DEPARTMENT OF ENGINEERING MANAGEMENT

Two-level designs to estimate all main effects and two-factor interactions

Pieter T. Eendebak & Eric D. Schoen

UNIVERSITY OF ANTWERP
Faculty of Applied Economics

City Campus
Prinsstraat 13, B.226
B-2000 Antwerp
Tel. +32 (0)3 265 40 32
Fax +32 (0)3 265 47 99
www.uantwerpen.be

FACULTY OF APPLIED ECONOMICS

DEPARTMENT OF ENGINEERING MANAGEMENT

Two-level designs to estimate all main effects and two-factor interactions

Pieter T. Eendebak & Eric D. Schoen

RESEARCH PAPER 2015-019
NOVEMBER 2015

University of Antwerp, City Campus, Prinsstraat 13, B-2000 Antwerp, Belgium
Research Administration – room B.226
phone: (32) 3 265 40 32
fax: (32) 3 265 47 99
e-mail: joeri.nys@uantwerpen.be

The research papers from the Faculty of Applied Economics are also available at www.repec.org (Research Papers in Economics - RePEc)

D/2015/1169/019

Two-level designs to estimate all main effects and two-factor interactions

Pieter T. Eendebak[1], Eric D. Schoen[1,2]

[1]University of Antwerp, Belgium
[2]TNO, Zeist, Netherlands

November 17, 2015

Abstract

We study the design of two-level screening experiments with N runs and n factors large enough to estimate a model with all the main effects and all the two-factor interactions, while yet an effect hierarchy assumption suggests that main effect estimation should be given more prominence than the estimation of two-factor interactions. Orthogonal arrays (OAs) favor main effect estimation. However, complete enumeration becomes infeasible for cases relevant for practitioners. We develop a partial enumeration procedure for these cases and we establish upper bounds on the D-efficiency of arrays that have not been generated by the partial enumeration. We propose an optimal design procedure that favors main effect estimation as well. Designs created with this procedure have smaller D-efficiencies than D-optimal designs, but standard errors for main effects are improved. Generated OAs for 7–10 factors and 32–72 runs are smaller or have a higher D-efficiency than the smallest OAs from the literature. Designs obtained with the new optimal design procedure or strength-3 OAs (which have main effects that are not correlated with two-factor interactions) are recommended under effect hierarchy. D-optimal designs are recommended if this assumption is not likely to hold.

KEY WORDS: coordinate exchange; D-efficiency; optimal design; orthogonal array; partial enumeration

1 Introduction

Experimenters using two-level factorial experiments usually think of the data as being generated from an additive model with main effects, two-factor interactions and higher-order interactions. To structure the analysis, they assume that main effects are more important than two-factor interactions, while two-factor interactions are more important than higher-order interactions. The assumption, called effect hierarchy, was coined first in the textbook Wu and Hamada (2000).

Empirical evidence in support of the effect hierarchy assumption was given by Li et al. (2006). These authors considered 46 two-level experiments with 3–7 factors. They found that the median main effect size was four times larger than the median size of the two-factor interactions and eight times larger than the median three-factor interactions. In addition, about 40% of the main effects were active, as opposed to 11% of the two-factor interactions and 6.8 % of the three-factor interactions. While three-factor interactions evidently cannot be ruled out, including only main effects and two-factor interactions in a model for responses from two-level experiments seems a reasonable first approach.

In this paper, we consider the design of two-level experiments large enough to estimate a model with all the main effects and all the two-factor interactions, while yet the effect hierarchy

assumption suggests that main effects should be given more prominence than two-factor interactions. An example of this type of experiment was carried out recently at TNO, Eindhoven, the Netherlands. The experiment was concerned with the making of phantoms to calibrate medical devices. Phantoms are cylindrical pieces of gelatinous material that mimic human tissues; these tissues are to be investigated with the device once it is properly calibrated. A phantom is tested by exposing it to light of various wavelengths. For each of the wavelengths, the reflection is recorded, which can be affected by the concentrations of seven colorants. The main interest was in the size of the factorial effects. Only a few of the colorants are expected to be active for any given wave length. Further, optical laws suggest that main effects are much more prominent than interaction effects. The experimental budget permitted construction of as many as 40 phantoms. Clearly, this number should be sufficient to construct a model with an intercept, all seven main effects and all 21 two-factor interactions. In the rest of this paper, we call such a model the interaction model. The purpose of this paper is to develop procedures for generating designs that can fit the interaction model with good D-efficiencies (to be defined formally later in the paper), while giving the main effect estimation more prominence than the estimation of the two-factor interactions.

Design alternatives that might be considered for the phantom experiment include orthogonal arrays (OAs) and D-optimal designs. We contributed to the development of both types of design. In the rest of this section, we provide more details on these types and we outline the further organization of the paper.

1.1 Orthogonal arrays

Generally, an OA of strength t, N runs and n factors at s levels is an $N \times n$ array of s symbols such that for every t columns every s^t t-tuple occurs equally often (Rao, 1947; Hedayat et al., 1999). Such an array is denoted $\mathrm{OA}(N, n, s, t)$. Our present interest is in arrays with $s = 2$, and we omit the reference to the number of levels of the factors in the notation for an OA.

An attractive feature of OAs is that main effects have the maximum possible precision. Therefore, OAs meet the above effect hierarchy assumption. However, the extent to which the assumption is met depends on the strength of the OA.

OAs of strength 4 are D-optimal for the interaction model, because all subsets of four factors form an equally replicated full factorial design. For this reason, all main effect contrast vectors and all two-factor interaction contrast vectors are orthogonal to each other, and both the main effect estimators and two-factor interaction estimators have a maximum precision.

A disadvantage of these arrays is their run size. For the seven factor phantom design, an OA of strength 4 requires 64 runs, which is a substantial larger than both the experimental budget of 40 runs and the number of parameters in the interaction model, which equals 29. At the same time, the effect hierarchy assumption suggests that it is not important that all the factorial effects have maximum precision. It is therefore natural to study OAs of strength $t < 4$ capable of fitting the interaction model with smaller run sizes than a strength-4 array.

OAs of strength 3 retain mutual independence of main effects and independence of main effects with interactions. Therefore, main effects are estimated with maximum precision. The estimators of two-factor interactions are correlated. Therefore, at least some of these interactions are not estimable with maximum precision in a full interaction model. This need not be a problem if the effect hierarchy assumption holds, however. It is therefore of interest to study strength-3 OAs with maximum D-efficiencies.

OAs of strength 2 have orthogonal main effect contrast vectors, but these are correlated with the contrast vectors of two-factor interactions. Therefore, the main effect estimators have maximum precision only in a first-order model. At the same time, the D-efficiencies for the interaction model can be higher than in strength-3 arrays, because the combinatorial restrictions are less severe.

A naive way to find out OAs of strength 2 or 3 with the best possible D-efficiency is to enumerate all $\mathrm{OA}(N, n, t)$, and to check subsequently their D-efficiencies. The problem one is faced with in carrying out such a procedure is the large number of different designs. For example,

we were able to establish a set of 530,469,996 OA$(32, 7, 2)$. Any OA$(32, 7, 2)$ not in the set can be obtained from an array in the set by a sequence of column permutations, row permutations or level switches in a column. The arrays in the set cannot be obtained from each other by such a sequence of permutations. There are five OAs with the best D-efficiency for the interaction model; the efficiency value is 0.8432.

The enumeration of the set of OA$(32, 7, 2)$ took about 7 days on a PC with an Intel Core i7 870 CPU at 2.93GHz. It is computationally infeasible to enumerate all OA$(N, n, 2)$ for $N \geq 36$ and $n \geq 6$. Similarly, it is not feasible to enumerate all OA$(N, n, 3)$ for $N \geq 64$ and $n \geq 8$. The first contribution of this paper is the introduction of a partial enumeration procedure for cases with $t \leq 3$ where a complete enumeration is not feasible and to introduce a simple method to establish upper bounds on the D-efficiency of arrays that have not been generated by the partial enumeration. Our partial enumeration of OA$(32, 7, 2)$ took just 5 hours of computing time, produced all five D-optimal arrays and resulted in an upper bound of 0.8799 for D-efficiencies in arrays that were not generated.

1.2 Optimal designs

In the D-optimal approach, a design of N runs and n factors is sought that maximizes the D-efficiency of the interaction model (Atkinson et al., 2007). Because no combinatorial restrictions are imposed, the D-efficiency of a D-optimal design for this model will generally be higher than the D-efficiency of the best OAs under this model. However, such a D-optimal design does not support effect hierarchy. For example, we generated a D-optimal design for the phantom experiment with standard errors for the main effects in the range (0.1614; 0.1667), while those for the two-factor interactions are in the same range. The second contribution of this paper is the development of an optimal design procedure that favors the main effect estimation. For the phantom example, we created a design with standard errors for main effects ranging from 0.1581 to 0.1614. The standard errors for two-factor interactions are accordingly less precise: they range from 0.1637 to 0.2083.

1.3 Organization

The rest of this paper is organized as follows. In Section 2, we return to the motivating example in more detail. We consider four different candidate designs and introduce design measures to characterize the designs. In Section 3, we introduce the enumeration algorithm for OAs and the optimal design algorithm for D-efficient designs that favor main effect estimation. In Section 4, we detail the numbers and best efficiencies of the generated OAs, give upper bounds for those that might have been obtained by enumeration of the complete set, and contrast these with efficiencies obtained by optimal design algorithms. Next, we study in detail the statistical properties of the best designs for up to 72 runs and up to 10 factors and compare these with the best literature designs known to us. Finally, there is a brief discussion of the strengths and weaknesses of our approach in Section 6. Software to generate orthogonal arrays and optimal designs is provided on request.

2 Optimality measures and candidate designs

In this section, we introduce three optimality measures for designs that fit the interaction model. We illustrate these measures with four candidate designs for the phantom experiment.

2.1 Optimality measures

The interaction model based on a two-level design A can be stated formally as $y = \mathbf{X}\beta + e$, where y is an $N \times 1$ vector of responses and $\mathbf{X}$ an $N \times p$ model matrix with an intercept, n main effect contrast vectors and $n(n-1)/2$ two-factor interaction contrast vectors. Finally, β is the

$p \times 1$ vector of the factorial effects and e is an $N \times 1$ vector of random errors with expectation zero and variance σ^2.

The parameters of the model can be estimated with the OLS estimator b of β with $b = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T y$; its covariance matrix is $(\mathbf{X}^T\mathbf{X})^{-1}\sigma^2$. Therefore, the capability of a design to return precise estimates of the factorial effects is maximized if $\mathbf{X}^T\mathbf{X}$ is maximized in some sense. One meaningful way is the maximization of $|\mathbf{X}^T\mathbf{X}|$, because maximizing this determinant minimizes the volume of a joint confidence region of the parameters under normal distribution of the random error e (Atkinson et al., 2007).

For convenience, the determinant $|\mathbf{X}^T\mathbf{X}|$ is scaled by the number of parameters in the model and the run size. The scaled version of the determinant is designated $\mathrm{D}(A)$, with $\mathrm{D}(A) = |\mathbf{X}^T\mathbf{X}/N|^{1/p}$. We call $\mathrm{D}(A)$ the D-efficiency of A, thereby omitting the reference to the interaction model. It is well known that $0 \leq \mathrm{D}(A) \leq 1$. $\mathrm{D}(A) = 0$ if and only if the columns of $\mathbf{X}$ are linearly dependent, while orthogonal columns of $\mathbf{X}$ give a D-efficiency of 1.

To address the joint precision of the main effects in the interaction model we slightly modify a criterion used by Schoen (2010) based on the concept of D_s-optimality (Atkinson et al., 2007). For this purpose, we divide the parameter vector β in a vector β_1 with main effect coefficients and a vector β_{02} with the coefficients for intercept and the two-factor interactions. The model matrix $\mathbf{X}$ is split in an analogous way into $\mathbf{X_1}$ and $\mathbf{X_{02}}$ so that $y = \mathbf{X_1}\beta_1 + \mathbf{X_{02}}\beta_{02} + e$. A D_s optimal design maximizes

$$\mathcal{D}_s = |\mathbf{X}^T\mathbf{X}|/|\mathbf{X_{02}}^T\mathbf{X_{02}}|, \tag{1}$$

assuming that $\mathbf{X}_{02}$ is of maximum rank. It is easy to show that

$$|\mathbf{X}^T\mathbf{X}|/|\mathbf{X_{02}}^T\mathbf{X_{02}}| = |\mathbf{X}_1^T(I - \mathbf{X_{02}}(\mathbf{X_{02}}^T\mathbf{X_{02}})^{-1}\mathbf{X_{02}}^T)\mathbf{X_1}|. \tag{2}$$

The right hand side of (2) is the determinant of the residual sums of squares and products matrix after regressing the main effects collected in $\mathbf{X_1}$ on the intercept and two-factor interactions collected in $\mathbf{X_{02}}$. The scaled version of the determinant is designated $D_s(A)$, with $D_s(A) = \mathcal{D}_s^{1/n}$. In the rest of the paper, we call $D_s(A)$ the D_s-efficiency of A.

If $\mathbf{X_{02}}$ is indeed of maximum rank, $0 \leq D_s(A) \leq 1$. $D_s(A) = 0$ if the columns of $\mathbf{X}$ are linearly dependent, while $D_s(A) = 1$ if the main effect columns of $\mathbf{X_1}$ are orthogonal to each other and also orthogonal to the intercept and two-factor interaction columns in $\mathbf{X_{02}}$.

