Songklanakarin J. Sci. Technol.
42 (6), 1259-1273, Nov. - Dec. 2020

http://www.sjst.psu.ac.th

Original Article

A weighted D-optimality criterion for constructing model-robust designs in the presence of block effects

Peang-or Yeesa[1], Patchanok Srisuradetchai[1*], and John J. Borkowski[2]

[1] *Department of Mathematics and Statistics, Faculty of Science and Technology, Thammasat University, Khlong Luang, Pathum Thani, 12121 Thailand*

[2] *Department of Mathematical Sciences, Montana State University, Bozeman, Montana, 59717-2400 United States of America*

Received: 30 March 2019; Revised: 22 July 2019; Accepted: 2 September 2019

Abstract

It is generally known that blocking can reduce unexplained variation, and in response surface designs block sizes can be pre-specified. This paper proposes a novel way of weighting D-optimality criteria obtained from all possible models to construct robust designs with blocking factors and addresses the challenge of uncertainty as to whether a first-order model, an interaction model, or a second-order model is the most appropriate choice. Weighted D-optimal designs having 2 and 3 variables with 2, 3, and 4 blocks are compared with corresponding standard D-optimal designs in terms of the D_N-efficiencies. Effects of blocking schemes are also investigated. Both an exchange algorithm (EA) and a genetic algorithm (GA) are employed to generate the model-robust designs. The results show that the proposed D_w-optimality criterion can be a good alternative for researchers as it can create robust designs across the set of potential models.

Keywords: experimental design, D-optimality, weak heredity, genetic algorithm

1. Introduction

Many real-life situations exist whereby experimental designs can help solve a research question. Experimentation is a scientific approach to learn how a system or process works, and experimental design is an essential tool for improving the performance of an industrial manufacturing process. Response surface designs are considered a class of experimental designs that are useful for developing, improving and optimizing a process (Myers, Montgomery, & Anderson-Cook, 2016). Response surface methodology (RSM) involves both statistical and mathematical techniques and is useful for three purposes: (i) fitting a response surface model over a specific region of interest, (ii) finding the optimal response, and (iii) selection of operating conditions to achieve some specifications or customer requirements. RSM is primarily concerned with approximating a complicated unknown function with a low-order polynomial, usually either a first-order model, an interaction model, or a second-order model.

If data for every combination of factor levels cannot be collected under identical experimental conditions, then blocks should be formed. A second-order response surface model with *k* design variables and *b* blocks can be expressed as;

$$y(x_1,x_2,...,x_k,l)=\beta_0+\sum_{i=1}^{k}\beta_i x_i+\sum_{i=1}^{k}\beta_{ii}x_i^2+\sum_{i=1}^{k-1}\sum_{j=i+1}^{k}\beta_{ij}x_i x_j+\sum_{m=1}^{b-1}\delta_l I(m)+\varepsilon, \quad (1)$$

where $y(x_1,x_2,...,x_k,l)$ is an observed response, given $x_1,x_2,...,x_k$ are the k design variables, and l is the block identifier with $I(m)$ being a block indicator function. The β's and the δ's correspond to the second-order model and block

*Corresponding author
Email address: spatchan@tu.ac.th

effect parameter coefficients to be estimated, and ε is a and variance σ^2. This second-order model is commonly used for describing a process of interest. Generally, a proposed model has many choices of design; therefore, selecting a "good" design is very important. To generate a design, many criteria and desired characteristics such as the number of design points, the number of blocks, and block sizes are required to be predetermined. One approach to select a design involves using optimality criteria, which strongly depend on the proposed 'prior' approximating the response surface model, e.g. equation (1). If a different model space is assumed, then the efficiency of the design changes. After the data are collected and parameters corresponding to terms of the proposed model are fitted, insignificant terms are removed so that the reduced 'posterior' model retains only 'significant' terms (Borkowski & Valeroso, 2001).

