, w_2^2, s_1, s_2, s_1 s_2, w_1 s_1, w_1 s_2, w_2 s_1, w_2 s_2, s_1^2, s_2^2 \right), \quad (9)$$

with design space scaled to be $\mathcal{X} = \{\mathbf{w} = (w_1, w_2), \mathbf{s} = (s_1, s_2) : -1 \leq \mathbf{w}, \mathbf{s} \leq 1\}$.

Vining et al. (2005) used a face-centered central composite design (CCD) modified to accommodate the split-plot structure. Another design approach would be to use the \mathcal{D} -optimal exchange algorithm of Goos and Vandebroek (2003) assuming (9) to be the true model. In retrospect, this model seemed to be a good approximation of the true model (Vining et al., 2005), but assuming that there was *a priori* uncertainty about the necessity of the quadratic model, our model-robust technique can be used to construct designs that are less efficient for (9) but more efficient for submodels of

the quadratic. For instance, we might choose $\mathcal{F}_1 = \{\mathbf{f}'_i(\mathbf{w}, \mathbf{s})\tau_i, 1 \leq i \leq 3; \mathbf{w}, \mathbf{s} \in \mathcal{X}\}$ with

$$\mathbf{f}'_1(\mathbf{w}, \mathbf{s}) = (1, w_1, w_2, s_1, s_2), \quad (10)$$

$$\mathbf{f}'_2(\mathbf{w}, \mathbf{s}) = (\mathbf{f}'_1, w_1w_2, s_1s_2, w_1s_1, w_1s_2, w_2s_1, w_2s_2), \quad (11)$$

$$\mathbf{f}'_3(\mathbf{w}, \mathbf{s}) = (\mathbf{f}'_2, w_1^2, w_2^2, s_1^2, s_2^2). \quad (12)$$

A weakness of optimal design—addressed in the CRD case by, for instance, DuMouchel and Jones (1994), Goos, Kobilinsky, O’Brien, and Vandebroek (2005) and Gilmour and Trinca (2012)—is that it devotes all of the available experimental resources to increasing the efficiency of the design with respect to the assumed model. Certainly, this approach will allow relevant submodels to be fit, but will not provide flexibility should a higher-order model be required. The present model-robust framework provides a solution to the single-model assumption, but also suggests a way to make the design robust for the higher-order model: Construct a design that is efficient with respect to a larger model than might likely be fit. This hedges the experimenter against this higher-order model’s unlikely appearance while still allowing more plausible submodels to be fit with relatively high efficiency. (We note that a more systematic method could be to incorporate a lack-of-fit criterion into the model-robust criterion, but this is beyond the scope of this work.) With this in mind, we also take $\mathcal{F}_2 = \{\mathcal{F}_1, \mathbf{f}'_4\}$ with

$$\mathbf{f}_4 = (\mathbf{f}_3, w_1w_2s_1, w_1w_2s_2, w_1s_1s_2, w_2s_1s_2, w_1^2w_2, w_1^2s_1, w_1^2s_2, w_2^2w_1, w_2^2s_1, w_2^2s_2, s_1^2w_1, s_1^2w_2, s_1^2s_2, s_2^2w_1, s_2^2w_2, s_2^2s_1, w_1^3, w_2^3, s_1^3, s_2^3), \quad (13)$$

the full cubic model.

In Table 1, we evaluate several designs with respect to the four models defined above. For each design, we give the determinant of the information matrix with respect to the four models defined above; the determinant scaled by the number of parameters in the particular model; and the \mathcal{D} -efficiency. We give results for $d = 1$ and $d = 5.65$, but in both cases the model-robust designs have been calculated assuming $d = 1$, for lack of any prior knowledge that would suggest a different ratio. In the $d = 5.65$ columns, the measures are calculated assuming the true ratio of variance components as estimated by Vining et al. (2005), and it can be seen that d typically has little effect on the \mathcal{D} -efficiencies. In calculating this table’s model-robust and optimal designs, we used 50 algorithm tries, and a 5^4 candidate list of possible design points (each of the four factors could take any value in the set $\{-1, -0.5, 0, 0.5, 1\}$).

The scaled product design with respect to \mathcal{F}_1 is more efficient than the design optimal for (12), for the smaller models (10) and (11), while giving up some efficiency for (12), the full quadratic model. Note that while the \mathcal{F}_1 -Scaled Product design is relatively efficient for models (10)-(12), it cannot estimate the larger cubic model.

If some protection is desired for a lurking, higher-order model we consider the set \mathcal{F}_2 . The \mathcal{F}_2 -Scaled Product design shows, unsurprisingly, that adding the full cubic model reduces the efficiency with which each of the other three models can be estimated. However, if we take $\mathbf{v} = (.8, .8, 1, .5)$ and compare the resulting scaled product design to the optimal design for (12), we see that the design with unequal weights estimates the smaller models nearly as well, gives up just 8 percentage points for the quadratic model and simultaneously allows for excellent estimation of the full cubic model

(91.7%-efficiency). Again, unless there is strong prior information suggesting unequal weights are appropriate, we would recommend a model interest vector with equal elements.

In terms of \mathcal{D} -efficiency, the design proposed by Vining et al. (2005) is not competitive, but this does not necessarily preclude it from consideration. It has other qualities—for instance, it allows for a model-independent estimate of the variance components and a replication-based lack-of-fit test—that recommend it to an experimenter who wishes to remain within the structure of classic designs (i.e. CCDs).