It might seem unusual to maximize a determinant for main effect contrast vectors after accounting for two-factor interaction contrast vectors, as carried out in (2), because this reverses the roles of main effects and two-factor interactions. Indeed, interactions are defined as the part of the joint effect of factors left over when main effects are accounted for. We nevertheless think that the D_s criterion is useful as a design selection criterion in case main effects are of primary interest, because of the following argument.

Consider the main effect of a factor, F, say, in an interaction model. The two-factor interactions can be split into those involving F and those not involving F. If the interactions involving F are substantial, the main effect of F cannot be interpreted on its own. However, the effect hierarchy principle suggests that the interactions involving F may well be inactive, and the interactions involving F can be dropped. In case the design is level balanced, all these interactions are orthogonal to the main effect of F, and the standard error of this main effect will not change by dropping the interactions. To optimize the standard error of the main effect, the main effect contrast vector of F should be as much as possible orthogonal to the main effect contrast vectors of the other factors as well as to the interaction contrast vectors not involving F. The D_s criterion captures this kind of orthogonality.

Finally, our third optimality criterion, designated D_1-efficiency, is defined as $D_1(A) = |\mathbf{X_1}^T\mathbf{X_1}/N|^{1/(n+1)}$, where $\mathbf{X_1}$ is the model matrix with intercept and main effect contrast vectors. This criterion addresses the special case when only the main effects are active.

2.2 Candidate designs for the motivating example

To illustrate the optimality criteria outlined in the previous section, we introduce four candidate designs for the phantom case with 40 runs and 7 factors.

1. Using a complete set of OA$(40, 7, 3)$, we establish that the OA reported by Schoen and Mee (2012) is the only strength-3 OA of this size capable of fitting the interaction model. Its D-efficiency equals 0.8030.

2. Using a procedure that is discussed later in the paper, we generated 300 D-efficient OAs of strength 2 and include the most D-efficient array as a candidate design.

3. Using a coordinate change algorithm, we generated a D-optimal design.

4. Using another procedure discussed later in the paper, we generated designs with $\mathrm{D} + 2\mathrm{D}_s$ as optimality criterion. We include the best design according to this criterion as the fourth candidate; the design is designated compromise design.

An overview of the various efficiency measures for the candidate designs is given in Table 1. Orthogonal arrays necessarily have D_1-efficiencies equalling 1. However, the high D_1-efficiencies for designs from the optimal design procedures show that these are nearly orthogonal.

OAs of strength 3 are D_s-optimal. This follows from equation (2), because $\mathbf{X_{02}}^T\mathbf{X_1} = \mathbf{0}$. The high D_s-efficiency for the compromise design shows that this design has near orthogonality of the main effects with respect to the interactions.

The D-efficiency of the strength-3 candidate is substantially smaller than the D-efficiency of the D-optimal design, while the D_s-efficiency of the D-optimal design is worse than the D_s-efficiency of the strength-3 design. The compromise design is indeed a compromise as it has an improved D-efficiency when compared to the strength-3 design and an improved D_s-efficiency when compared to the D-optimal design.

The strength-2 candidate design, while its D-efficiency is substantially better than that of the strength-3 design, seems inferior to the D-optimal design. The D_1-efficiency of the latter design is only slightly less, while its D-efficiency and D_s-efficiency both are better.

To illustrate the connection between the various optimality criteria and the precision of main effects and interactions, we present boxplots of the standard errors of the coefficients in Figure 1, assuming an error variance of 1. We consider two model classes. The first one is the single full interaction model in seven factors. The second class consists of the seven models where all interactions of one particular factor are dropped from the full interaction model. We call these models reduced models.

The upper panel of the figure shows the standard errors of the main effects. There are four pairs of boxplots, one pair for each candidate design. Each broad boxplot shows the seven standard errors for the full interaction model based on the respective designs. Each narrow box shows the 42 standard errors for all the main effects in the reduced models.

The minimum standard error is $1/\sqrt{40}$, equalling about 0.1581. All main effects of the strength-3 option and three of the main effects in the compromise option have this minimum value. The compromise design has main effect standard errors between those of the D-optimal

Table 1: Efficiencies of four candidate designs for the motivating example.

Design	D-efficiency	D_s-efficiency	D_1-efficiency
strength 3	0.8030	1	1
strength 2	0.9245	0.8495	1
D-optimal	0.9534	0.9343	0.9864
compromise	0.8875	0.9884	0.9898

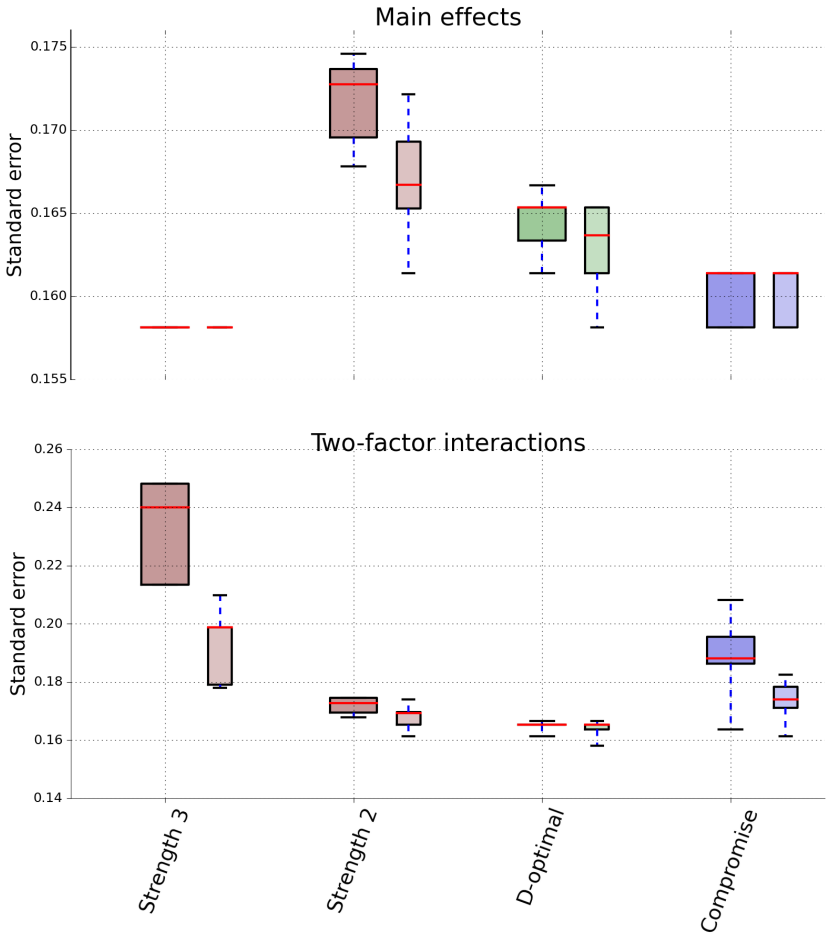


Figure 1: Standard errors in interaction models based on four designs of 40 runs and 7 factors. Broad boxes: full interaction model; narrow boxes: seven models lacking interactions of one of the factors.

design and the strength-3 design. The strength-2 design has the worst values of the main effect standard errors.

As expected, there is no change in main effect standard errors for the reduced models based on the strength-3 option. Remarkably, this is also the case for the compromise design. The standard errors for the strength-2 option are considerably improved, while those of the D-optimal option are also improved. The ranking of the four options regarding the main effect standard errors does not change, however.

The lower panel in the figure shows standard errors of the 21 interaction coefficients in the interaction model (broad boxplots) and of the 105 coefficients in the seven reduced models (narrow boxplots). The D-optimal design is clearly superior here, while the compromise design is intermediate between the D-optimal design and the strength-3 design. The standard errors in the reduced models of the strength-3 and compromise designs are considerably lower than those in the full interaction model. This shows the value of these designs to fit interactions under effect hierarchy.

The design actually used for the phantom experiment was the strength-3 option. Analysis of eight response variables (not shown) revealed that, depending on the response variable, there were 2-7 active interactions and 4-6 active main effects. Median effect sizes for interactions were roughly between 0.4 and 0.7 times the estimated standard deviation of individual observations,

while those for the main effects were roughly between 0.9 and 2.2 times this standard deviation. These findings show that effect hierarchy assumptions were in place here.

We would prefer the D-optimal design for cases where effect hierarchy is not likely to hold. This would be the case when the focus is on the search for interactions among a limited number of factors known to be active. The strength-2 alternative has little to add, because it is outperformed by the three other options for the standard errors of the main effects and by the D-optimal design for those of the interactions. If the compromise design had been known in time, we might have recommended that design for the phantom experiment.

3 Generation of designs

3.1 Orthogonal arrays

We want to generate OAs with good D-efficiency for the interaction model. Earlier work on D-efficient OAs (Tang and Zhou, 2013) is restricted to strength-2 OAs for the special case that the OA is embedded in a saturated OA with N runs and $N-1$ factors, while there are only a few specified interactions of interest. In this paper, we consider the case that all interactions are of equal interest, while the OA need not be embedded in a saturated OA. To generate OAs, we slightly modified the algorithm of Schoen et al. (2010) (SEN). The complete source for the system is available on the world wide web (Eendebak, 2015). Here, we review the key elements of the original algorithm. Its goal is to obtain a set of all $\mathrm{OA}(N, n, t)$. For any specific set of parameters t, N and n, there may be many arrays. These can be partitioned in isomorphism classes. All arrays within one isomorphism class can be obtained from each other by a sequence of row permutations, column permutations or level permutations. These arrays are mathematically and statistically equivalent. Therefore, it suffices to study only one instance of every isomorphism class.

The algorithm of SEN features lexicographic ordering of arrays. An array Q is lexicographically smaller than an array R if there exists a column index $k > 0$ such that $Q_k < R_k$, whereas $Q_i = R_i, i = 1, \ldots, k-1$. Here, $Q_k < R_k$ if there exists a row index $m > 0$ such that $Q_{mk} < R_{mk}$, while $Q_{jk} = R_{jk}, j = 1, \ldots, m-1$. So, reading column-wise, the first element for which Q and R differ has a smaller value in Q.

For any set of parameters of an OA, the algorithm produces a minimum complete set of arrays. This is a set with one unique representative array for each isomorphism class called the lexicographically minimal (LM) array.

Definition 1. An array is lexicographically minimal (LM) in its isomorphism class if no row, column, or level permutation results in a lexicographically smaller array.

To generate a minimum complete set of $\mathrm{OA}(N, n, t)$, the algorithm starts with a single array with t columns in lexicographically minimal form, which is called the root array. This array is a representative for the single isomorphism class in $\mathrm{OA}(N, t, t)$. Two further steps turn a minimum complete set of arrays with $t \leq k \leq n-1$ columns into a minimum complete set with $k+1$ columns:

1. *Extension*: for each array in the minimum complete set with k factors, a set of extensions with the required strength is generated that is guaranteed to contain all LM arrays that can be reached from the original array.

2. *LM check*: for each generated array, a test is performed to check whether the array is in LM form or not. The arrays not in LM form are rejected.

SEN show that a repeated application of the two steps results in a minimal complete set of $\mathrm{OA}(N, n, t)$. The arrays generated by these authors include all $\mathrm{OA}(N, n, 2)$ with $N \leq 28$ and $n \leq 6$ and all $\mathrm{OA}(N, n, 3)$ with $N \leq 48$. So, strength-2 alternatives to the well known $\mathrm{OA}(32, 6, 5)$ with run sizes up to 28 can be found by searching through the list of designs that

they generated. Similarly, Schoen and Mee (2012) found strength-3 alternatives to OA$(64, 7, 4)$, OA$(64, 8, 4)$ and OA$(128, 9, 4)$ with run sizes up to 48 by searching through the list of strength-3 designs.

To address strength-2 cases with $N \geq 32$ and strength-3 cases with $N \geq 56$, we restrict the extension of arrays in the minimal complete sets. First, we partition each set in arrays that permit fitting the interaction model and those that do not. We only extend the arrays of the first set. The minimal complete set with extended arrays is guaranteed to contain the D-optimal array.

If there are many arrays that permit estimation of the interaction model we applied a further restriction. We order the arrays according to their D-efficiencies and we extend only the best designs with an additional column. There is no guarantee that the set thus generated contains the D-optimal array. However, it is possible to establish upper bounds for the best possible D-efficiency of arrays that might have been generated based on the best efficiencies of the arrays that were not extended. These bounds are based on two theorems. The first one predicts what might happen if an array is extended with one extra column. The result is as follows:

Theorem 1. *Let A be an orthogonal array with N rows and k columns that can fit the interaction model. Let $P = [A\, E]$ be an array that results from extending A with a single column E. Let $p_k = 1 + k + k(k-1)/2$. Then $\mathrm{D}(P) \leq \mathrm{D}(A)^{p_k/p_{k+1}}$.*

The purpose of Theorem 2 is to sharpen the bounds established by Theorem 1 by taking previous extensions into account.

Theorem 2. *Let A be an orthogonal array in LM form. Define $\mathcal{D}(A) = |\mathbf{X}^T\mathbf{X}/N|$. Let $P_i = [A\, E_i]$ be an LM orthogonal array that results from extending A with a single column E_i. Let Q be an array that results from extending P_i with q columns and let $L_i = \mathcal{D}(P_i)/\mathcal{D}(A)$. Then*

$$\mathcal{D}(Q) \leq L^q \mathcal{D}(P_i) \tag{3}$$

with $L = \max_j L_j$. The maximum is over all possible LM extensions P_j derived from A.