The design optimality criteria focused on in this research are based on D-optimality proposed by Wald (1943). The goal of D-optimality is to find a design that minimizes the value of $|(X'X)^{-1}|$ or the generalized variance $|\mathrm{var}(\hat{\theta})|$ where $\hat{\theta}$ is the vector of parameter estimates and X is the model matrix in the linear model. For the model in equation (1), $\hat{\theta}$ is the vector containing all $\hat{\beta}$ estimates for the second-order model and $\hat{\delta}$ estimates for the block effects. Thus, also the block effects will be estimated. Because $|X'X|$ equals $1/|(X'X)^{-1}|$, minimizing $|(X'X)^{-1}|$ is equivalent to maximizing $|X'X|$. The simplest and most common efficiency measure for D-optimality is the following:

$$D_N\text{-efficiency} = \left(\frac{|(X'X)|}{N^p}\right)^{1/p} \times 100 = \frac{|(X'X)|^{1/p}}{N} \times 100, \qquad (2)$$

where p is the number of the model parameters. The D_N-efficiency in (2) is the typical measure calculated by various software packages (e.g., by PROC OPTEX in the SAS software package). D_N-efficiency can be interpreted as the relative efficiency of a design compared to an optimal hypothetical orthogonal design in the hypercube (Mitchell, 1974b). Note that D_N-efficiencies can be used to compare designs of different sizes. For additional details and examples of design optimality criteria see Atkinson, Donev, and Tobias (2007).

Many different strategies have been developed to examine a set of potential reduced models. Chipman (1996) presented two classes of reduced models based on weak heredity (WH) and strong heredity (SH) principles. A model can be summarized by vector Δ containing '1' and '0' where '1' indicates that a term is included in the model and '0' that it is not in the model. The notations Δ_i, Δ_{ii}, and Δ_{ij} represent the indicator function values of the i^{th} first-order effect, the ii^{th} second-order effect, and the ij^{th} interaction effect, respectively. Weak heredity (WH) requires that if the $\beta_{ij}x_ix_j$ term is in the model, then either the $\beta_i x_i$ or $\beta_j x_j$ term is (or both of them are) contained in the model, and if the $\beta_{ii}x_i^2$ term is in the model, then the $\beta_i x_i$ term must also be in the model. For $k=2$, the second-order model (without blocks) has 6 parameters, and there are 17 WH reduced models corresponding to vectors $\Delta = (\Delta_0, \Delta_1, \Delta_2, \Delta_{11}, \Delta_{22}, \Delta_{12})$. For $k=3$, there are 185 reduced models where the second-order model has 10 parameters (without blocks).

Because of experimental uncertainty with the reduced response surface model prior to data collection, the researcher should consider robust experimental designs with respect to a design optimality criterion over a set of potential posterior models. Thus, the purpose of this paper is to find a response surface design that has a good optimality criterion evaluated across the set of reduced models.

Many publications have used a weighted criterion based on the arithmetic mean, for example, Chomtee and Borkowski (2005) developed D, A, G, and IV optimality criteria using prior probability assignments to model effects. Three design variables $(k=3)$ were considered in a spherical design region and over sets of reduced models based on weak and strong heredity. The spherical response surface designs compared were central composite designs (CCDs), Box-Behnken designs (BBDs), small composite designs (SCDs), uniform shell designs (Doehlert, 1970; Doehlert & Klee, 1972), and hybrid designs (Roquemore, 1976). Chairojwattana, Chaimongkol, and Borkowski (2017) studied the weighted D- and G-optimality criteria (D_w and G_w) for second-order response surface designs using prior probability assignments to model effects and developed a genetic algorithm (GA) to generate designs that optimize D_w and G_w.

In our paper, D-optimality is used to define a weighted criterion to generate designs that are robust to a set of potential models. That is, the weighted D-optimality criterion (D_w) is used to evaluate designs. The goal of weighted D-optimality criterion (D_w) is to maximize the weighted average of D_N-efficiencies in the design region over a set of reduced models. The weights must be supplied by experimenters. One approach is to assign weights based on the number of parameters in each model.

Again, if experimental runs cannot be collected under identical experimental conditions, blocks should be formed. When observations can be recorded in blocks of homogeneous units, the blocking scheme depends on the nature of the experiment. Blocks introduce extra parameters into the model, considered nuisance parameters, while appropriate blocking of experimental designs can produce desirable experimental run features. The blocks may not necessarily have the same number of experimental runs; therefore, the novelty of this research involves introducing a weighted optimality across a set of reduced models in experiments that require blocking. Moreover, both an exchange algorithm and a genetic algorithm are implemented to generate optimal designs.