4.2. An Experiment With Mixture-Process Variables

We next consider an example given in Cornell (1990) and used later by Kowalski, Cornell, and Vining (2002) and Goos and Donev (2007). This experiment measured the thickness of manufactured vinyl for automobile seat covers and included two process factors (extrusion rate, w_1 , and drying temperature, w_2) and three mixture factors (plasticizers, s_1 , s_2 , and s_3). Complete randomization was unwieldy, so the process variables were treated as whole plot factors, and the mixture variables as subplot factors. Though the original design was run in eight whole plots each of size six, Kowalski et al. (2002) and Goos and Donev (2007) considered a design with only 28 total runs, with 7 whole plots each of size 4. We adopt this approach as well.

Design	Measure	(15)	(16)	(17)	(18)	(19)	(20)	Product
\mathcal{F}_1 -Scaled Product	Scaled Determinant	4.70	.710	4.59	4.64	1.79	1.94	246.80
	\mathcal{D} -efficiency	.892	.859	.896	.906	.988	.987	.607
Optimal for (14)	Scaled Determinant	4.38	.792	4.26	4.34	1.82	1.96	228.78
	\mathcal{D} -efficiency	.831	.958	.831	.847	1.00	1.00	.560
Kowalski et al. (2002)	Scaled Determinant	2.27	.458	1.81	1.92	.862	.945	2.94
	\mathcal{D} -efficiency	.430	.555	.354	.375	.475	.481	.007

Table 2: Designs for vinyl-thickness experiment, with \mathcal{D} -efficiencies for models (15)-(20), assuming $d = 1$

Design	Measure	(15)	(16)	(17)	(18)	(19)	(20)	(21)	(22)	(23)	(24)	(25)	(26)	Product
\mathcal{F}_2 -Scaled-Product	Scaled Determinant	4.33	.783	3.93	3.90	1.70	1.80	.296	2.80	.949	1.04	1.53	.954	189.85
	\mathcal{D} -efficiency	.822	.948	.767	.762	.937	.918	.859	.875	.928	.910	.986	.958	.235
Optimal for (26)	Scaled Determinant	4.02	.747	3.39	3.37	1.52	1.61	.329	2.73	.926	1.01	1.49	.996	104.66
	\mathcal{D} -efficiency	.763	.904	.662	.658	.838	.819	.954	.854	.906	.882	.960	1.00	.129

Table 3: Designs for vinyl-thickness experiment, with \mathcal{D} -efficiencies for models (15)-(26), assuming $d = 1$

Kowalski et al. (2002) and Goos and Donev (2007) used the split-plot model, (1), with $\mathbf{w}, \mathbf{s} \in \mathcal{X}$ where

$$\mathbf{f}'(\mathbf{w}, \mathbf{s}) = (s_1, s_2, s_3, s_1 s_2, s_1 s_3, s_2 s_3, w_1 s_1, w_2 s_1, w_1 s_2, w_2 s_2, w_1 s_3, w_2 s_3, w_1 w_2) \quad (14)$$

and

$$\mathcal{X} = \left\{ \mathbf{w} = (w_1, w_2), \mathbf{s} = (s_1, s_2, s_3) : -1 \leq \mathbf{w}, \mathbf{s} \leq 1; \sum_{i=1}^3 s_i = 1 \right\}.$$

Since this experiment involved both mixture and process variables, the model is a combination of an interaction model for the process factors and a second-order Scheffé polynomial model for the mixture factors (see Kowalski et al., 2002). The designs of Kowalski et al. (2002) and Goos and Donev (2007) assumed the model in (14) but in the analysis done by Kowalski et al. (2002) it was concluded that few of the terms were significant and thus the assumed model was overspecified. This was not, of course, known before the experiment was conducted, and illustrates the desirability of accounting for more than a single assumed model at the design stage.

To demonstrate the flexibility of our approach, we consider two experimental scenarios. The first assumes that there is a high degree of confidence that the true model includes some combination of terms in the original fitted model. In this case, we choose

$\mathcal{F}_1 = \{\mathbf{f}'_{1i}(\mathbf{w}, \mathbf{s})\tau_{1i}, 1 \leq i \leq 6; \mathbf{w}, \mathbf{s} \in \mathcal{X}\}$ with

$$\mathbf{f}'_{11}(\mathbf{w}, \mathbf{s}) = (s_1, s_2, s_3), \quad (15)$$

$$\mathbf{f}'_{12}(\mathbf{w}, \mathbf{s}) = (\mathbf{f}'_{11}, s_1 s_2, s_1 s_3, s_2 s_3), \quad (16)$$

$$\mathbf{f}'_{13}(\mathbf{w}, \mathbf{s}) = (\mathbf{f}'_{11}, w_1 s_1, w_2 s_1, w_1 s_2, w_2 s_2, w_1 s_3, w_2 s_3), \quad (17)$$

$$\mathbf{f}'_{14}(\mathbf{w}, \mathbf{s}) = (\mathbf{f}'_{13}, w_1 w_2), \quad (18)$$