Appendix A includes the proofs of the theorems. An illustration of the way that the bounds work out is given in Appendix B.

Cheng et al. (2002) established an approximate relation between the average D-efficiency of a model containing all the main effects and g two-factor interactions, and the first two elements of the generalized word length pattern (GWLP; Tang and Deng, 1999). The GWLP of an OA is a vector $(A_3, A_4, \ldots, A_n)$, where A_i is the sum of squared correlations between i-factor interaction contrast vectors and the intercept. In case $g = 0.5n(n-1)$, there is just one D-efficiency to consider; the relation for this case is $1/\mathrm{D} \propto 6A_3 + 6A_4$. In Appendix C, we confirm that the best D-efficiencies indeed are found for designs with small $A_3 + A_4$. It is important to note that the relation between D and A_3 on its own is much weaker. So a minimum G_2-aberration design, which minimizes the elements of the GWLP from left to right, does not necessarily have the best D-efficiency.

3.2 Optimal designs

We implemented a coordinate exchange algorithm in Python and Matlab. The algorithm is slightly more complicated than the original algorithm of Meyer and Nachtsheim (1995). It optimizes $O = \alpha_1 \mathrm{D} + \alpha_2 \mathrm{D}_s$, where D and D_s are defined in Section 2.1. A specification of the algorithm is given in Appendix D. The implementations are available on request.

For all cases where we generated D-efficient orthogonal arrays, we also generated D-optimal designs for the interaction model using the Python implementation with $\alpha_2 = 0$ and 5,000 initial tries. To generate compromise designs, we need to set the parameters α_1 and α_2 in our exchange algorithm. For this purpose, we set $\alpha_1 = 1$ and made seven-factor designs in 40 runs for α_2 ranging from 0 up to 6 in steps of 0.2. We repeated the process with eight-factor designs in 80 runs. We found that the D_s-efficiencies of the designs became stable for $\alpha_2 \geq 2$, while, for the 80

run designs, the D-efficencies slightly decreased from that value onward. For these reasons, we used $\alpha_1 = 1$ and $\alpha_2 = 2$ to construct the compromise designs. See Appendix D for more details.

Our compromise designs are intended for situations where main effects are more likely to be important than two-factor interaction effects. They permit efficient estimation of the full interaction model, because the D-efficiency is included in the goal function to be optimized. At the same time, they favor estimation of main effects by the inclusion of D_s in the goal function. Therefore, they can be considered as model-robust designs. The most important difference between our approach and earlier approaches to model robust designs is that the run sizes we consider permit estimation of the full interaction model. Therefore, no special attention is needed to account for nonestimable models, such as in DuMouchel and Jones (1994), Li and Nachtsheim (2000) and Smucker et al. (2012), or aliasing between primary and potential model terms such as in Jones and Nachtsheim (2011).

4 Generated designs

We generated D-efficient OAs of strength 2 and strength 3, D-optimal designs and compromise designs.

The strength-2 OAs we generated have 6–8 factors and 32–44 runs. An OA to estimate the interaction model in 9 factors requires at least 48 runs. It was infeasible to do even a partial enumeration of strength-2 arrays with this run size or larger run sizes because of the very large numbers of nonisomorphic arrays (even for the extension of an array with a single column).

As regards OAs with a strength $t \geq 3$, there is a single $\mathrm{OA}(32, 6, 5)$, which naturally permits estimation of the interaction model with maximum D-efficiency. For $n \geq 7$, strength-3 arrays for the interaction model only exist for run sizes $N \geq 40$. We generated D-efficient OAs of strength 3 in up to 10 factors requiring up to 72 runs.

Unlike OAs, D-optimal designs and compromise designs are not restricted to run sizes that equal multiples of 4 (strength 2 OAs) or 8 (strength-3 OAs). However, for direct comparisons with OAs, we generated optimal and compromise designs for run sizes $28 \leq N \leq 72$ equalling a multiple of 4.

4.1 Strength-2 arrays and alternative designs

Table 5 shows results for the strength-2 OAs and alternative designs. For the alternatives to OAs, we used our optimal design software with 5,000 initial tries and we kept the best design either according to D-efficiency or to the compound criterion $\mathrm{D} + 2\mathrm{D}_s$. The OA series with five factors were fully generated (results not shown). OA series with $n > 5$ factors were partially generated by extending only a small fraction of the designs with $n - 1$ factors based on their D-efficiency. To get an appreciation of the arrays that were missed in a partial generation, we studied for $N = 32$ fully generated as well as partially generated series of OAs. In Appendix E, we show details of the number of OAs generated, and the cut-offs we used.

The first two columns in each half of Table 5 give the run size N and the number of factors n. The third column shows the type of design, which is either an OA, a D-optimal design or a compromise design that has an optimized value of $\mathrm{D} + 2\mathrm{D}_s$. Then, we show the the D-efficiency of the designs. For OAs we subsequently provide an upper bound on D-efficiencies for arrays that might have been obtained if the series were generated fully. This bound, designated B, was obtained using Theorem 3 and Theorem 4 of Section 3.1. Finally, the last column in each half of Table 5 shows the D_s-efficiencies obtained.

All OAs generated have a D_1-efficiency equalling one. The designs generated with the optimal design software were all nearly orthogonal; the smallest D_1-efficiency equalled 0.9761. For this reason, we did not include such efficiencies in Table 5.

The results for 32 runs and six factors show that the best designs obtained with any of the three methods all have a D-efficiency equalling 1. They correspond to the unique $\mathrm{OA}(32, 6, 5)$. The fact that the optimal design software returns a design that is known to be best in terms

Table 2: Strength-2 arrays and alternative designs

N	n	Type	D	B	D_s	N	n	Type	D	B	D_s
32	6	OA	1	1	1	40	7	OA	0.9245	0.9414	0.8495
		D-optimal	1		1			D-optimal	0.9534		0.9343
		compromise	1		1			compromise	0.8875		0.9884
32	7	OA	0.8432	0.8432	0.8131	40	8	OA	0.8019	0.9516	0.6411
		D-optimal	0.8868		0.8325			D-optimal	0.8517		0.6967
		compromise	0.8033		0.9406			compromise	0.7463		0.9734
36	6	OA	0.9374	0.9374	0.8713	44	6	OA	0.9622	0.9622	0.9242
		D-optimal	0.9773		0.9659			D-optimal	0.9770		0.9595
		compromise	0.9743		0.9884			compromise	0.9602		0.9912
36	7	OA	0.9022	0.9389	0.8000	44	7	OA	0.9449	0.9531	0.8926
		D-optimal	0.9369		0.9506			D-optimal	0.9563		0.9381
		compromise	0.8716		0.9836			compromise	0.9113		0.9895
40	6	OA	0.9657	0.9657	0.9333	44	8	OA	0.8524	0.9567	0.7789
		D-optimal	0.9695		0.9549			D-optimal	0.8800		0.8010
		compromise	0.9601		0.9865			compromise	0.8034		0.9796

of D-efficiency and D_s-efficiency shows that the implementation is sufficiently powerful to find efficient designs.

There are eight partially generated series of OAs in the table. Although the generation is only partial, all the six-factor series include an array with the best possible D-efficiency of the interaction model. The seven-factor series have discrepancies of at most 0.0367 between the D-efficiency in the best array obtained and the upper bound.

For the best OA$(40, 8, 2)$ and the best OA$(44, 8, 2)$, the upper bound B is higher by 0.1487 and 0.1043, respectively. These are substantial discrepancies. However, in both cases, the D-optimal design generated has a higher D-efficiency by 0.0498 and 0.0276, respectively. This suggests that the upper bound for these two instances is particularly weak. Further, we show in the next section that the best OAs actually obtained are competitive with the best strength-2 arrays known from the literature. We therefore did not search for better OAs.

As expected, the D-optimal designs have a better D-efficiency than the OAs. The largest discrepancy is the one stated above for OA$(40, 8, 2)$. The compromise designs generally have the smallest D-efficiency of the three types of design. The most substantial discrepancy with respect to the D-optimal design again occurs for the case of 40 runs and 8 factors; the difference in efficiencies is 0.1054.

There are very substantial differences in D_s-efficiencies between the compromise designs and the D-optimal designs or the OAs. The difference between OAs and compromise designs can be as large as 0.3323, for 40 runs and 8 factors. The discrepancy between the compromise designs and the D-optimal designs is generally smaller, but can nevertheless be substantial.

The observations on the various efficiencies suggest that in general D-optimal designs should be preferred if the assumption of effect hierarchy is not likely to hold, while the compromise designs should be preferred otherwise. In general, orthogonal arrays of strength 2 do not seem to be particularly favorable to estimate all the interactions or to estimate main effects independently from interactions. One notable seven-factor exception will be discussed in Section 5.1.

4.2 Strength-3 arrays and alternative designs

Table 6 shows results for the strength-3 OAs and alternative designs. The table has the same layout as the one for the strength-2 OAs plus alternatives. For these alternatives, we used our

Table 3: Strength-3 arrays and alternative designs

N	n	Type	D	B	D_s	N	n	Type	D	B	D_s
40	7	OA	0.8030	0.8030	1	56	9	OA	0.7610	0.7610	1
		D-optimal	0.9534		0.9343			D-optimal	0.8723		0.7746
		compromise	0.8875		0.9884			compromise	0.8067		0.9256
40	8	OA	0	0	1	64	8	OA	1	1	1
		D-optimal	0.8517		0.6967			D-optimal	1		1
		compromise	0.7463		0.9734			compromise	0.9780		1
48	7	OA	0.9585	0.9585	1	64	9	OA	0.9254	0.9626	1
		D-optimal	0.9646		0.9500			D-optimal	0.9190		0.8097
		compromise	0.9585		1			compromise	0.8831		0.9782
48	8	OA	0.8365	0.8365	1	64	10	OA	0.8247	0.9692	1
		D-optimal	0.9053		0.8222			D-optimal	0.8371		0.7074
		compromise	0.8450		0.9859			compromise	0.7604		0.8864
48	9	OA	0.6753	0.6753	1	72	8	OA	0.9283	0.9439	1
		D-optimal	0.7951		0.5875			D-optimal	0.9824		0.9759
		compromise	0.7250		0.8759			compromise	0.9730		0.9926
56	7	OA	0.9192	0.9192	1	72	9	OA	0.8818	0.9391	1
		D-optimal	0.9757		0.9626			D-optimal	0.9473		0.8844
		compromise	0.9585		0.9912			compromise	0.9117		0.9655
56	8	OA	0.8642	0.8642	1	72	10	OA	0	0.9369	1
		D-optimal	0.9547		0.9114			D-optimal	0.8935		0.7752
		compromise	0.9040		0.9903			compromise	0.8180		0.9330

optimal design software with 5,000 initial tries and we kept the best design either according to D-efficiency or to the compound criterion $D + 2D_s$.

For the 40-run and 48-run OAs we used a complete enumeration. For the 56-run arrays, we completely generated the series with eight-factor arrays. Nine-factor arrays were only generated by extension of the 10,491 eight-factor arrays that support an interaction model. The extension resulted in only 28 nine-factor arrays that permit fitting an interaction model in this number of factors. Further extension resulted in a single ten-factor array. However, it is not possible to fit an interaction model based on this array. For the 64-run arrays, we completely generated the series with six factors. We extended all 326 arrays that support an interaction model. From seven factors onward, we retained at most a few thousands of arrays that support an interaction model. For the 72-run arrays, we again started with generating all six-factor arrays. We extended all 872 arrays that support an interaction model. Extension of these arrays resulted in more than two million seven-factor arrays. From seven factors onward, we retained only a part of the arrays that support an interaction model. We obtained five nine-factor arrays and we failed to obtain a ten-factor array. Details on the numbers of generated OAs and cutoffs used can be found in Appendix E.

All OAs generated have a D_1-efficiency equalling one. The designs generated with the optimal design software were all nearly orthogonal; all D_1-efficiencies were larger than 0.95. For this reason, we did not include such D-efficiencies in Table 6.

The seven-factor OA in 48 runs is markedly better than the 40-run OA used in the motivating example. We prefer this OA to the D-optimal design of the same run size because the OA's D_s-efficiency is better, while its D-efficiency is only slightly less. The compromise design happens to be the same OA.

The OAs of 64 runs and 72 runs shown in Table 6 were partially generated. The table shows two 64-run designs for 8 factors with a D-efficiency of 1. So these are strength-4 OAs.

The compromise design is a strength-3 OA with a D-efficiency near 1, so that it has almost a strength of 4.

For the nine-factor OAs in 64 runs, the best D-efficiency found is 0.0372 lower than the upper bound, while the discrepancy is 0.1445 for the ten-factor arrays. However, the D-efficiencies obtained are very near those of the D-optimal designs; one is slightly worse and one is even slightly better. In addition, as shown in Section 5, the nine-factor and ten-factor arrays we did obtain are competitive with the best literature arrays. For these reasons, we did not intensify the search of good ten-factor OAs.

For the 72-run OAs, the discrepancy between the best D-efficiency and the upper bound for eight and nine factor arrays is 0.0156 and 0.0573, respectively. Our failure to find a 72-run ten-factor array did not prompt us to extend more than the few hundred seven-factor or eight-factor arrays with this run size, because even a few arrays extra lead to millions of new extensions. We believe that 72 is the maximum run size for which our methodology can yield useful OAs. Note that the compromise design of 10 factors has a good D_s-efficiency, while the D-optimal 10-factor design has a good D-efficiency.