2. Materials and Methods

2.1 Weighted D-optimality criterion (D_w)

Let $w_i = \dfrac{p(i)}{N_p \times m(p(i))}$ as the weight for model *i*, where $m(p)$ is the number of models having p parameters, $p(i)$ is the difference between the number of parameters in model i and the number of blocks, and $N_p = \sum_{p=1}^{\binom{k+2}{2}} p$. Here, the model with more parameters has more weight and $\sum_{i=1}^{M} w_i = 1$ for a set of model weights $\{w_1, w_2, ..., w_M\}$ where M is the number of reduced models for a given full model. Based on the authors' experience and knowledge, they believe that each of the terms in the full model has a reasonably high probability of being significant. For this reason, the full model must have the highest probability or the largest weight. It is clearly seen that one term not being in the model will lead to a model with a slightly lower weight than the full model. And, as more terms are absent from the model, the weight will constantly decrease. This is the justification for why we require several distinct levels of weighting. The second-order model given in equation (1) stands for the full model in this study. These weights are used to calculate the weighted D-optimality criterion (D_w). For $k=2$, the second-order model (without blocks) has 6 parameters and $N_p = \sum_{p=1}^{6} p = 21$. For $k=3$, $N_p = \sum_{p=1}^{10} p = 55$ where the second-order model (without blocks) has 10 parameters.

Let Ξ be the set of all possible exact designs on design space χ, then the newly-proposed D_w-optimality criterion seeks a design ξ^* satisfying

$$\xi^* = \arg\max_{\xi \in \Xi} \left(\prod_{i=1}^{\mathrm{M}} \left| \mathrm{M}_i(\xi) \right|^{w_i} \right), \tag{3}$$

where $\mathrm{M}_i(\xi) = \mathrm{X}'_{(i)}\mathrm{X}_{(i)} / N$ is a moment matrix, for $\mathrm{X}_{(i)}$ which is the model matrix with columns corresponding to the terms in model i, and N is the design size. Therefore, the D_w-efficiency as a weighted optimality criterion can be defined as

$$\mathrm{D}_w = \prod_{i=1}^{M} \mathrm{D}_i^{w_i} \ , \tag{4}$$

where $\mathrm{D}_i = \dfrac{100 \left| \mathrm{X}'_{(i)}\mathrm{X}_{(i)} \right|^{1/p}}{N}$ is the D_N-efficiency of the i^{th}

reduced model. Thus, the use of geometric mean is considered for defining the D_w-optimality criterion. The design must be robust to model reduction and should be able to fit all parameters for all reduced models. Note that in the geometric mean if any $D_i \approx 0$, then $D_w \approx 0$. Also, if $D_i = 0$, then $D_w = 0$ implying that not all reduced model parameters can be fitted by that design. In particular, if $D_w = 0$, then the full model cannot be fitted. Thus, D_w addresses the robustness problem better than a weighted optimality criterion based on the arithmetic mean $D_A = \sum_{i=1}^{M} w_i D_i$. With D_A, there is no guarantee that all reduced models can be fitted, which is contrary to the goal of finding a model-robust design. Note that D_w, as defined in (4), can be used for any assignment of w_i weights that sum to 1. This gives the experimenter the flexibility to use a weighting scheme different than the one used in this research.

2.2 Exchange and genetic algorithms

This research includes an exchange algorithm (EA) and a genetic algorithm (GA) to generate designs that optimize the D_w-optimality criterion.

2.2.1 Exchange algorithm

Originally, exchange algorithms (EAs) were created by starting with a randomly chosen *n*-run design and then exchanging design points with points in a candidate set of points such that the initial set of *n* runs was improved by (i) adding an (n + 1)st run, chosen to achieve the maximum possible increase in $|X'X|$, and then (ii) subtracting (removing) the run in the resulting design to obtain the minimum possible decrease in $|X'X|$. Variations of EAs were developed by Fedorov (1972), Wynn (1972), Mitchell (1974a), and Cook and Nachtsheim (1980). For example, the original design was improved by subtracting a point first and then adding a point (Mitchell, 1974).