$$\mathbf{f}'_{15}(\mathbf{w}, \mathbf{s}) = (\mathbf{f}'_{12}, w_1 s_1, w_2 s_1, w_1 s_2, w_2 s_2, w_1 s_3, w_2 s_3), \quad (19)$$

$$\mathbf{f}'_{16}(\mathbf{w}, \mathbf{s}) = (\mathbf{f}'_{15}, w_1 w_2), \quad (20)$$

as a set of possible models, the largest of which is equivalent to (14). \mathcal{D} -optimal designs based upon these models have little to no estimation capabilities for models with higher order terms, so if the experimenter wishes to guard against this possibility, the augmentation of \mathcal{F}_1 with several such models might be considered: $\mathcal{F}_2 = (\mathcal{F}_1, \{\mathbf{f}'_{2i}(\mathbf{w}, \mathbf{s})\tau_{2i}, 1 \leq i \leq 6; \mathbf{w}, \mathbf{s} \in \mathcal{X}\})$ where

$$\mathbf{f}'_{21}(\mathbf{w}, \mathbf{s}) = (\mathbf{f}'_{12}, s_1 s_2 s_3), \quad (21)$$

$$\mathbf{f}'_{22}(\mathbf{w}, \mathbf{s}) = (\mathbf{f}'_{14}, w_1^2, w_2^2), \quad (22)$$

$$\mathbf{f}'_{23}(\mathbf{w}, \mathbf{s}) = (\mathbf{f}'_{15}, s_1 s_2 s_3), \quad (23)$$

$$\mathbf{f}'_{24}(\mathbf{w}, \mathbf{s}) = (\mathbf{f}'_{16}, s_1 s_2 s_3), \quad (24)$$

$$\mathbf{f}'_{25}(\mathbf{w}, \mathbf{s}) = (\mathbf{f}'_{16}, w_1^2, w_2^2), \quad (25)$$

$$\mathbf{f}'_{26}(\mathbf{w}, \mathbf{s}) = (\mathbf{f}'_{24}, w_1^2, w_2^2). \quad (26)$$

In all the model-robust (and individually optimal) designs for this example, we use 50 algorithm tries with a 90-element candidate list (a 3^2 list for the whole plot variables crossed with a 10-element set of mixture points: $(1, 0, 0)$, $(0, 1, 0)$, $(0, 0, 1)$, $(1/2, 1/2, 0)$, $(1/2, 0, 1/2)$, $(0, 1/2, 1/2)$, $(1/3, 1/3, 1/3)$, $(2/3, 1/6, 1/6)$, $(1/6, 2/3, 1/6)$, and $(1/6, 1/6, 2/3)$). We

have assumed $d = 1$ throughout.

In Table 2 we show, for several designs, the determinant for models (15)-(20) scaled by the number of parameters in each model, as well as the associated \mathcal{D} -efficiencies. Though the design optimal for the original assumed model (14) appears optimal for (19) as well, the scaled-product design is superior in terms of the scaled-product criterion and \mathcal{D} -efficiencies, as it should be. Notice that both designs favor the two largest models in \mathcal{F}_1 , (19) and (20), in terms of efficiencies.

The second scenario is one in which the experimenter believes that a model more complex than (14) is somewhat likely. In this case, we can explicitly account for the higher-order models in \mathcal{F}_2 , while still considering the models in \mathcal{F}_1 as well. In Table 3, it is evident that the design optimal for the largest model, (26), does a relatively poor job of estimating the models (15)-(20); for these models, the model-robust design is on average 8.5 percentage points better than the design optimal for the largest model, though it is slightly worse in terms of average efficiency for the models unique to \mathcal{F}_2 . Overall, if there is robustness desired for all 12 models, it is clear that the model-robust design is preferred to the optimal design for the largest model.

The goal here is to provide a design that performs well over a range of possible models. Higher efficiencies roughly translate to small parameter variances and smaller correlations between parameter estimates. Thus, in the absence of strong prior knowledge concerning the model form, a model-robust design will give a better chance for smaller variances and correlations than a traditional optimal design.

5. Discussion

We have presented a generalization of the split-plot exchange algorithm of Goos and Vandebroek (2003) which relaxes the model form assumption inherent in optimal design. Our algorithm tries to choose the design which maximizes the product of the scaled determinants of the information matrices with respect to a set of specified possible models. This approach accounts for all the models in the set, finding a design that is “good” for each one, and also allows model weights to be specified corresponding to prior information of the likelihood of each particular model.

We also discuss another model-robust criterion—the minimum \mathcal{D} -efficiency—but conclude that it induces a tradeoff which takes more than it gives. On the one hand, the maximin designs tend to produce designs with \mathcal{D} -efficiencies that are less variable with respect to a particular set of models, and in addition have a larger minimum \mathcal{D} -efficiency. On the other hand, this has the effect of reducing the efficiencies with respect to models with larger numbers of parameters. Since these larger models are already more difficult to estimate—as evidenced by generally higher parameter variances—smaller efficiencies mean even higher variances. Consequently, we have focused on the scaled-product criterion instead of the maximin.

Our procedure is based upon an exchange algorithm that uses a candidate list to facilitate exchanges. A computationally superior approach is the algorithm of Jones and Goos (2007), an exchange algorithm for split-plot designs which requires no candidate list. However, it is difficult to implement such an approach when mixture components are involved (such as the second example in this work), because of the associated equality constraint

on the design space. Published work (Piepel, Cooley, and Jones, 2005; Jones and Goos, 2007) suggests that this impediment might be overcome, but conversations with both Drs. Piepel and Jones indicate that in the CRD case, at least, their approach cannot be generally recommended.