A further comparison among the designs in Table 6 leads to the following conclusions.

- In general, the D-optimal designs have a substantially better D-efficiency than OAs of the same run size and number of factors. Exceptions are the 64-run cases of 9 and 10 factors and the OA$(48, 7, 3)$.
- D-optimal designs generally have substantially worse D_s-efficiencies than the compromise designs.

5 Study of specific cases

In this section, we study the standard errors in the interaction models based on the most efficient designs we found for 7–10 factors. We compare our OAs with the smallest orthogonal arrays from the literature. Many of the new OAs are smaller, or have a higher D-efficiency, or have smaller standard errors for the coefficients, than the literature OAs. The OAs are contrasted with D-optimal designs and compromise designs. In general, we prefer designs for which the maximum standard error, either of the main effects or of the interactions, is minimized over the competing designs of the same run size. All the designs discussed are available from the authors.

5.1 Seven factors

The two earliest literature OAs that we were able to find for the interaction model in seven factors have, 48 runs, a strength of 2, a D-efficiency of 0.9222 and D_s-efficiencies of 0.8187 and 0.8823, respectively; see Mee (2009, p. 291) and Addelman (1961) for their construction. Schoen and Mee (2012) recommended seven-factor arrays of strength 3 in 40 and 48 runs capable of fitting the interaction model. These correspond to the seven-factor OAs of strength 3 and 40 or 48 runs from Table 6. The 48-run OA from this table has a D-efficiency of 0.9585 and a maximum D_s-efficiency, while the best seven-factor arrays of strength 2 and 40 or 44 runs from Table 5 have D-efficiencies of 0.9245 and 0.9449, and D_s-efficiencies of 0.8495 and 0.8926, respectively. So these alternatives have a better D-efficiency and either a greater strength or a smaller run size than the early literature arrays.

In the remainder of this section, we restrict attention to 32-run and 36-run designs. While their D-efficiencies are smaller than those of the earlier literature designs, their run size is also smaller. The 32-run OAs are the smallest possible orthogonal arrays that support the interaction model for seven factors. The 36-run designs allow 7 residual degrees of freedom to conduct t tests, assuming that higher-order effects are negligible.

Five OA$(32, 7, 2)$ have the globally best D-efficiency for OAs of this size, which equals 0.8432. The D-efficiency for the best 36-run design we found is considerably higher; its value is 0.9022. Assuming an error variance of 1, we calculated standard errors of the coefficients in the interaction

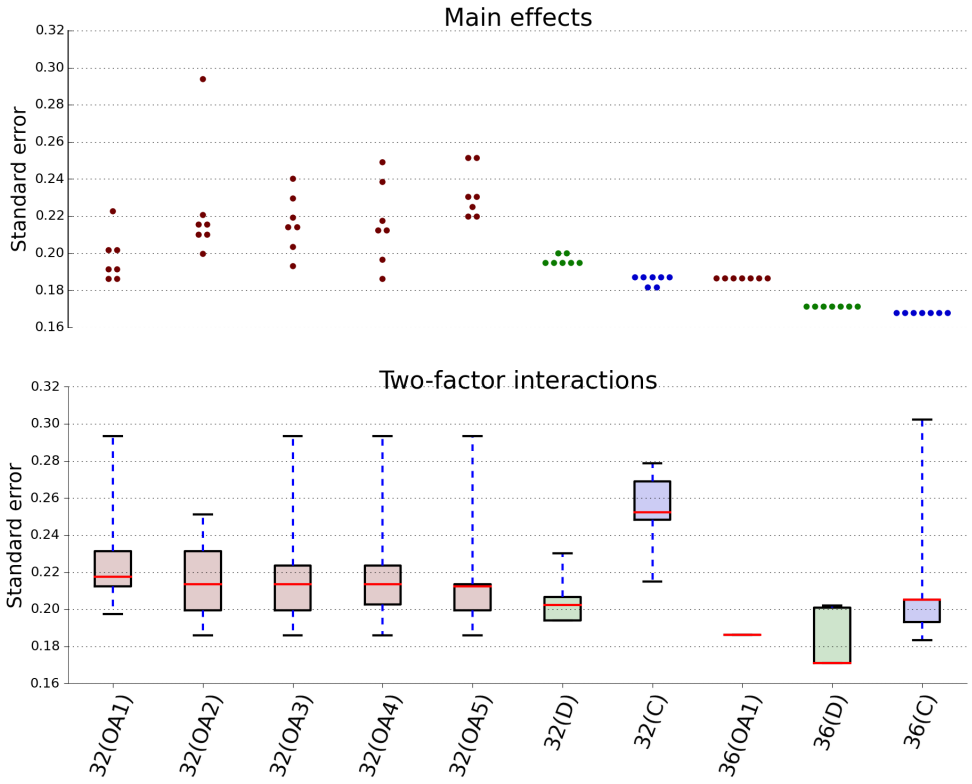


Figure 2: Standard errors for the coefficients of interaction models based on seven-factor designs of 32 or 36 runs. OA: orthogonal array of strength 2; D: D-optimal design; C: compromise design.

model for the five best 32-run OAs, the 36 run OA, the D-optimal designs and the compromise designs of the same run sizes. Figure 2 shows the results. The upper panel in the figure is a dotplot of the standard error of the main effects, while the lower panel shows boxplots for the interactions. The results for OAs are in brown-red, those for the D-optimal designs in green and those for the compromise designs in purple.

For the 32-run designs, the best design under effect hierarchy is the compromise design, because it minimizes the maximum standard errors for the main effects. In fact, these standard errors are smaller than most of those of the OA and D-optimal alternatives. In case many interactions are likely to be substantial, we recommend the D-optimal design. This design minimizes the maximum standard error for interactions when compared with the alternative designs. There is no clear reason to recommend a 32-run OA, as judged by the standard errors both of the the main effects and of the interactions.

The most remarkable feature of Figure 2 is the complete uniformity of standard errors based on the 36-run design. Indeed, all standard errors equal 0.1860. As this value also minimizes the maximum standard error of the interactions, we would favor the OA in case many interactions are expected. However, under effect hierarchy, we would still favor the compromise design, which has slightly smaller standard errors for the main effects.

5.2 Eight factors

Addelman (1961) and John (1962) present a design for eight factors in 48 runs which turns out to be an orthogonal array of strength 2; see also Mee (2009). This design has a D-efficiency of 0.8927. We offer alternative strength-2 designs with a smaller run size, albeit also with a

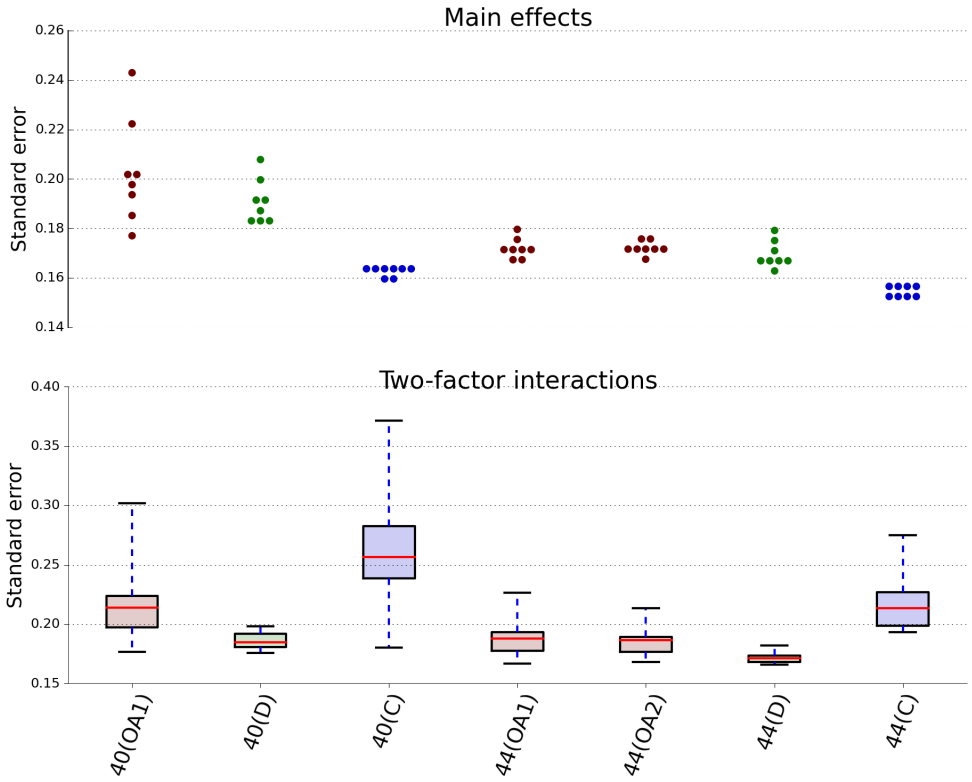


Figure 3: Standard errors for the coefficients of the interaction model based on eight-factor designs of 40 or 44 runs. OA: orthogonal array of strength 2; D: D-optimal design; C: compromise design.

smaller efficiency. The maximum D-efficiency for our set of 40-run designs is 0.8019; remaining efficiencies are below 0.80. The best two designs in our 44-run series have efficiencies of 0.8511 and 0.8524, respectively; remaining efficiencies are below 0.85.

We show the standard errors for the factorial effects in the most D-efficient of the 40-run OAs and the two most D-efficient 44-run OAs in Figure 3 along with those of D-optimal designs and compromise designs. Note that design 44(OA1) is slightly less D-efficient than 44(OA2).

For the eight-factor cases shown here the compromise designs have smaller standard errors for the main effects than the OAs and the D-optimal alternatives. Therefore, the compromise designs are recommended under effect hierarchy. The D-optimal designs are recommended in case many interactions could be active, because the standard errors for the two-factor interactions are small and homogeneous. OAs of strength 2 are not recommended for the eight-factor cases studied here.

5.3 Nine factors

The smallest orthogonal designs to estimate the interaction model for nine factors have 48 runs. Schoen and Mee (2012) give a 48-run design of strength 3 that permits estimation of the interaction model in nine factors. However, its D-efficiency is only 0.6753, while the variance inflation factors for the interaction coefficients range between 1.6 and 33, with a median value of 3. In view of these properties, we do not recommend the 48-run design to estimate a full interaction model.

There are 28 56-run designs of strength 3 to estimate the interaction model. The best of these

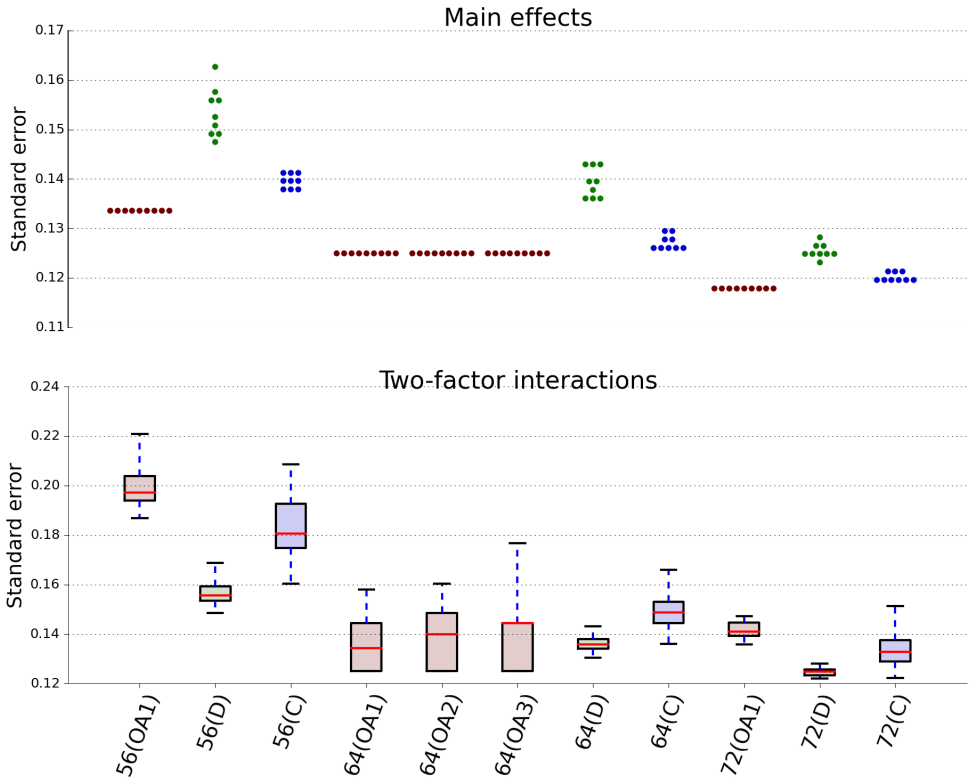


Figure 4: Standard errors for the interaction coefficients in nine-factor designs of 56, 64 or 72 runs. OA: orthogonal array of strength 2; D: D-optimal design; C: compromise design.

designs has a D-efficiency of 0.7610. At the cost of eight additional runs, Mee (2004) proposes a far more efficient design, which has a D-efficiency of 0.9230. We found two designs with a slightly larger D-efficiency than the literature design, viz., 0.9254 and 0.9231, respectively.

In addition to the extra 64-run designs, we found five 72-run designs, whose efficiencies range from 0.8750 to 0.8818.