The methodology for generating designs that optimize the D_w -criterion using an EA is as follows:

1) Specify the number of design variables (k = 2 or k =3) and number of blocks (b = 2, 3, or 4) for N design points where $N = p, p+1, p+2, \ldots, p+9$ and p is the number of model parameters in equation (1).
2) Generate a candidate set C with N_C points, and then randomly generate a starting design point matrix of size $N \times k$ from points in C. In this research, $N_C = 21^k$ for $k = 2$ or $k = 3$. The 21 values are $x_i \in \{-1, -0.9, \ldots, 0.9, 1\}$.
3) Replace a point in the starting design with a point in C and calculate D_w. Do this for all $N \times N_C$ exchanges.
4) Keep the exchange and the design that has the largest D_w value .This is the *new best design* .
5) Iterate steps 3 and 4 until no further improvement is found in the D_w value .
6) Repeat steps 1 to 5 for 20 starting design. Keep the best designs resulting from these 20 starting designs.

2.2.2 Genetic algorithm

Basic genetic algorithms (GAs) were developed by Holland (1975) and applied to find solutions for complex problems in optimization, machine learning, programming, and job scheduling. GAs have recently been applied to generate optimal response surface designs based on the survival of the fittest biological imperative. That is, individuals (designs) adapt to their environment and then evolve into more desirable forms (Sivanandam & Deepa, 2008).

GAs have been extensively used in research. For example, Borkowski (2003) developed a GA to generate near-optimal D, A, G, and IV small exact *N*-point response surface designs in the hypercube. Designs were assessed for 1, 2, and 3 factors and performances of exact optimal designs were compared with classical responses having the same design size. Heredia-Langner, Carlyle, Montgomery, Borror, and Runger (2003) developed GAs to create D-optimal designs. Their results showed that GAs can be maintained at a level of performance comparable to coordinate exchange, k-exchange and modified Fedorov exchange algorithms. Thongsook, Borkowski, and Budsaba (2014) proposed and developed a GA to generate optimal designs for constrained mixture regions when quadratic terms are of primary interest. Limmun, Borkowski, and Chomtee (2018a) developed a GA to create a weighted A-optimality criterion to generate robust mixture designs. Limmun, Chomtee, and Borkowski (2018b) also developed a GA to generate weighted IV-optimal mixture designs and the results showed that their GA-generated designs were robust across a set of potential mixture models. A primary benefit of using a GA is that it does not limit the selection of design points from a finite candidate set, allowing points to be selected throughout a continuous region in the selection and reproduction processes. Mahachaichanakul and Srisuradetchai (2019) used the GA to construct robust response surface designs against missing data.

A gene can be defined as one row of a chromosome (design), and a genetic variable can be any of the k design variables in a gene (or row). Let x_{ij} be the j^{th} genetic design variable in i^{th} row of a chromosome. The methodology for generating designs that optimize the D_w-optimality criterion with a GA is as follows:

1) Specify the number of design variables (k = 2 or k =3), number of blocks (b = 2, 3, or 4), design size N where $N =$ p, p+1, p+2,…, p+9, and the number of chromosomes (designs) M which is an odd number in the GA population.
2) Randomly generate M chromosomes representing the population of design matrices for a hypercube design region.
3) Calculate the objective function D_w for each chromosome .
4) Select the elite chromosome giving the largest D_w value . The remaining $M-1$ chromosomes are randomly partitioned into $(M-1)/2$ pairs of parent chromosomes.
5) Apply 7 operators to each of these parental pairs in the reproduction process .The reproduction process operates on the genes to produce offspring chromosomes. Six of the operators are :swap rows, swap cut point pieces, swap coordinates, zero gene, extreme gene, and creep reproduction. Furthermore, a new operator called swap blocks is also used in this step. An operator is applied if a probability test is passed (PTIP). A PTIP happens if a random u is less than or equal to a value of α_i where $u \sim \text{Uniform}\,[0,1]$ and the α_i values are specified by the experimenter.
6) After the reproduction processes, we calculate D_w for each of the parent chromosomes and each of the offspring chromosomes . There are 1 elite, $M-1$ parents, and $M-1$ offspring chromosomes giving a total of $2M-1$ chromosomes at the end of a reproduction process.
7) Compare the objective function values for each parent and its corresponding offspring .The chromosome in that parent/offspring pair that produces the larger D_w value survives as a future parent, while the other chromosome is removed from the population .At the end of each generation, M chromosomes plus an elite chromosome exist to form the next generation .The chromosome with the best objective function values becomes the elite chromosome for the next generation.
8) Steps 6 and 7 are iterated until the GA cannot evolve a larger D_w value.
9) Take the best D_w design generated when the GA terminates.