In this paper, we have assumed relatively small sets of possible models. The space of all possible models, or model spaces preserving heredity of effects, is much, much larger. In fact, beyond experiments with just two or three factors, the complete model space becomes prohibitively large. However, if the model sets in this paper are viewed as *ad hoc* approximations to the larger, complete model space, one might consider the more principled approach of choosing an approximating set of models based upon some specified criterion. Such an endeavor engenders its own difficulties, both of evaluation and practicality, and is beyond the scope of this work. Still, for model-robust split-plot designs, the idea is a natural extension to our work.

Extensive simulations have been done (e.g. Goos and Vandebroek, 2001) demonstrating that the ratio of variance components, d , has a minimal effect on the optimal designs. This work—in particular Table 1—confirms that. The second example, though we do not show the efficiencies using the value of d estimated in the experiment, gives similar results. In short, little information is lost by a lack of *a priori* knowledge of d , particularly when compared to the precision lost when an incorrect model is assumed.

6. Acknowledgements

The first author gratefully recognizes the U.S. Census Bureau’s Dissertation Fellowship Program, which

has financially supported portions of this work. We are also thankful for the referees who have taken the time to make helpful comments that have significantly improved the final result. One referee, in particular, of an earlier version made an insightful critique that led to a significantly different emphasis and greatly improved the article.

Appendix A. Matrix Results Used in Exchange Algorithms

For reference purposes, we reproduce some results from Arnouts and Goos (2010) which are used as computational shortcuts in our algorithm. Point exchanges in the split-plot design context are more complicated than in CRDs because there are several types of exchanges, each having a different impact upon the determinant and inverse of the information matrix. In what follows we give an information matrix inverse updating formula for the case in which a point is added to the i^{th} whole plot, followed by updating formulae for the information matrix, its inverse, and its determinant, for each of three relevant exchange scenarios.

We first consider the situation in which a design point is added to the i^{th} whole plot, in which case the updated information matrix inverse can be written:

$$(M^*)^{-1} = M^{-1} - M^{-1} \mathbf{U}_1' (\mathbf{T}_1^{-1} + \mathbf{U}_1 M^{-1} \mathbf{U}_1')^{-1} \mathbf{U}_1 M^{-1}, \quad (\text{A.1})$$

where $\mathbf{T}_1 = \text{diag}\left(1, \frac{d}{1+u_i d}, \frac{-d}{1+(u_i+1)d}\right)$, $\mathbf{U}_1 = \left[\mathbf{f}(\mathbf{w}_i, \mathbf{s}_{i,u_i+1}), (X_i' \mathbf{1}_{u_i}), (X_i^{*'} \mathbf{1}_{u_i+1})\right]'$, and $X_i^{*'} \mathbf{1}_{u_i+1} = X_i' \mathbf{1}_{u_i} + \mathbf{f}(\mathbf{w}_i, \mathbf{s}_{i,u_i+1})$.

Next, we consider the case in which a design point is exchanged for a candidate point, both having the same whole plot factor levels. The design point is $(\mathbf{w}_i, \mathbf{s}_{ij})$,

whereas the candidate point to be exchanged is $(\mathbf{w}_i, \mathbf{s}_{ij}^*)$.

It is shown in Arnouts and Goos (2010) that:

$$|M^*| = |M| |\mathbf{T}_2| |\mathbf{T}_2^{-1} + \mathbf{U}_2 M^{-1} \mathbf{U}_2'|, \quad (\text{A.2})$$

and

$$(M^*)^{-1} = M^{-1} - M^{-1} \mathbf{U}_2' (\mathbf{T}_2^{-1} + \mathbf{U}_2 M^{-1} \mathbf{U}_2')^{-1} \mathbf{U}_2 M^{-1}, \quad (\text{A.3})$$

where $\mathbf{T}_2 = \text{diag}(-1, 1, \frac{d}{1+u_i d}, \frac{-d}{1+u_i d})$, $\mathbf{U}_2 = [\mathbf{f}(\mathbf{w}_i, \mathbf{s}_{ij}), \mathbf{f}(\mathbf{w}_i, \mathbf{s}_{ij}^*), (X_i' \mathbf{1}_{u_i}), (X_i^{*'} \mathbf{1}_{u_i})]$, $|\mathbf{T}_2| = \frac{d^2}{(1+u_i d)^2}$, and $X_i^{*'} \mathbf{1}_{u_i} = X_i' \mathbf{1}_{u_i} - \mathbf{f}(\mathbf{w}_i, \mathbf{s}_{ij}) + \mathbf{f}(\mathbf{w}_i, \mathbf{s}_{ij}^*)$.