Figure 4 shows the standard errors of the interaction coefficients in the interaction models based on selected 56-run, 64-run and 72-run OAs, along with those of D-optimal designs and compromise designs. The OAs shown are the most D-efficient 56-run OA, the OAs with best, second-best and third best D-efficiencies found for 64 runs and the most D-efficient OA found for 72 runs. The standard errors for the main effects in the OAs of 56, 64 and 72 runs are 0.134, 0.125 and 0.118, respectively.

The three 64-run OAs represented in Figure 4 are in decreasing order of D-efficiency. The design recommended by Mee (2004) is 64(OA3). We prefer OA1 to OA2 and OA3 because it has the smallest maximum standard error of the interactions. However, the D-optimal design outperforms the OAs when the standard errors for interactions is considered.

The most D-efficient OA we obtained has a higher D-efficiency than the D-optimal design obtained with our software after 5,000 tries. The efficiencies were 0.9254 for the OA and 0.9190 for the alternative design. The figure shows that both designs are fundamentally different, however. The high efficiency of the OA is spent on the strength-3 property, while the efficiency of the alternative design is spent on the small standard errors for the interactions. This suggests that it pays of to collect several designs with high D-efficiency, and study further properties of these designs.

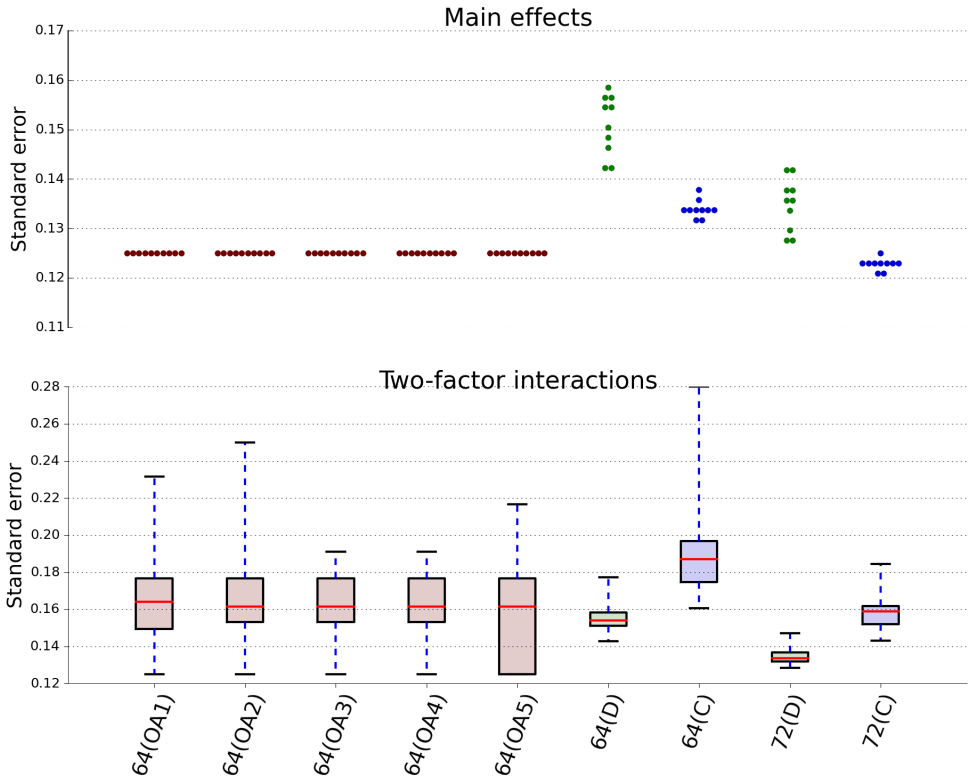


Figure 5: Standard errors in ten-factor designs of 64 and 72 runs. OA: orthogonal array of strength 3; D: D-optimal design; C: compromise design. D-efficiencies of OAs are 0.8247 (OA1) and 0.8238 (remaining OAs).

The comparison among the nine-factor designs confirms that strength-3 OAs are good candidates under effect hierarchy, while D-optimal designs are preferred if many interactions are expected. Compromise designs are better than D-optimal designs for main-effect estimation and worse than D-optimal designs for the estimation of interactions. A convenient feature of compromise designs and D-optimal designs is that they can be constructed for all run sizes that permit estimation of the interaction model.

5.4 Ten factors

A ten-factor design that supports the interaction model must have at least 56 runs. Our work shows that there are no such strength-3 OAs. Our software returned D-optimal and compromise designs with D-efficiencies of 0.7492 and 0.6373 and D_s-efficiencies of 0.5275 and 0.8070, respectively. We do not recommend these designs. We obtained 60-run D-optimal and compromise designs with D-efficiencies of 0.7978 and 0.7100 and D_s-efficiencies of 0.6309 and 0.8627, respectively. However, it is usually prudent to include a few extra runs for model checking or estimation of random error. So it is natural to consider 64-run designs.

The 64-run strength-3 OA given by Mee (2004), designated 64 (OA5), is the smallest and most D-efficient literature orthogonal design in 10 factors. Its D-efficiency equals 0.8238. For reasons explained below, our enumeration did not include this design. However, we found three other designs with the same D-efficiency and one design with a slightly higher D-efficiency of 0.8247.

Figure 5 shows the standard errors of based on the five 64-run OAs, the D-optimal and

compromise alternative designs, and 72 run D-optimal and compromise designs. A good 72-run OA was not obtained; see Section 4.2.

The smallest standard errors for the main effects are reached for the compromise 72-run design, while the smallest standard errors for the interactions are reached for the 72-run D-optimal design. If budget allows, we prefer these designs to the 64-run alternatives.

For the 64-run cases, it is obvious that all OAs are equally good regarding the main effect standard errors, while the compromise designs do better than the D-optimal designs and worse than the OAs. Under effect hierarchy, we prefer one of the OAs. We prefer OA3 or OA4, because the maximum standard error for the interactions is minimal in these OAs. OA3 and OA4 are nonisomorphic, but as it turns out, they have the same frequency distribution of standard errors for the interactions.

The first seven columns of the lexicographically minimal version of design 64 (OA5) form a seven-factor design with a D-efficiency of 0.9310. In our enumeration, we did not extend 64-run seven-factor OAs with a D-efficiency of 0.9410 or less; see Appendix E. This explains why design 64 (OA5) was not included in our list of 10-factor designs. The finding illustrates that extension of two k-factor OAs A_1 and A_2, where $\mathrm{D}(A_1) < \mathrm{D}(A_2)$ can lead to extended designs A_1^+ and A_2^+ for which the order of D-efficiencies is reversed.

Finally the set of standard errors for the interactions based on the D-optimal 64-run design is much more homogeneous and also has a smaller maximum than the corresponding sets for the OAs. As the D-efficiency of the D-optimal design is very similar to the best efficiency of a strength-3 OA, the standard errors of the main effects based on the D-optimal designs are considerably higher. Our conclusion is again that the optimal design is preferred if effect hierarchy is not likely to hold, while the OAs are preferred under effect hierarchy.

6 Discussion

In this paper, we studied two-level experiments large enough to estimate a model with all the main effects and all the two-factor interactions, while yet the effect hierarchy assumption suggests that main effects should be given more prominence than two-factor interactions. We considered approaches based on orthogonal arrays and optimal designs.

Strength-3 OAs have maximum main effect precision irrespective of the number of interactions in the model. Therefore, the most D-efficient OA of strength 3 is an attractive option under effect hierarchy. If this assumption is not likely to hold, we would generally recommend D-optimal designs, because these have better precisions for the interactions.

Strength-2 OAs have maximum precision of the main effects only if no interactions are active. D-optimal designs for the interaction model were nearly orthogonal for the main-effect only model; for run sizes $N \leq 44$, the D_1-efficiencies were all larger than 0.98; for larger run sizes, they were all larger than 0.95. In addition, these designs have a better precision for the interaction coefficients. For these reasons, we generally do not recommend a strength-2 OA to fit the interaction model, with one notable exception: we found an $\mathrm{OA}(36, 7, 2)$ for which all standard errors in the interaction model are equal. The D-optimal design we found has several standard errors for interactions that are higher than the standard error in the OA. Therefore, we recommend the OA for this case.

To attain a better D_s-efficiency when using an optimal design approach, we implemented a coordinate exchange procedure that optimizes $\alpha_1 \mathrm{D} + \alpha_2 \mathrm{D}_s$. The results presented here were obtained with the settings $\alpha_1 = 1$ and $\alpha_2 = 2$. The designs are a compromise between D-optimal designs and strength-3 OAs, both in terms of D-efficiency (D-optimal designs are generally better and strength-3 OAs worse) and D_s-efficiency (D-optimal designs are substantially worse and strength-3 OAs are better).

As the D_s-efficiency of the compromise designs is generally better than the D_s-efficiency of strength-2 OAs, the compromise designs provide an attractive alternative to these OAs under effect hierarchy. Further, as the run size increases, it becomes increasingly difficult to obtain good strength-3 OAs, and compromise designs could be used instead. The compromise designs have

the general advantage that they can be constructed for every run size that is compatible with fitting the interaction model. Interesting subjects for further research include the adaptation of this optimal design approach to fitting other models than the complete interaction model and optimization of this approach for larger cases than we studied here.

One problem in the generation of D-efficient OAs is the huge amount of different designs. The largest strength-2 case that we could handle completely is $\mathrm{OA}(32, 7, 2)$ with 530,469,996 different designs, while the largest strength-3 case is $\mathrm{OA}(48, 9, 3)$ with 166,081 different designs. This was the reason to develop a partial enumeration approach. We established upper bounds for the best possible D-efficiency of arrays that were not generated. Our approach resulted in smaller or more efficient alternatives to literature designs. We believe, however, that we reached the limits of its usefulness for the OAs with 10 factors and 64 or 72 runs.

A second interesting subject for further research is to explore multilevel designs either with our partial enumeration approach for orthogonal designs or with the compromise optimal design approach. For example, for four three-level factors, our methodology might yield a suitable 36-run design. For five factors the nearest run size for an orthogonal design is 54. Sartono et al. (2012) showed that the four strength-3 designs of this size do not support the interaction model. Our methodology might give strength-2 designs or compromise designs that are capable of estimating this model. For six factors, there are strength-3 designs in 81 runs that support the interaction model (Sartono et al., 2012).

Acknowledgements

The research of the second author was supported by the Flemish Fund for Scientific Research FWO.

A Proofs of the theorems

In this section, we prove three theorems that state upper bounds on the D-efficiencies of orthogonal arrays obtained by extending an orthogonal array A with $k < n$ factors. One of the upper bounds is based on the general decrease of the determinant of an interaction model when an extra factor is added to the original array. The other bounds are based on the determinant of an interaction model when two or several extra factors are added, and on all results for the intermediate arrays with only one extra factor. We call the upper bounds one-step bound, two-step bound, and multi-step bound, respectively. The main paper gives the one-step bound and the multi-step bound.

As in the main paper, we work with a normalized version X of the interaction model matrix $\mathbf{X}$ of an array A. The columns in X are normalized to a length of 1. It is readily seen that $|X^T X| = |\mathbf{X}^T\mathbf{X}/N| = [\mathrm{D}(A)]^p$, with $p = 1 + k + k(k-1)/2$. Despite the fact that the columns of X are normalized, we call $[\mathrm{D}(A)]^p$ the unnormalized D-efficiency of array A, because this efficiency has not been normalized by the number of parameters p in the interaction model. We use the symbol $\mathcal{D}(A)$ for the unnormalized D-efficiency.

A.1 One-step bound

The one-step bound exploits the fact that determinants of the type $|X^T X|$ generally decrease when X is extended by extra normalized columns. The main result is the following theorem.

Theorem 3. *Let A be an orthogonal array with N rows and k columns that can fit the interaction model. Let $P = [A\,E]$ be an array that results from extending A with a single column E. Let $p_k = 1 + k + k(k-1)/2$. Then $\mathrm{D}(P) \leq \mathrm{D}(A)^{p_k/p_{k+1}}$.*

Proof. Consider the extension of X, which is the normalized interaction model matrix based on A, with a single column x, which is either the normalized main effect contrast vector of E or the

element wise product of this contrast vector with one of the main effect contrast vectors in X. Denote the resulting matrix with $X_+ = [Xx]$. Then

$$X_+^T X_+ = \begin{pmatrix} X^T X & X^T x \\ x^T X & x^T x \end{pmatrix}.$$

The columns of X and the column vector x are normalized. Therefore, the diagonal elements of $X^T X$ are all equal to 1, $|X^T X| \leq 1$ and $|X_+^T X_+| \leq 1$. Using a well-known result on the determinant of a partitioned matrix (Schur, 1917),

$$\begin{aligned} |X_+^T X_+| &= |X^T X| \times |x^T x - r^T (X^T X)^{-1} r| \\ &= |X^T X| \times |1 - r^T (X^T X)^{-1} r| \end{aligned}$$

where $r = X^T x$. The matrix $(X^T X)^{-1}$ is positive definite. Therefore $|X_+^T X_+| \leq |X^T X| \leq 1$. The theorem follows by extending X with the main effect contrast vector of E and all the two-factor interaction contrast vectors involving the new factor E, and taking the appropriate power of the determinants. □

A.2 Multi-step bound

The multi-step bound predicts what might happen if an array is extended with several extra columns.

Theorem 4. *Let A be an orthogonal array in LM form. Let $P_i = [A\, E_i]$ be an LM orthogonal array that results from extending A with a single column E_i. Let Q_k be an array that result from extending P_i with q columns and let $L_i = \mathcal{D}(P_i)/\mathcal{D}(A)$. Then*

$$\mathcal{D}(Q_k) \leq L^q \mathcal{D}(P_i) \tag{4}$$

with $L = \max_j L_j$. The maximum is over all possible extensions E_j of A that lead to an LM array.