A brief description of the reproduction operators used in step 5 is shown in Figure 1. Let A and B be the two parent designs paired in the reproduction process. For each operator, a probability test is performed on each row of A and B. For the swap rows gene operator, if a PTIP occurs for row A_a of A, the operator exchanges A_a with a random row B_b of B. For the swap cut point gene operator, if a PTIP occurs for

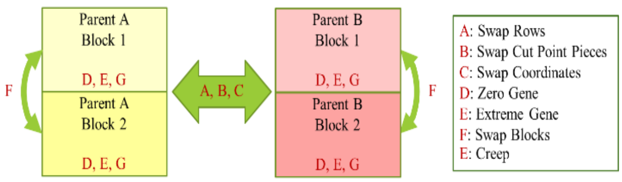

Figure 1. Diagram of GA reproduction operators.

row A_a of A, the operator changes the last two decimal digits of the *k* genetic design variables of A_a with the last 2 decimal digits of the *k* genetic design variables for a random row B_b of B. For the swap block gene operator, if a PTIP occurs for row l in block *b* (in either A or B), the operator exchanges row l in block *b* with a random row from another block. The remaining operators are applied to genetic variables in rows of either A or B. For the swap coordinates gene operator, if a PTIP occurs for x_{ij} of A, the operator exchanges x_{ij} of A with a random x_{kl} of B. For the zero gene operator, if a PTIP occurs for an x_{ij}, then x_{ij} is changed to 0. For the extreme gene operator, if a PTIP occurs for an x_{ij}, the x_{ij} is randomly set to either ±1. For the creep operator, if a PTIP occurs for an x_{ij}, then a random variate $\in \sim N(0,\sigma^2)$ is added to x_{ij} to create a new $x_{ij}^* = x_{ij} + \in$. The variance σ^2 is set by the researcher. If the creep operator takes $x_{ij}^* > 1$ or $x_{ij}^* < -1$, the value of x_{ij}^* is set to 1 or -1, respectively. The α values, which are Bernoulli parameters, for swap rows, swap cut point pieces, swap coordinates, zero gene, extreme gene, swap blocks, and creep are α_{sr}, α_{scp}, α_{sc}, α_z, α_e, α_{sb}, and α_c, respectively. The set of α_i values are constrained as follows: $0.002 \le \alpha_{sr}, \alpha_{sc}, \alpha_{sb} \le 0.02$, $0.005 \le \alpha_{scp} \le 0.02$, $0.01 \le \alpha_z \le 0.05$, $0.01 \le \alpha_e \le 0.10$, and $0.025 \le \alpha_c \le 0.10$.

3. Results and Discussion

Comparisons are divided into 2 parts for algorithms and for criteria.

3.1 Comparing algorithms

The resulting D_w-efficiencies for robust designs using EA and GA and their properties are shown in Tables 1 and 2. The $D_{w(EA)}$ and $D_{w(GA)}$ maximizing the D_w-optimality criterion are the D_w values of the designs generated by EA and GA, respectively. Design efficiencies tend to increase as design size, *N*, increases, and as the number of blocks increases, D_w-efficiencies tend to decrease for both designs generated by EA and GA for all choices of *N*.

D_w-efficiencies of GA designs are always greater than D_w-efficiencies of EA designs for all choices of *k*, *b*, and *N*, although the differences are only in the decimals. For example, in Table 1, $N = 7$ when sample sizes in the 1st and 2nd blocks are 3 and 4, respectively while the corresponding D_w-efficiencies of EA and GA designs are equal to 45.3186 and 45.3299, respectively. This happens because a GA uses all of the design space while an EA uses only a finite candidate set. With a larger candidate set, the D_w of an EA design would be closer to that of a GA design. Furthermore, for $k = 3$ compared to the case of $k = 2$, difference between the D_w-efficiencies of EA and GA increased. For example, in Table 2, designs with $k = 3$, $b = 4$, $N = 13$ for sample sizes in the 1st, 2nd, 3rd, and 4th blocks are 3, 3, 3, and 4, respectively, while D_w-efficiency of the GA design is equal to 33.1626 and greater than that of the EA design, which is 32.9659. These results are confirmed in Figure 2 where the boxplots of the differences of D_w-efficiencies between GA and EA designs are greater than zero for all cases. Thus, the GA designs are more efficient than the EA designs.