Another way we perturb a split-plot design is by exchanging existing design points from two different whole plots (i and l). To perform such a swap, the whole plot factor levels for both should be the same (otherwise, the result would be whole plots with more than one level of whole plot factors).

$$|M^*| = |M| |\mathbf{T}_3| |\mathbf{T}_3^{-1} + \mathbf{U}_3 M^{-1} \mathbf{U}_3'|, \quad (\text{A.4})$$

and

$$(M^*)^{-1} = M^{-1} - M^{-1} \mathbf{U}_3' (\mathbf{T}_3^{-1} + \mathbf{U}_3 M^{-1} \mathbf{U}_3')^{-1} \mathbf{U}_3 M^{-1}, \quad (\text{A.5})$$

where $\mathbf{T}_3 = \text{diag}(\frac{d}{1+u_i d}, \frac{d}{1+u_l d}, \frac{-d}{1+u_i d}, \frac{-d}{1+u_l d})$, $\mathbf{U}_3 = [(X_i' \mathbf{1}_{u_i}), (X_l' \mathbf{1}_{u_i}), (X_i^{*'} \mathbf{1}_{u_i}), (X_l^{*'} \mathbf{1}_{u_i})]$, $|\mathbf{T}_3| = \frac{d^4}{(1+u_i d)^2 (1+u_l d)^2}$, $X_i^{*'} \mathbf{1}_{u_i} = X_i' \mathbf{1}_{u_i} - \mathbf{f}(\mathbf{w}_i, \mathbf{s}_{ij}) + \mathbf{f}(\mathbf{w}_l, \mathbf{s}_{lm})$, and $X_l^{*'} \mathbf{1}_{u_i} = X_l' \mathbf{1}_{u_i} - \mathbf{f}(\mathbf{w}_l, \mathbf{s}_{lm}) + \mathbf{f}(\mathbf{w}_i, \mathbf{s}_{ij})$.

If the j^{th} run in whole plot i is changed, all other runs in that whole plot will have to be changed as well because of the restriction that the whole plot factor levels be the same within a given whole plot. Letting \mathbf{w}^* be the updated whole plot factor levels, Arnouts and Goos (2010) show that the result of changing the whole plot

factor level for whole plot i is:

$$|M^*| = |M| |\mathbf{T}_4| |\mathbf{T}_4^{-1} + \mathbf{U}_4 M^{-1} \mathbf{U}_4'|, \quad (\text{A.6})$$

and

$$(M^*)^{-1} = M^{-1} - M^{-1} \mathbf{U}_4' (\mathbf{T}_4^{-1} + \mathbf{U}_4 M^{-1} \mathbf{U}_4')^{-1} \mathbf{U}_4 M^{-1}, \quad (\text{A.7})$$

where $\mathbf{T}_4 = \text{diag}(-I_{u_i}, I_{u_i}, \frac{d}{1+u_i d}, \frac{-d}{1+u_i d})$, $\mathbf{U}_4 = [X_i', X_i^{*'}, (X_i' \mathbf{1}_{u_i}), (X_i^{*'} \mathbf{1}_{u_i})]$, $|\mathbf{T}_4| = \frac{d^2}{(1+u_i d)^2} (-1)^{u_i+1}$, and $X_i^{*'} = [\mathbf{f}(\mathbf{w}_i^*, \mathbf{s}_{i1}), \dots, \mathbf{f}(\mathbf{w}_i^*, \mathbf{s}_{iu_i})]$.

Appendix B. Scaled-Product Model-Robust Split-Plot Exchange Algorithm

To describe our algorithm, we adopt the same basic structure (though different notation) as Goos and Vandebroek (2003): We assume c candidate points, the set of which is denoted by C . Similarly, we let the set of possible whole plot factor levels be P . There are b whole plots in the design and the set of them is B , and the u_i design points in the i^{th} whole plot for a design ξ_{bu} are denoted by H_i . Thus, the total number of design points is $n = \sum_{i=1}^b u_i$. The whole plot factor level combinations for the i^{th} whole plot are given by \mathbf{w}_i , and the set of candidate points with the same whole plot factor combinations as the i^{th} whole plot is given by C_i . The number of algorithm tries is given by t and the current try is t_c .

For the algorithm, we must specify 1) a candidate list, C ; 2) the number of whole plots, b ; 3) the size of each whole plot, $\mathbf{u} = (u_1, u_2, \dots, u_b)$; 4) the ratio of variance components, d ; 5) the set of r possible model forms, \mathcal{F} ; and 6) a model interest vector, \mathbf{v} , specifying the relative preference for each model (if no preferences, enter $v_f = 1$ for all $\mathbf{f} \in \mathcal{F}$). We also initialize $|M_f^*| = 0$ for all $f \in \mathcal{F}$, to serve as the initial benchmark against

which designs will be judged. Because our algorithm utilizes the prediction variance to assist in initial design construction, we give it here:

$$P(\mathbf{w}_i, \mathbf{s}_{ij}, \xi_{bu}) = \mathbf{f}'(\mathbf{w}_i, \mathbf{s}_{ij})M^{-1}(\xi_{bu}, d)f(\mathbf{w}_i, \mathbf{s}_{ij}). \quad (\text{B.1})$$

The algorithm is as follows.