In the next section, we proof a simpler version of this theorem, showing what might happen if an array is extended with two extra columns. We do not provide a proof of Theorem 4 as it generalizes the simpler version to cover multiple extra columns.

A.3 Two-step bound

The two-step bound relates the unnormalized D-efficiency of an array A after extension with two factors to the minimal decrease when A is extended with a single factor. The theorem stating this bound builds on four lemmas on the relationship between the squared volume of a matrix M, denoted with $\mathcal{V}(M)$, and defined as $\mathcal{V}(M) = \det(M^T M)$, and the squared volume of extended versions of M. As before, we denote the normalized interaction model matrix based on A with X. So $\mathcal{D}(A) = \mathcal{V}(X)$.

A.3.1 Four lemmas

First, consider the extension of A with a single factor, which we denote as E. The interaction model matrix based of the new array corresponds to the matrix X extended with $k+1$ columns. These columns are the main effect contrast vector of the extra factor E and the k contrast vectors modelling the interaction between this factor and the factors in A. We denote the extra columns of X by the $N \times (k+1)$ matrix Y. We establish the following Lemma.

Lemma 1. *Let X and Y be matrices with N rows. Denote by $X^\perp$ the orthogonal complement to the space spanned by the columns of X. Denote by $Y^*(X)$ the matrix formed by orthogonal projection of the columns of Y on $X^\perp$. Then*

$$\mathcal{V}([XY]) = \mathcal{V}(X)\, \mathcal{V}(Y^*(X)).$$

By this lemma, $\mathcal{D}([A\,E])$ is the product of $\mathcal{D}(A)$ and the volume of the parts of the columns of Y which are orthogonal to the columns of X.

Proof. Using a well-known result on the determinant of a partitioned matrix (Schur, 1917), it is easy to show that

$$\mathcal{V}([XY]) = \mathcal{V}(X) \times \det[Y^TY - Y^TX(X^TX)^{-1}X^TY].$$

We rewrite the determinant as follows

$$\begin{aligned} \det[Y^TY - Y^TX(X^TX)^{-1}X^TY] &= \det[Y^T(I - X(X^TX)^{-1}X^T)Y] \\ &= \det[Y^T(I - X(X^TX)^{-1}X^T)(I - X(X^TX)^{-1}X^T)Y] \\ &= \mathcal{V}([Y - X(X^TX)^{-1}X^TY]). \end{aligned} \tag{5}$$

The columns of the matrix in the last line of this equation are differences between the columns of Y and the projection of these columns on the space spanned by the columns of X. This difference therefore is the projection of the columns of Y on $X^\perp$. □

Second, consider a set of column vectors other than Y, collected in the matrix K. The new set of vectors can be thought of as arising from the extension of the interaction model matrix based on $[A\,E]$ with columns from a newly added factor. Denoting the projection of Y on $X^\perp$ with Z, we link the squared volume of the projection of K on $[X\,Z]^\perp$ to the squared volume if K were projected on X alone by Lemma 2.

Lemma 2. *Let K, X and Z be matrices with N rows, with $Z^TX = 0$. Then*

$$\mathcal{V}(K^*([XZ])) \leq \mathcal{V}(K^*(X)).$$

We first prove a simplified version of this Lemma, as stated below.

Lemma 3. *Let Y be an $N \times k$ matrix and let X be an $N \times a$ matrix. Denote by $X^\perp$ the orthogonal complement to the space spanned by the columns of X. Denote by $Y^*(X)$ the matrix formed by orthogonal projection of the columns of Y on $X^\perp$. Then*

$$\mathcal{V}(Y^*(X)) \leq \mathcal{V}(Y).$$

Proof. Without loss of generality we assume both X and Y have full column rank. We factorize the matrix Y using QR decomposition as $Y = QR$ and write $\lambda = \det(R)$. Then

$$\begin{aligned} \mathcal{V}(Y) &= \det Y^TY = \det R^TQ^TQR \\ &= \det R^TR = \lambda^2. \end{aligned}$$

Next, we denote the orthogonal projection operator that projects a vector on $X^\perp$ with P. So $Y^*(X) = PY$. We have

$$\begin{aligned} \mathcal{V}(Y^*(X)) &= \det\left((PY)^TPY\right) = \det R^TQ^TP^TPQR \\ &= \lambda^2 \det(PQ)^TPQ. \end{aligned}$$

We denote the columns of Q by Q_i. The diagonal elements of the matrix $G = (PQ)^TPQ$ are equal to $g_{ii} = \langle PQ_i, PQ_i\rangle$, where $\langle .,.\rangle$ denotes the inner product of two vectors. We have $g_{ii} = \langle PQ_i, PQ_i\rangle \leq \langle Q_i, Q_i\rangle = 1$. This implies $\mathrm{tr}(G) \leq N$. The matrix G is definite nonnegative. Therefore, its eigenvalues are nonnegative. Therefore, $\det G \leq 1$. We conclude that $\mathcal{V}(Y^*(X)) = \lambda^2 \det G \leq \lambda^2$. □

Lemma 2 can now be proven as follows.

Proof. We assume without loss of generality that Y, X and $[XZ]$ are non-singular. We write P for the projection onto the orthogonal complement of X and write S for projection onto the orthogonal complement of Z. We decompose the matrix PY using QR-decomposition as $PY = QR$ with $\lambda = \det(R)$. Then

$$\mathcal{V}(Y^*(X)) = \det(PY)^T(PY) = \det R^T R = \lambda^2.$$

For the projection of Y on $[XZ]^\perp$ we first project on $X^\perp$ and we project the result on $Z^\perp$: $Y^*([XZ]) = [Y^*(X)]^*(Z) = [PY]^*(Z) = SPY$. Noting that $Z^T X = 0$, it is easy to show that the same result is obtained by first projecting on $Z^\perp$ and then on $X^\perp$. Further,

$$\mathcal{V}(Y^*([XZ])) = \det(SPY)^T(SPY) = \lambda^2 \det(SQ)^T(SQ).$$

Proceeding as in Lemma 3 completes the proof. □

Finally, consider the extension of A with two extra columns, which we denote as E_1 and E_2. We compare the unnormalized D-efficiency of the twice extended array, $\mathcal{D}([AE_1E_2])$ with the unnormalized D-efficiencies $\mathcal{D}([AE_1])$ and $\mathcal{D}([AE_2])$ using the following Lemma.

Lemma 4. *Let E_1 and E_2 be two different columns with which an $N \times k$ two-level array A can be extended to a two-level array with one more column. Then*

$$\mathcal{D}(A)\mathcal{D}([AE_1E_2]) \leq \mathcal{D}([AE_1])\mathcal{D}([AE_2]). \tag{6}$$

Proof. Let X be the interaction model matrix of A. So the dimension of X is $N \times (1+k(k+1)/2)$. Let Y_1 be the $N \times (k+1)$ extension of X due to the extension of A to $[AE_1]$. Let Y_{2+} be the $N \times (k+2)$ extension of $[XY_1]$ due to the extension of $[AE_1]$ to $[AE_1E_2]$. Finally, let Y_2 be the $N \times (k+1)$ extension of X due to the extension of A to $[AE_2]$. So the single column that is in $Y_{2+} - Y_2$ is the element wise product of E_1 and E_2. We have

$$\begin{aligned}
\mathcal{D}(A)\mathcal{D}([AE_1E_2]) &= \mathcal{V}(X)\,\mathcal{V}([XY_1Y_{2+}]) \\
&= \mathcal{V}(X)\,\mathcal{V}([XY_1])\,\mathcal{V}(Y_{2+}^*([XY_1])) \\
&\leq \mathcal{V}(X)\,\mathcal{V}([XY_1])\,\mathcal{V}(Y_{2+}^*(X)) = \mathcal{V}([XY_1])\,\mathcal{V}([XY_{2+}]) \\
&\leq \mathcal{V}([XY_1])\,\mathcal{V}([XY_2]) = \mathcal{D}([AE_1])\mathcal{D}([AE_2]).
\end{aligned} \tag{7}$$

The second and third equalities follow from Lemma 1. The first inequality follows from Lemma 2. The second inequality follows from Lemma 1 and taking $Y = Y_{2+} - Y_2$. □

A.3.2 Theorem

Building on Lemma 4, we establish an upper bound on the D-efficiency of an array with $k+2$ factors constructed by extending the k-factor array A with two columns. This upper bound requires knowledge of all $\mathcal{D}([A\,E_i])$, where the set $\{E_i\}$ is the set of all possible columns with which A can be extended to an LM orthogonal array with $k+1$ columns. Let $P_i = [A\,E_i]$. We define the loss factor L_i of P_i as $L_i = \mathcal{D}(P_i)/\mathcal{D}(A)$. The minimum value for L_i is 0; this value is reached if the interaction matrix based on P_i is singular. The maximum value is 1; this value is reached if E_i and the element wise products of E_i with the columns of A are orthogonal to the interaction model matrix based on A.

We state the upper bound in the following theorem.

Theorem 5. *Let A be an orthogonal array of N rows and k columns in LM form. Let $P_i = [A\,E_i]$ be an LM orthogonal array that results from extending A with a single column E_i. Let $L_i = \mathcal{D}(P_i)/\mathcal{D}(A)$. Let Q be an LM array that results from extending P_i with a single column F, and let $L_F = \mathcal{D}([A\,F])/\mathcal{D}(A)$. Then*

$$\mathcal{D}(Q) \leq L\mathcal{D}(P_i) \tag{8}$$

with $L = \max_j L_j$. The maximum is over all possible extensions E_j of A that lead to an LM array.

Proof. The model matrix of Q is of the form $[XY_iY_{j+}]$. Note that $\mathcal{D}(Q) = \mathcal{V}(M)$, with M the normalized interaction model matrix of Q. Then, by Lemma 4,

$$\begin{aligned}\mathcal{V}(M)\,\mathcal{V}(X) \le \mathcal{V}([X\,Y_i])\,\mathcal{V}([X\,Y_j]) &= L_F\,\mathcal{V}(X)\,\mathcal{V}([X\,Y_i]) \\ &= L_F\,\mathcal{V}(X)\mathcal{D}(P_i),\end{aligned}$$

Hence $\mathcal{V}(M) \le L_F\mathcal{D}(P_i)$. The fact that extension of P_i with column F leads to an LM array does not imply that $[AF]$ is an LM array. Therefore, F might not be among the set $\{E_i\}$. However, since $[P_iF]$ is LM, there is an index j such that $[AF]$ is isomorphic to $[AL_j]$. Hence $L_F = L_j$ for some index j. Since $L_F = L_j \le L$, the theorem is proved. □

B Application of the theorems

B.1 One-step bound

To illustrate the application of Theorem 3, consider the generation of OA$(40, 7, 2)$, which is summarized in Table 4. The generation starts with a minimum complete set of OA$(40, 4, 2)$. There are 32 nonisomorphic arrays in the set. We extend the 30 arrays that permit estimation of the interaction model and find 3,910 arrays in OA$(40, 5, 2)$. The best D-efficiency for the interaction model based on an array in the set of five-factor arrays equals 0.9781. Extension of the remaining two four-factor arrays could only result in arrays with zero D-efficiencies. Therefore, the most D-efficient array in the set with five-factor arrays has the best possible D-efficiency based on any OA$(40, 5, 2)$.

Continuing our illustration, we extend the 600 most D-efficient arrays in the set of OA$(40, 5, 2)$ to OA$(40, 6, 2)$. The extension generates 6,954,211 arrays. The best array according to the D-efficiency criterion has $D = 0.9657$. The highest D-efficiency among 3,310 arrays that have not been extended equals 0.8899. The best D-efficiency that might be achieved when this array would be extended, designated $O_{5,6}$ equals $0.8899^{16/22} = 0.9186$. This shows that the maximum D-efficiency of any OA$(40, 6, 2)$ equals 0.9657.

Finally, we extend the 300 most D-efficient OA$(40, 6, 2)$ to OA$(40, 7, 2)$. This extension results in 2,889,203 arrays. The highest D-efficiency is 0.9245. The best D-efficiency that might be attained when the best of the discarded five-factor arrays would have been extended, $O_{5,7}$, equals 0.9376. The efficiency bound based on discarded six-factor arrays, $O_{6,7} = 0.9506$. We conclude that the best possible D-efficiency based on any OA$(40, 7, 2)$, $D_{\max}$, is $0.9245 \le D_{\max} \le 0.9506$.