3.2 Comparing criteria

A comparison of D_N-efficiency for each best design under a first-order model (FOM), interaction model (INT), and second-order model (SOM) having $k = 2$, and 3 variables and $b = 2, 3$, and 4 blocks is shown in Tables 3 to 8. The “all models” columns correspond to designs generated by an EA or GA that maximize the D_w-efficiency and refers to robust designs obtained from weighting all WH reduced models (or “all models” in short) and the “full model only” columns correspond to designs generated by an EA or GA to maximize the D_N-efficiency for only the full second-order

Table 1. Summary of $\mathbf{D}_w$ -efficiencies for EA and GA designs having k = 2 variables and b = 2, 3, and 4 blocks.

		N									
k	b	7	8	9	10	11	12	13	14	15	16
2	2	(3,4)	(4,4)	(4,5)	(5,5)	(5,6)	(6,6)	(6,7)	(7,7)	(7,8)	(8,8)
$D_{w(EA)}$		45.3186	45.9035	46.2679	47.1559	47.1615	47.2751	47.5404	47.5576	47.7073	47.8090
$D_{w(GA)}$		45.3299	45.9092	46.2759	47.1619	47.1672	47.2860	47.5532	47.5731	47.7264	47.8163
k	b	8	9	10	11	12	13	14	15	16	17
2	3	(2,3,3)	(3,3,3)	(3,3,4)	(3,4,4)	(4,4,4)	(4,4,5)	(4,5,5)	(5,5,5)	(5,5,6)	(5,6,6)
$D_{w(EA)}$		35.6210	36.9209	37.2597	37.5597	38.1380	38.7030	38.8167	39.0539	38.9044	38.8806
$D_{w(GA)}$		35.6314	36.9380	37.2632	37.5632	38.1458	38.7045	38.8252	39.0642	38.9153	38.8849
k	b	9	10	11	12	13	14	15	16	17	18
2	4	(2,2,2,3)	(2,2,3,3)	(2,3,3,3)	(3,3,3,3)	(3,3,3,4)	(3,3,4,4)	(3,4,4,4)	(4,4,4,4)	(4,4,4,5)	(4,4,5,5)
$D_{w(EA)}$		28.3263	29.1803	30.0590	30.8923	31.3364	31.3622	31.4973	31.7441	31.9341	32.1744
$D_{w(GA)}$		28.3322	29.1855	30.0625	30.8942	31.3382	31.4776	31.5582	31.7458	31.9396	32.1782

Table 2. Summary of $\mathbf{D}_w$ -efficiencies for EA and GA designs having k = 3 variables and and 4 blocks.

		N									
k	b	11	12	13	14	15	16	17	18	19	20
3	2	(5,6)	(6,6)	(6,7)	(7,7)	(7,8)	(8,8)	(8,9)	(9,9)	(9,10)	(10,10)
$D_{w(EA)}$		47.4978	47.8026	47.8895	48.1629	48.4414	48.3310	48.4012	48.5206	48.7078	48.8252
$D_{w(GA)}$		47.5341	47.8063	47.9011	48.1754	48.4627	48.3637	48.4159	48.7449	48.7496	48.8372
k	b	12	13	14	15	16	17	18	19	20	21
3	3	(4,4,4)	(4,4,5)	(4,5,5)	(5,5,5)	(5,5,6)	(5,6,6)	(6,6,6)	(6,6,7)	(6,7,7)	(7,7,7)
$D_{w(EA)}$		39.5462	40.2678	40.8071	41.1888	41.4746	41.4849	41.6035	41.6806	41.7261	41.8125
$D_{w(GA)}$		39.5726	40.2773	40.8126	41.2111	41.4806	41.5194	41.6355	41.6917	41.7445	41.8219
k	b	13	14	15	16	17	18	19	20	21	22
3	4	(3,3,3,4)	(3,3,4,4)	(3,4,4,4)	(4,4,4,4)	(4,4,4,5)	(4,4,5,5)	(4,5,5,5)	(5,5,5,5)	(5,5,5,6)	(5,5,6,6)
$D_{w(EA)}$		32.9659	33.8070	34.2087	34.7366	35.1474	35.2465	35.6855	35.9687	35.9227	35.9383
$D_{w(GA)}$		33.1626	33.8767	34.2710	34.7740	35.1614	35.3555	35.7272	35.9708	35.9282	35.9716