1. Set $t_c = 1$.
2. Determine $p_{w,\mathbf{f}}$, the number of coefficients for whole plot factors only, for all $\mathbf{f} \in \mathcal{F}$.
3. Determine $p_{s,\mathbf{f}}$, the number of coefficients for sub-plot factors only, for all $\mathbf{f} \in \mathcal{F}$.
4. Set $M_{\mathbf{f}} = \omega I$ (where ω is a small constant set at 0.01) for all $\mathbf{f} \in \mathcal{F}$; set $H_i = \emptyset$.
5. Construct initial design:
 - (a) Randomly assign $p_{w,\mathbf{f}_{max}}$ unique whole plot factor settings to $p_{w,\mathbf{f}_{max}}$ whole plots, where \mathbf{f}_{max} is the model with the most parameters (to ensure estimability of whole plot coefficients for largest model).
 - (b) Randomly assign $b - p_{w,\mathbf{f}_{max}}$ levels of the whole plot factors to the rest of the whole plots.
 - (c) Randomly choose a ($1 \leq a \leq p_{\mathbf{f}_{max}}$), where $p_{\mathbf{f}_{max}}$ is the number of parameters for model \mathbf{f}_{max} .
 - (d) Do a times:
 - i. Randomly select $i \in B$ (select a whole plot at random)
 - ii. Randomly select $j \in C_i$ (select a candidate point with the whole plot setting corresponding to the i^{th} whole plot, at random)
 - iii. If $\#H_i < u_i$, then $H_i = H_i \cup j$; otherwise, go back to step i.

- iv. Update $M_{\mathbf{f}}^{-1}$ for all $\mathbf{f} \in \mathcal{F}$ via (A.1).
- (e) Do $n - a$ times (this step attempts to give initial design a measure of quality with respect to all models $\mathbf{f} \in \mathcal{F}$):
 - i. Randomly select $\mathbf{f} \in \mathcal{F}$.
 - ii. Set $l = 1$.
 - iii. For model \mathbf{f} , determine $j \in C$ with the l^{th} biggest prediction variance via (B.1).
 - iv. Find i , where $i \in B$, $j \in C_i$, and $\#H_i < u_i$ (find a nonfull whole plot into which j can be inserted). If no such i exists, set $l = l + 1$ and return to step iii.
 - v. $H_i = H_i \cup j$.
 - vi. Update $M_{\mathbf{f}}^{-1}$ for all $\mathbf{f} \in \mathcal{F}$ via (A.1).
6. Compute $M_{\mathbf{f}_r}$, the information matrix for the largest model, and determine whether it is invertible. If not, go back to step 4. Otherwise, use $M_{\mathbf{f}_r}$ to find the information matrices for all other models, calculate $M_{\mathbf{f}}^{-1}$ and $|M_{\mathbf{f}}|$ for all $\mathbf{f} \in \mathcal{F}$ and continue.
7. Set $\kappa = 0$.
8. Evaluate design point exchanges (swapping design points with candidate points, where whole plot factors settings are the same):
 - (a) Set $\gamma = 1$.
 - (b) $\forall i \in B, \forall j \in H_i, \forall k \in C_i, j \neq k$:
 - i. Determine the effect $\delta_{jk,\mathbf{f}}^i = |M_{jk,\mathbf{f}}^i|/|M_{\mathbf{f}}|$ of exchanging, in the i^{th} whole plot, points j and k , using (A.2), for all $\mathbf{f} \in \mathcal{F}$.
 - ii. If $\prod_{\mathbf{f} \in \mathcal{F}} \delta_{jk,\mathbf{f}}^i > \gamma$, then $\gamma = \prod_{\mathbf{f} \in \mathcal{F}} \delta_{jk,\mathbf{f}}^i$ and store i, j , and k .
9. If $\gamma > 1 + \epsilon$, then go to step 10; otherwise, go to step 11.
10. Perform the best exchange:
 - (a) $H_i = H_i \setminus j \cup k$.

- (b) Update $M_{\mathbf{f}}^{-1}$ and $|M_{\mathbf{f}}|$ for all $\mathbf{f} \in \mathcal{F}$ via (A.3) and (A.2), respectively.
- (c) Set $\kappa = 1$.
11. Evaluate interchanges of points within whole plots with the same factor levels:
- (a) Set $\gamma = 1$.
- (b) $\forall i, j \in B, i < j, \mathbf{w}_i = \mathbf{w}_j, \forall k \in H_i, \forall l \in H_j, k \neq l$:
- i. Determine the effect $\delta_{ik,\mathbf{f}}^{jl} = |M_{ik,\mathbf{f}}^{jl}|/|M_{\mathbf{f}}|$ of moving k to whole plot j (from whole plot i) and l to whole plot i (from whole plot j), via (A.4), for all $\mathbf{f} \in \mathcal{F}$.
 - ii. If $\prod_{\mathbf{f} \in \mathcal{F}} \delta_{ik,\mathbf{f}}^{jl} > \gamma$, then $\gamma = \prod_{\mathbf{f} \in \mathcal{F}} \delta_{ik,\mathbf{f}}^{jl}$ and store i, j, k , and l .
12. If $\gamma > 1 + \epsilon$, go to step 13; otherwise, go to step 14.
13. Perform the best interchange:
- (a) $H_i = H_i \setminus k \cup l$.
- (b) $H_j = H_j \setminus l \cup k$.
- (c) Recalculate $M_{\mathbf{f}}^{-1}$ and update $|M_{\mathbf{f}}|$ via (A.4).
- (d) Set $\kappa = 1$.
14. Evaluate exchanges of whole plot factor settings:
- (a) Set $\gamma = 1$.
- (b) $\forall i \in B, \forall j \in P, \mathbf{w}_i \neq \mathbf{w}_j$:
- i. Determine the effect $\delta_{ij,\mathbf{f}} = |M_{ij,\mathbf{f}}|/|M_{\mathbf{f}}|$ of exchanging \mathbf{w}_i by \mathbf{w}_j in the i^{th} whole plot, via (A.6), for all $\mathbf{f} \in \mathcal{F}$.
 - ii. If $\prod_{\mathbf{f} \in \mathcal{F}} \delta_{ij,\mathbf{f}} > \gamma$, then $\gamma = \prod_{\mathbf{f} \in \mathcal{F}} \delta_{ij,\mathbf{f}}$ and store i and j .
15. If $\gamma > 1 + \epsilon$, go to step 16; otherwise, go to step 17.
16. Perform best exchange:
- (a) Update H_i and C_i .
- (b) Update $M_{\mathbf{f}}^{-1}$ and $|M_{\mathbf{f}}|$ for all $\mathbf{f} \in \mathcal{F}$ via (A.7) and (A.6), respectively.
- (c) Set $\kappa = 1$.