Table 4: Application of Theorem 3 to OA$(40, n, 2)$.

n	Not extended		Extended		Upper bound
	Number	max(D)	Number	max(D)	
4	2	0	30	0.9889	
5	3,310	0.8899	600	0.9781	
6	6,953,911	0.9355	300	0.9657	$O_{5,6} = 0.9186$
7	2,889,203	0.9245	-	-	$O_{5,7} = 0.9376$, $O_{6,7} = 0.9506$

B.2 Multi-step bound

To illustrate the application of Theorem 5, we continue the example started in Section B.1. Extension of the 32 four-factor arrays results in 32 groups of arrays, with 3,910 OA$(40, 5, 2)$ in total. The D-efficiencies for 15 of these groups are shown in Figure 6 as small bullets. To enhance the legibility of the figure, the other groups are not shown. Efficiencies based on arrays with a common four-factor parent have the same horizontal coordinate in the figure. A horizontal

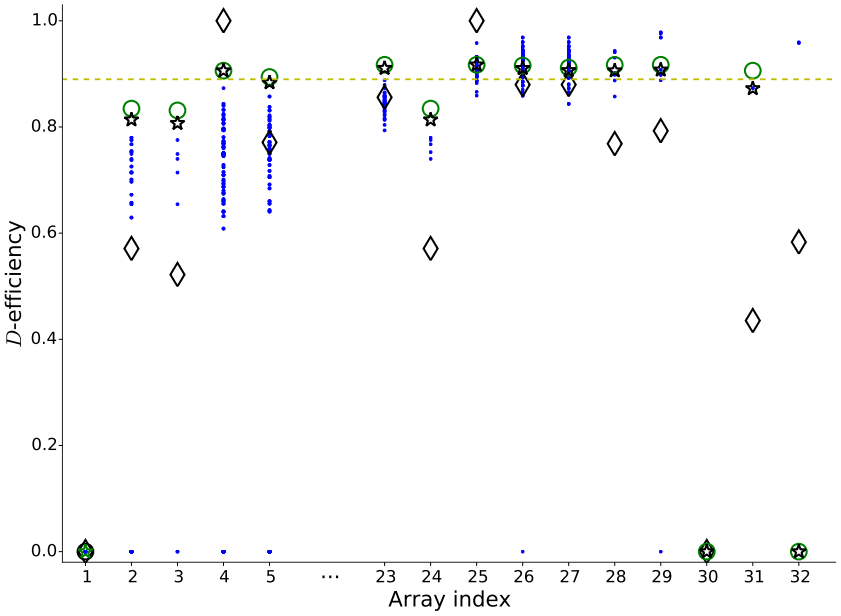


Figure 6: Two-step bounds for OA$(40, 6, 2)$ based on extensions from the 32 OA$(40, 4, 2)$. D-efficiencies of five-factor arrays (small bullets) are grouped according to the four-factor parent arrays. A dashed line separates efficiencies of 600 extended five-factor arrays from those of 3310 arrays that were not extended. Lozenges: loss factors for each group of five-factor arrays; stars: two-step bounds for each group; circles: one-step bounds for each group.

dashed line separates the efficiencies of 600 arrays that were extended from those of the 3,310 arrays that were not extended.

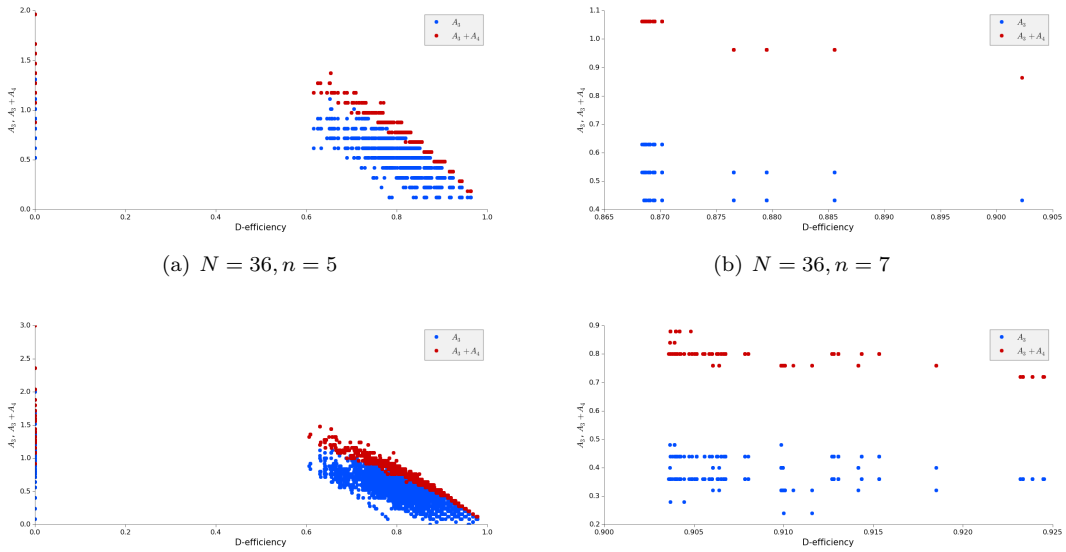
Each of the 32 groups of arrays has its own maximum loss factor L_i, which is shown as a lozenge in the group. (The two loss factors equalling 1 correspond to cases where the fifth factor is orthogonal to the main effects and interactions of the previous four factors.) We determined $t_{5,6i} = (D_{Mi}^{16} L_i)^{1/22}$, where $t_{5,6i}$ is the two-step bound for extension from five to six factors in group i, and D_{Mi} is the maximum D-efficiency for the discarded arrays in group i. The values are shown as five-pointed stars in the figure.

The two step bound over all 32 groups, $T_{5,6} = \max(t_{5,6i}) = 0.9183$. Compared with the one-step bound of 0.9186, the two step bound is sharper. However, the 600 extended five-factor arrays resulted in an OA$(40, 6, 2)$ with a D-efficiency larger than the upper bound based on the arrays that were not extended. This D-efficiency, which equals 0.9657, is therefore the globally best D-efficiency for any OA$(40, 6, 2)$.

The open circles in the figure show the values of $o_{5,6i}$. It is easy to see that $t_{5,6i} \leq o_{5,6i}$ with equality if and only if $L_i = 1$. Therefore, calculating the one-step bound makes sense only for the minimum complete set in $k < n$ factors from which we started the generation of arrays with extra factors, as for these cases we do not have the loss factors available. In the present example, the four-factor arrays that were not extended, which have index 1 and index 30 in the figure, had $D = 0$ and the one-step bounds were not needed besides the two-step bounds. However, in other cases one-step bounds were really needed to establish the upper bound.

For array 32, $t_{5,6}$ and $o_{5,6}$ are not relevant, because the two possible extensions are both retained for further extension. We indicated this situation by assigning a value of zero to both bounds.

As the arrays 1 and 30 have a singular interaction model matrix, all extensions of these arrays



(a) $N = 36, n = 5$ (b) $N = 36, n = 7$

(c) $N = 40, n = 5$ (d) $N = 40, n = 7$

Figure 7: Relation between A_3 and A_4 values and D-efficiency. Blue bullets: A_3 versus D-efficiency, red bullets: $A_3 + A_4$ versus D-efficiency

also have singular interaction model matrix, and both bounds as well as the loss factors equal zero.

Continuing our application of Theorem 5, we now consider extension of the 300 retained six-factor arrays to OA$(40, 7, 2)$. Table 4 shows that the most D-efficient seven-factor array has $D = 0.9245$. There are 3,310 discarded five-factor arrays as well as 6,953,911 discarded six-factor arrays. The five-factor arrays are divided over 32 groups according to their four-factor parent array. We determined $t_{5,7i} = (D_{Mi}^{16} L_i^2)^{1/29}$, where $t_{5,7i}$ is the two-step bound for extension from five to seven factors in group i, and D_{Mi} is the maximum D-efficiency for the discarded arrays in group i. The two step bound over all 32 groups, $T_{5,7} = \max(t_{5,7i}) = 0.9374$.

Each of the 600 five-factor arrays that were extended has its own loss factor. The 6,954,211 generated six factor arrays can be grouped according to their parent arrays. We determined $t_{6,7j} = (D_{Mj}^{22} L_j)^{1/29}$, where $t_{6,7j}$ is the two-step bound for extension from six to seven factors in group j, and D_{Mj} is the maximum D-efficiency for the discarded arrays in group j. The two step bound over all 600 groups, $T_{6,7} = \max(t_{6,7j}) = 0.9414$, which is larger than the value for $T_{5,7}$. We conclude that the best possible D-efficiency for any OA$(40, 7, 2)$, $D_{\max}$, is $0.9245 \leq D_{\max} \leq 0.9414$. Observe that the one-step bound equals 0.9506. The two-step bound is therefore sharper.

C Relation between D-efficiency and GWLP

Cheng et al. (2002) established an approximate relation between the average D-efficiency of a model containing all the main effects and g two-factor interactions and the first two elements of the generalized word length pattern (GWLP; Tang and Deng, 1999). The GWLP of an OA is a vector $(A_3, A_4, \ldots, A_n)$, where A_i is the sum of squared correlations between an i-factor interaction contrast and the intercept. In case $g = 0.5n(n-1)$, there is just one D-efficiency to consider; the relation for this case is $1/D \propto 6A_3 + 6A_4$.

In Figure 7, we study the quality of the approximation in 36-run and 40-run OAs for five and seven factors. We plotted both A_3 values and $A_3 + A_4$ values against D-efficiencies. The five-factor results are based on a complete enumeration (1242 OAs in 36 runs and 3919 OAs in

40 runs), while those for seven factors are a selection of 100 OAs in 36 runs and 200 arrays in 40 runs.

The results of 36-run OAs in five factors show that a set of OAs with one and the same A_3 or $A_3 + A_4$ value may include OAs that do not support the interaction model (D-efficiencies equal zero) as well as OAs that support this model with an efficiency of larger than 0.75. However, the best D-efficiencies occur in OAs with the smallest values of $A_3 + A_4$. The best D-efficiencies are also included in the set with smallest A_3, but there is a substantial variation in efficiencies of the designs in that set. The much larger variation in D-efficiencies for sets of OAs with one and the same A_3 value shows that it does not suffice to consider A_3 alone.

The results of 40-run OAs in five factors confirm the findings stated for OA$(36, 5, 2)$. In this case, the smallest A_3 value equals zero. These designs therefore have are strength of 3; clearly these do not have the best D-efficiency among all 5-factor designs.

The results for seven-factor designs show that designs with small $A_3 + A_4$ values have good D-efficiencies; within a set of OAs with the same value of this sum, the variation in D-efficiencies is small.

We conclude first that a minimum G_2-aberration design, which minimizes the elements of the GWLP from left to right, does not necessarily have the best D-efficiency. Second, minimizing $A_3 + A_4$ is approximately equivalent to maximizing D-efficiency. One interesting subject for further research would therefore be the design generation with small values of this sum.

D Coordinate exchange algorithm

Our coordinate exchange algorithm optimizes $F = \alpha_1 \mathrm{D} + \alpha_2 \mathrm{D}_s + \alpha_3 D_1$, where the parameters $\alpha = (\alpha_1, \alpha_2, \alpha_3)$ are adjustable. Note that for any choice of parameters α the set $(s\alpha_1, s\alpha_2, s\alpha_3)$ for $s > 0$ results in the same set of designs.

The algorithm requires specification of the run size N, the number of factors n, the number of restarts, and the weights α. The end result is a set of Pareto optimal designs, where the designs are classified according to the efficiencies that have a non-zero value in the vector α. In the main paper, we present the best designs for $\alpha = (1, 0, 0)$ and $\alpha = (1, 2, 0)$.

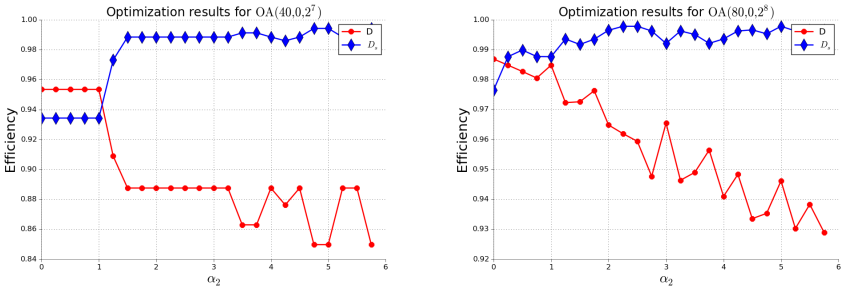
A brief description of the algorithm follows.

- Specify the number of restart groups G and the number of restarts per group T.
- For each restart generate a starting design and determine a random order to visit all of the coordinates. For each coordinate, find out whether a sign switch improves the objective function and immediately apply any improvement found. Proceed until no further improvements can be found.
- For each group of T designs, retain the Pareto optimal designs and reduce to a set of non-isomorphic designs. The Pareto-optimality of a design is determined from the efficiencies that have a non-zero value in the vector α.
- Merge the G sets of designs into one set.
- For each design in the retained set, evaluate swapping two coordinates of the design until no further improvement occurs. Then evaluate sign switches until no further improvement occurs. Perform several rounds of swapping and switches until no further improvement occurs.
- Reduce the resulting set of designs by retaining only the non-isomorphic Pareto optimal designs.

Typical computing times were about 30 minutes for 5,000 tries on 64-run 10-factor designs down to about 50 seconds for 28-run 6-factor designs. Computing times can be improved by adopting fast updating formulas (Meyer and Nachtsheim, 1995) to the three components of the objective function F.

D.1 Selection of optimization parameters

Maximizing the D_s-efficiency of a design implies maximizing the D_1-efficiency. Even for $\alpha = (1, 0, 0)$, the D_1-efficiencies of the designs turn out to be high. We therefore set $\alpha_3 = 0$ for all applications we discuss. To determine an optimal setting of the two remaining parameters, we can fix any of them and study the effect of varying the other one on D-efficiencies and D_s-efficiencies of the resulting designs. We performed series of optimizations with $\alpha_1 = 1$ and $0 \leq \alpha_2 < 6$ in steps of 0.25. The cases we used were 40-run 7-factor designs and 80-run 8-factor designs. In Figure 8 the results are plotted. In both cases, the D_s-efficiency does not really improve for $\alpha_2 > 2$, while the D-efficiency is decreasing for the 80-run case. For this reason we select $\alpha = (1, 2, 0)$ as the optimization parameters for the compound criterion.