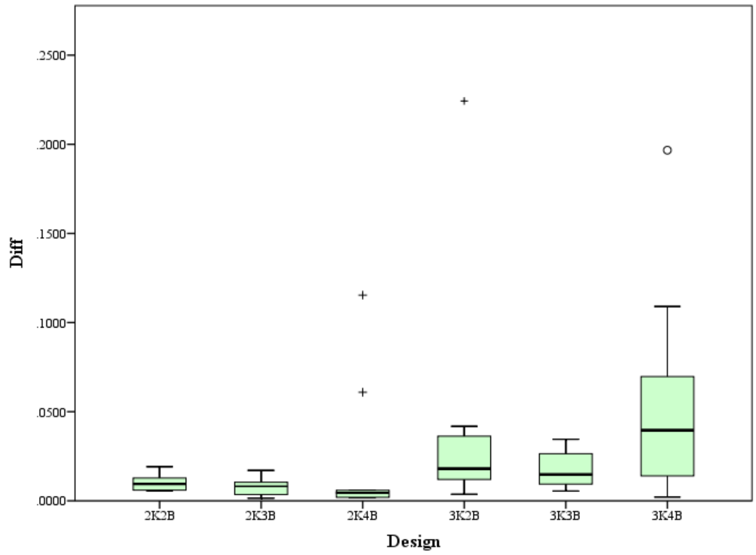

Figure 2. Boxplots of differences in D_w for GA and EA designs.

model with blocks.

Based on D_N-efficiencies, calculated for FOM and INT, designs from the D_w-optimality criterion are better than those obtained from the D_N-optimality criterion. This is because the best D_w designs provided higher D_N values than designs from the "full model only" for all choices of *k*, *b*, and N. This confirms that the D_w-optimality criterion can support all possible reduced models better than the D_N-optimality criterion. For example, in Table 3, for $k=2$, $b=2$ and $N=8$ when both block sizes are 4, D_N-efficiencies of the "all models" design and the "full model only" design for FOM of EA are equal to 59.1628 and 56.7554, respectively. For INT, they are 57.3188 and 55.3854, respectively. Similarly, D_N-efficiencies of the "all models" design and the "full model only" design for FOM of GA are equal to 59.2947 and 56.8460, respectively, while for INT, they are 57.4589 and 55.4381, respectively. Thus, the "all models" designs are more robust to model-misspecification than the "full model only" designs for all choices of *N*.

For EAs, D_N-efficiencies of designs generated for the "full model only" of SOM are greater than D_N-efficiencies of designs generated for the "all models" (best D_w designs) of SOM for all choices of *k*, *b*, and N because the goal of the "full model only" designs is to optimize the full second-order model with blocks. However, D_N-efficiency values for "all models" (best D_w designs) are close to optimal D_N-efficiency for the "full model only". Similar patterns are true for GAs. In situations where the full second-order model is fitted, results show that use of the D_w-optimality criterion provides a design which also has a D_N close to that of the full second-order model.

Boxplots of the differences in D_N values between "all models" designs and "full model only" designs for FOM, INT, and SOM are shown in Figure 3. For the FOM and INT, the boxplots are greater than zero. This indicates that the D_N- efficiencies of "all models" designs are, in general, better than those of "full model only" designs for FOM and INT.

For SOM, the boxplots are less than, but close to, zero. That is, the D_N- efficiencies of "all models" designs are slightly less than those of "full model only" designs, and the range of differences for SOM is smaller than the ranges for FOM and INT. This suggests that the D_N- efficiency values for SOM of the "all models" designs are close to those of the "full model only" designs.

Table 3. Summary of $\mathbf{D}_N$ -efficiencies of the "all models" and "full model only" designs having k = 2 variables and b = 2 blocks for a first-order model, interaction model, and second-order model.