17. If $\kappa = 1$, go to step 7.
18. If $\prod_{\mathbf{f} \in \mathcal{F}} |M_{\mathbf{f}}| > \prod_{\mathbf{f} \in \mathcal{F}} |M_{\mathbf{f}}^*|$, then $\xi_{bu}^* = \xi_{bu}$ and $|M_{\mathbf{f}}^*| = |M_{\mathbf{f}}|$ (where ξ_{bu}^* represents the best design found so far, and ξ_{bu} is the best design found for algorithm try t_c).
19. If $t_c < t$, then $t_c = t_c + 1$, and go back to step 4; otherwise, STOP.

Notes: 1) This algorithm, including Step 5 which generates the initial design, generalizes Goos and Vandebroek (2003) but follows its basic outline. 2) In the description above (and in Step 6 in particular), an assumption is made that the set of models is nested, which means that each successively larger model includes the previous as a subset. If this is not the case, it will not be possible to derive information matrices for all smaller models from the largest one. 3) In 13(c), we recalculate M^{-1} by directly taking its inverse, because of numerical issues associated with the updating formula in (A.5). This should have a minimal effect on the speed of the algorithm because it occurs only once per iteration. 4) In steps 9, 12, and 15, we have changed the original algorithm of Goos and Vandebroek slightly by requiring that, for any considered exchange/interchange, the increase in the minimum efficiency is greater than $\epsilon = 0.000001$ (instead of 0, as it was originally), to encourage algorithmic stability.

Anbari, F. T., Lucas, J. M., 1994. Super-efficient designs: How to run your experiment for higher efficiency and lower cost. In: ASQC 48th Annual Quality Congress Proceedings.

Anbari, F. T., Lucas, J. M., 2008. Design and running super-efficient experiments: Optimum blocking with one hard-to-change factor. Journal of Quality Technology 40(1), 31–45.

Arnouts, H., Goos, P., 2010. Update formulas for split-plot and block designs. Computational Statistics & Data Analysis 54(12), 3381–3391.