(a) $N = 40, n = 7$ (b) $N = 48, n = 8$

Figure 8: D-efficiencies and D_s-efficiencies for various settings of α_2 in the objective function $F = D + \alpha_2 D_s$. The designs whose efficiencies are shown have the best F obtained after 1600 restarts.

E Generation of OAs

We generated strength-2 arrays for efficient estimation of the interaction model in 5–8 factors and 16–44 runs, and strength-3 arrays in 7–10 factors and 40–72 runs. This appended section records details of the numbers on isomorphism classes, the number of arrays that were generated in each class, their D-efficiencies and the computing times needed.

E.1 Strength-2 arrays

Table 5 shows results for the strength-2 arrays. The first two columns give the run size N and the number of factors n. For run sizes 32, 36, 40 and 44, respectively, the series with five factors were fully generated and series with more than five factors were partially generated based on the theory developed in Section A. To get an appreciation of the arrays that were missed in a partial generation, we included for $N = 32$ results for fully generated series as well as for partially generated series.

The third column in Table 5 gives the number of isomorphism classes. For the fully generated cases, the stated number is exact. For the partially generated cases, the stated number is the estimated order of magnitude of the total number of isomorphism classes, as indicated by the tilde. To estimate the order of magnitude, we started with the complete series of five-factor arrays. Then, we took a small randomly chosen set of these arrays and expanded arrays in this set to the maximum number of factors that might permit estimation of the interaction model. The total number of arrays found for the various factor numbers is then divided by the fraction of five-factor arrays that were extended and rounded to the nearest power of 10.

The set of generated arrays was split in a set of arrays that were extended with an extra factor and a set of arrays that were not. The table shows the number of arrays in each of these

Table 5: Enumeration results for strength-2 arrays to fit the interaction model in up to 8 factors. Upper panel: full enumeration for OAs with $N \leq 28$ and partial enumeration of OAS with $32 \leq N \leq 44$; lower panel: full enumeration of OAs with $N = 32$.

N	n	Isomorphism classes	Not extended		Extended		B
			Number	max(D)	Number	max(D)	
16	5	11	0	-	11	1	1
20	5	11	0	-	11	0.8661	0.8661
24	5	63	0	-	63	0.939	0.939
	6	1,350	0	-	1,350	0.7926	0.7926
28	5	127	0	-	127	0.9409	0.9409
	6	17,826	0	-	17,826	0.8855	0.8855
32	5	491	139	0.7553	352	1	1
	6	266,217	221,522	0.8448	4,000	1	1
	7	530,469,996	5,861,610	0.8432	0	-	0.8799
36	5	1,242	896	0.892	346	0.9635	0.9635
	6	$\sim 10^6$	1,202,815	0.8972	2,777	0.9374	0.9374
	7	$\sim 10^{10}$	15,545,886	0.9022	0	-	0.9389
40	4	32	2	0	30	0.9889	0.9889
	5	3,910	3,310	0.8899	600	0.9781	0.9781
	6	$\sim 10^7$	6,953,911	0.9355	300	0.9657	0.9657
	7	$\sim 10^{12}$	2,888,903	0.9021	300	0.9245	0.9414
	8	$\sim 10^{16}$	97,949	0.8019	0	-	0.9506
44	5	10,151	10,051	0.9464	100	0.9769	0.9769
	6	$\sim 10^8$	2,046,240	0.9591	30	0.9622	0.9622
	7	$\sim 10^{14}$	121,193	0.9266	100	0.9449	0.9531
	8	$\sim 10^{19}$	7,618	0.8524	0	-	0.9567
32	5	491	0	-	491	1	1
	6	266,217	0	-	266,217	1	1
	7	530,469,996	0	-	530,469,996	0.8432	0.8432

sets as well as the D-efficiency of the best array in each of the sets. The total number of arrays we actually obtained is the sum of the numbers in each of the sets.

Finally, the last column of Table 5 shows an upper bound on D-efficiencies for arrays that might have been obtained if the series were generated fully. This bound, which is designated B, was obtained using the theory developed in the Section A.

Table 5 shows that the 32-run cases with five and six factors include arrays with a D-efficiency equalling 1. So these arrays have a strength $t \geq 4$. The arrays are included because any array of strength $t \geq 4$ also meets the definition of a strength-2 array. For example, the generated arrays series of OA$(32, 5, 2)$ also include the full factorial in five factors, which is of strength 5.

There are 11 partially generated series of in the table. In spite of the partial generation, all the six-factor series and the partially generated series of OA$(32, 7, 2)$ include an array with the best possible D-efficiency of the interaction model. The remaining seven-factor series have discrepancies of at most 0.0367 between the D-efficiency in the best array obtained and the upper bound.

For the best OA$(40, 8, 2)$ and the best OA$(44, 8, 2)$, the upper bound B is higher by 0.1487 and 0.1043, respectively. These are substantial discrepancies. However, in both cases, the bound is based on $T_{5,8}$. Extending more five-factor arrays would result in millions of extra arrays, which is computationally inconvenient. Further, we show in the main paper that the best arrays actually obtained are competitive with the best strength-2 arrays known from the literature. We therefore did not search for better arrays.

Table 6: Enumeration results for strength-3 arrays.

N	n	Isomorphism classes	Not extended		Extended		B
			Number	max(D)	Number	max(D)	
40	7	25	0	-	25	0.8030	0.8030
	8	105	105	0	0	-	0
48	7	397	0	-	397	0.9585	0.9585
	8	8,383	0	-	8,383	0.8365	0.8365
	9	166,081	166,081	0.6753	0	-	0.6753
56	8	757,190	746,699	0	10,491	0.8642	0.8642
	9	$\sim 10^8$	17,533	0	28	0.761	0.761
	10	$\sim 10^{10}$	1	0	0	-	0
64	6	358	32	0	326	1	1
	7	91,789	90,889	0.9414	900	1	1
	8	$\sim 10^7$	8,336	0.9241	3,000	1	1
	9	$\sim 10^{11}$	5	0	2,229	0.9254	0.9626
	10	$\sim 10^{14}$	52	0.8247	0	-	0.9692
72	6	906	34	0	872	0.9745	0.9745
	7	2,147,836	2,147,656	0.9535	180	0.9553	0.9553
	8	$\sim 10^9$	8,609	0.9232	2000	0.9283	0.9439
	9	$\sim 10^{13}$	0	-	5	0.8818	0.9391
	10	$\sim 10^{16}$	0	-	0	-	0.9369

E.2 Strength-3 arrays

Table 6 shows results for the strength-3 arrays. The table has the same layout as the one for the strength-2 results. The series in 40 and 48 runs were completely generated. For the 56-run arrays, we completely generated the series with eight-factor arrays. Nine-factor arrays were only generated by extension of the 10,491 eight-factor arrays that support an interaction model. The extension resulted in only 28 nine-factor arrays that permit fitting an interaction model in this number of factors. Further extension resulted in a single ten-factor array. However, it is not possible to fit an interaction model based on this array.

For the 64-run arrays, we completely generated the series with six factors. We extended all 326 arrays that support an interaction model. From seven factors onward, we retained at most a few thousands of arrays that support an interaction model.

For the 72-run arrays, we again started with generating all six-factor arrays. We extended all 872 arrays that support an interaction model. This resulted in more than two million seven-factor arrays. From seven factors onward, we retained only a part of the arrays that support an interaction model. We obtained five nine-factor arrays and we failed to obtain a ten-factor array.

Table 6 shows 64-run cases for 6–8 that include arrays with a strength $t \geq 4$. Therefore, both the D-efficiencies for these arrays and the upper bound for these efficiencies equal 1. For the nine-factor arrays, the best D-efficiency found is 0.0372 lower than the upper bound, while the discrepancy is 0.1445 for the ten-factor arrays. The upper bounds are determined by $T_{7,9}$ and $T_{7,10}$, respectively. Therefore, extending more seven-factor arrays might result in a smaller discrepancy. However, as shown in Section 5.3 for nine-factor arrays and in Section 5.4 for ten-factor arrays, the arrays we did obtain are competitive with the best literature arrays. At the same time, extending only a few more seven-factor arrays leads to millions of extra arrays in eight factors. It is computationally infeasible to process all these arrays. For these reasons, we did not increase the number of retained seven-factor arrays.

For the 72-run cases, the discrepancy between the best D-efficiency and the upper bound for

Table 7: Computing times (in hours) for the generation of the strength 2 arrays in Table 5 and strength 3 arrays in Table 6. All calculations have been performed on a PC with Intel Core i7 870 CPU at 2.93GHz.

Strength 2			Strength 3		
N	n	Time	N	n	Time
16	5	0.0	40	6	0.0
20	5	0.0		7	0.0
24	5	0.0		8	0.0
	6	0.0	48	6	0.0
28	5	0.0		7	0.0
	6	0.0		8	0.0
32	5	0.0		9	0.4
	6	0.0	56	8	1.4
	7	5.0		9	1.7
36	5	0.0		10	1.7
	6	0.4	64	6	0.0
	7	12.7		7	0.2
40	5	0.0		8	2.2
	6	3.4		9	3.3
	7	16.4		10	3.6
	8	23.0	72	6	0.0
44	5	0.0		7	4.8
	6	2.6		8	10.5
	7	10.9		9	37.9
	8	23.4			

eight and nine factor arrays is 0.0156 and 0.0573, respectively. Our failure to find a 72-run ten-factor array did not prompt us to extend more than the few hundred seven-factor or eight-factor arrays with this run size, because even a few arrays extra lead to millions of new extensions. We believe that 72 is the maximum run size for which our methodology can yield useful results.

E.3 Computing times

Table 7 shows the computing times for the OAs we generated. It is clear form the table that the computing times for strength-2 OAs become substantial from 36 runs and 7 factors onward. This is directly related to the huge numbers of isomorphism classes for these series. For strength-3 OAs, computing times become really substantial only for the 72-run cases.

References

Addelman, S. (1961). Irregular fractions of 2^n factorial experiments. *Technometrics*, 3:479–496.

Atkinson, A. C., Donev, A. N., and Tobias, R. D. (2007). *Optimum Experimental Designs, with SAS*. Oxford: Oxford University Press.

Cheng, C. S., Deng, L. Y., and Tang, B. (2002). Generalized minimum aberration and design effciency for nonregular fractional factorial designs. *Statistica Sinica*, 12:991–1000.

DuMouchel, W. and Jones, B. (1994). A simple Bayesian modification of D-optimal designs to reduce dependence on an assumed model. *Technometrics*, 36:37–47.

Eendebak, P. T. (2015). The Orthogonal Array package. `http://www.pietereendebak.nl/oapackage/index.html`.

Hedayat, A. S., Sloane, N. J. A., and Stufken, J. (1999). *Orthogonal Arrays: Theory and Applications*. Springer.

John, P. W. M. (1962). Three-quarter replicates of 2^n designs. *Biometrics*, 18:319–321.

Jones, B. and Nachtsheim, C. J. (2011). Efficient designs with minimal aliasing. *Technometrics*, 53:62–71.

Li, W. and Nachtsheim, C. J. (2000). Model-robust factorial designs. *Technometrics*, 42:345–352.

Li, X., Sudarsanam, N., and Frey, D. D. (2006). Regularities in data from factorial experiments. *Complexity*, 11:32–45.

Mee, R. W. (2004). Efficient two-level designs for estimating all main effects and two-factor interactions. *Journal of Quality Technology*, 36:400–412.

Mee, R. W. (2009). *A Comprehensive Guide to Factorial Two-Level Experimentation*. Springer-Verlag, New York.

Meyer, R. K. and Nachtsheim, C. J. (1995). The coordinate-exchange algorithm for constructing exact optimal experimental designs. *Technometrics*, 37:60–69.

Rao, C. R. (1947). Factorial experiments derivable from combinatorial arrangements of arrays. *Journal of the Royal Statistical Society Supplement*, 9:128–139.

Sartono, B., Goos, P., and Schoen, E. D. (2012). Classification of three-level strength-3 arrays. *Journal of Statistical Planning and Inference*, 142:794–809.

Schoen, E. D. (2010). Optimum designs versus orthogonal arrays for main effects and two-factor interactions. *Journal of Quality Technology*, 42:197–208.

Schoen, E. D., Eendebak, P. T., and Nguyen, M. V. M. (2010). Complete enumeration of pure-level and mixed-level orthogonal arrays. *Journal of Combinatorial Designs*, 18:123–140.

Schoen, E. D. and Mee, R. W. (2012). Two-level designs of strength 3 and up to 48 runs. *Journal of the Royal Statistical Society Series C*, 61:163–174.

Schur, J. (1917). Über Potenzreihen, die im Innern des Einheitskreises beschränkt sind. *Journal für die reine und angewandte Mathematik*, 147:205–232.

Smucker, B. J., Del Castillo, E., and Rosenberger, J. L. (2012). Model-robust two-level designs using coordinate exchange algorithms and a maximin criterion. *Technometrics*, 54:367–375.

Tang, B. and Deng, L. Y. (1999). Minimum G_2-aberration for nonregular fractional factorial designs. *Annals of Statistics*, 27:1914–1926.

Tang, B. and Zhou, J. (2013). D-optimal two-level orthogonal arrays for estimating main effects and some specified interactions. *Metrika*, 76:325–337.

Wu, C. F. J. and Hamada, M. (2000). *Experiments: Planning, Analysis and Parameter Design Optimization*. 1st edn, Wiley, New York.