- Atkinson, A. C., Donev, A. N., Tobias, R. D., 2007. Optimum Experimental Designs, with SAS. Oxford University Press, New York.
- Bingham, D., Sitter, R. R., 1999. Minimum-aberration two-level fractional factorial split-plot designs. *Technometrics* 41(1), 62–70.
- Bingham, D. R., Schoen, E. D., Sitter, R. R., 2004. Designing fractional factorial split-plot experiments with few whole-plot factors. *Journal of the Royal Statistical Society, Series C* 53, 325–339.
- Bingham, D. R., Sitter, R. R., 2001. Design issues in fractional factorial split-plot experiments. *Journal of Quality Technology* 33(1), 2–15.
- Bisgaard, S., 2000. The design and analysis of $2^{k-p} \times 2^{q-r}$ split plot experiments. *Journal of Quality Technology* 32(1), 39–56.
- Box, G. E. P., Draper, N. R., 1959. A basis for the selection of a response surface design. *Journal of the American Statistical Association* 54(287), 622–654.
- Cook, R. D., Nachtsheim, C. J., 1982. An analysis of the k-exchange algorithm. Tech. rep., University of Minnesota, Dept. of Operations and Management Science.
- Cornell, J., 1990. Experiments with Mixtures: Designs, Models, and the Analysis of Mixture Data, 2nd Edition. Wiley-Interscience.
- Dette, H., 1990. A generalization of \mathcal{D} - and \mathcal{D}_1 -optimal design in polynomial regression. *Annals of Statistics* 18, 1784–1804.
- Dette, H., Franke, T., 2001. Robust designs for polynomial regression by maximizing a minimum of \mathcal{D} - and \mathcal{D}_1 -efficiencies. *The Annals of Statistics* 29(4), 1024–1049.
- Donev, A. N., Atkinson, A. C., 1988. An adjustment algorithm for the construction of exact D-optimum experimental designs. *Technometrics* 30, 429–433.
- Draper, N. R., John, J. A., 1998. Response surface designs where levels of some factors are difficult to change. *Australian and New Zealand Journal of Statistics* 40(4), 487–495.
- DuMouchel, W., Jones, B., 1994. A simple Bayesian modification of D-optimal designs to reduce dependence on an assumed model. *Technometrics* 36(1), 37–47.
- Fang, Z., Wiens, D. P., 2003. Robust regression designs for approximate polynomial models. *Journal of Statistical Planning and Inference* 117, 305–321.
- Gilmour, S. G., Trinca, L. A., 2012. Optimum design of experiments for statistical inference. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*.
- Goos, P., 2002. The Optimal Design of Blocked and Split-plot Experiments. Springer.
- Goos, P., 2006. Optimal versus orthogonal and equivalent-estimation design of blocked and split-plot experiments. *Statistica Neerlandica* 60, 361–378.
- Goos, P., Donev, A. N., 2007. Tailor-made split-plot designs for mixture and process variables. *Journal of Quality Technology* 39(4), 326–339.
- Goos, P., Kobilinsky, A., O’Brien, T. E., Vandebroek, M., 2005. Model-robust and model-sensitive designs. *Computational Statistics & Data Analysis* 49, 201–216.
- Goos, P., Vandebroek, M., 2001. Optimal split-plot designs. *The Journal of Quality Technology* 33(4), 436–450.
- Goos, P., Vandebroek, M., 2003. D-optimal split-plot designs with given numbers and sizes of whole plots. *Technometrics* 45(3), 235–245.
- Goos, P., Vandebroek, M., 2004. Outperforming completely randomized designs. *Journal of Quality Technology* 36(1), 12–26.
- Heredia-Langner, A., Montgomery, D. C., Carlyle, W. M., Borrór, C. M., 2004. Model-robust optimal designs: A genetic algorithm approach. *Journal of Quality Technology* 36(3), 263–279.
- Huang, P., Chen, D., Voelkel, J. O., 1998. Minimum-aberration two-level split-plot designs. *Technometrics* 40(4), 314–326.
- Jones, B., Goos, P., 2007. A candidate-set-free algorithm for generating D-optimal split-plot designs. *Journal of the Royal Statistical Society, Series C: Applied Statistics* 56, 347–364.
- Jones, B., Goos, P., January 2012. I-optimal versus d-optimal split-plot response surface designs. Working Paper 2012002, University of Antwerp, Faculty of Applied Economics.
URL <http://ideas.repec.org/p/ant/wpaper/2012002.html>
- Jones, B., Nachtsheim, C. J., 2009. Split-plot designs: What, why, and how. *Journal of Quality Technology* 41(4), 340–361.
- Kowalski, S. M., Cornell, J. A., Vining, G. G., 2002. Split-plot designs and estimation methods for mixture experiments with process variables. *Technometrics* 44(1), 72–79.
- Läuter, E., 1974. Experimental design in a class of models. *Mathematische Operationsforschung und Statistik* 5, 379–398.
- Letsinger, J. D., Myers, R. H., Lentner, M., 1996. Response surface methods for bi-randomization structures. *Journal of Quality Technology* 28:4, 381–397.
- Li, W., Nachtsheim, C. J., 2000. Model-robust factorial designs. *Technometrics* 42(4), 345–352.
- Lucas, J. M., 1974. Optimum composite designs. *Technometrics* 16, 561–567.
- Lucas, J. M., Ju, H. L., 1992. Split plotting and randomization in in-

- dustrial experiments. In: ASQC Quality Congress Transactions.
- Parker, P. A., Kowalski, S. M., Vining, G. G., 2006. Classes of split-plot response surface designs for equivalent estimation. *Quality and Reliability Engineering International* 22(3), 291–305.
- Parker, P. A., Kowalski, S. M., Vining, G. G., 2007. Construction of balanced equivalent estimation second-order split-plot designs. *Technometrics* 49(1), 56–65.
- Piepel, G. F., Cooley, S. K., Jones, B., 2005. Construction of a 21-component layered mixture experiment design using a new mixture coordinate-exchange algorithm. *Quality Engineering* 17, 579–594.
- Smucker, B. J., del Castillo, E., Rosenberger, J. L., 2011. Exchange algorithms for constructing model-robust experimental designs. *Journal of Quality Technology* 43(1), 28–42.
- Smucker, B. J., del Castillo, E., Rosenberger, J. L., 2012. Model-robust two-level designs using coordinate exchange algorithms and a maximin criterion. Tentatively accepted to *Technometrics*.
- Stigler, S. M., 1971. Optimal experimental design for polynomial regression. *Journal of the American Statistical Association* 66(334), 311–318.
- Studden, W. J., 1982. Some robust-type D-optimal designs in polynomial regression. *Journal of the American Statistical Association* 77(380), 916–921.
- Trinca, L. A., Gilmour, S. G., 2001. Multistratum response surface designs. *Technometrics* 43(1), 25–33.
- Tsai, P.-W., Gilmour, S. G., 2010. A general criterion for factorial designs under model uncertainty. *Technometrics* 52(2), 231–242.
- Vining, G. G., Kowalski, S. M., Montgomery, D. C., 2005. Response surface designs within a split-plot structure. *Journal of Quality Technology* 37(2), 115–129.
- Welch, W. J., 1983. A mean squared error criterion for the design of experiments. *Biometrika* 70(1), 205–213.
- Yates, F., 1935. Complex experiments, with discussion. *Journal of the Royal Statistical Society, Series B* 2, 181–223.