				EA						GA					
				FOM		INT		SOM		FOM		INT		SOM	
k	b	N		All models	Full model only	All models	Full model only	All models	Full model only	All models	Full model only	All models	Full model only	All models	Full model only
2	2	7	(3,4)	58.7275	58.2751	56.6176	56.1919	39.9212	39.9337	58.5489	58.4685	56.6156	56.3704	39.9288	39.9494
		8	(4,4)	59.1628	56.7554	57.3188	55.3854	40.6651	40.7807	59.2947	56.8460	57.4589	55.4381	40.6464	40.8015
		9	(4,5)	61.1598	55.7252	61.0524	53.3313	40.1475	41.0263	61.2440	55.8305	61.1820	53.4133	40.1220	41.0411
		10	(5,5)	63.1529	61.2563	63.8663	60.5069	40.8423	41.3772	63.1452	61.1764	63.8934	60.3778	40.8399	41.3868
		11	(5,6)	63.2283	59.6267	65.0232	58.1888	40.8300	41.9482	63.2332	59.5643	65.0273	58.0788	40.8421	41.9572
		12	(6,6)	62.0589	60.5134	62.9543	59.5791	41.6696	42.0731	62.0998	60.5704	62.9914	59.6843	41.6748	42.0909
		13	(6,7)	61.4406	60.9440	61.4601	61.0624	42.4615	42.5602	61.5100	61.0341	61.5196	61.1391	42.4603	42.5770
		14	(7,7)	62.3148	59.1194	62.3092	58.7189	42.2303	42.7400	62.2758	59.0199	62.2733	58.6414	42.2600	42.7642
		15	(7,8)	62.8407	59.9977	63.3266	59.6361	42.2093	42.6294	62.8184	60.1480	63.2955	58.7252	42.2427	42.6416
		16	(8,8)	61.0093	60.8075	61.1568	60.8335	42.6336	42.6574	60.9429	60.8815	61.0474	60.9486	42.6671	42.6779

Table 4. Summary of $\mathbf{D}_N$ -efficiencies of the "all models" and "full model only" designs having k = 2 variables and b = 3 blocks for a first-order model, interaction model, and second-order model.

				EA						GA					
				FOM		INT		SOM		FOM		INT		SOM	
k	b	N		All models	Full model only	All models	Full model only	All models	Full model only	All models	Full model only	All models	Full model only	All models	Full model only
2	3	8	(2,3,3)	44.1623	42.1153	43.8433	41.9220	33.1459	33.5021	44.2519	42.2463	43.9198	41.9298	33.1352	33.5112
		9	(3,3,3)	45.7636	43.3963	46.5666	43.2675	34.3039	34.7871	45.7765	43.3539	46.7459	43.1732	34.3233	34.7883
		10	(3,3,4)	44.2395	44.1811	43.9253	43.6973	35.6013	35.6366	44.2330	44.0927	43.8906	43.5156	35.6171	35.6504
		11	(3,4,4)	44.6205	44.3695	45.1566	44.7487	35.9030	35.9908	44.5817	44.3552	45.0801	44.7001	35.9294	35.9978
		12	(4,4,4)	44.8219	44.5060	46.6128	46.1726	36.5458	36.5723	44.8910	44.6013	46.6635	46.3034	36.5432	36.5882
		13	(4,4,5)	45.6773	45.5774	48.0436	47.9312	37.1931	37.2087	45.6555	45.5470	48.0171	47.8977	37.2054	37.2295
		14	(4,5,5)	46.4584	46.3793	48.8013	48.6610	37.0995	37.1065	46.4405	46.3940	48.7671	48.6879	37.1184	37.1274
		15	(5,5,5)	47.1664	47.0453	49.7850	49.5215	37.1277	37.1489	47.1147	47.0905	49.6720	49.6193	37.1648	37.1690
		16	(5,5,6)	47.2648	45.8050	50.0711	47.9918	36.9059	37.1144	47.2380	45.7985	50.0415	47.9785	36.9292	37.1390
		17	(5,6,6)	47.2954	44.8725	50.5597	45.9577	36.8375	37.1238	47.3197	44.8705	50.5835	45.9484	36.8354	37.1409

Table 5. Summary of $\mathbf{D}_N$ -efficiencies of the "all models" and "full model only" designs having k = 2 ariables and b = 4 blocks for a first-order model, interaction model, and second-order model.

				EA						GA					
				FOM		INT		SOM		FOM		INT		SOM	
k	*b*	*N*		All models	Full model only	All models	Full model only	All models	Full model only	All models	Full model only	All models	Full model only	All models	Full model only
2	4	9	(2,2,2,3)	32.2993	31.9228	32.7652	32.4634	27.6857	27.7442	32.4172	31.8614	32.8563	32.4762	27.6634	27.7506
		10	(2,2,3,3)	33.7574	32.0701	34.1264	31.6522	28.5535	28.8527	33.6174	32.0741	34.0644	31.7263	28.5941	28.8664
		11	(2,3,3,3)	34.3593	32.6362	35.8542	33.1442	29.6293	29.6839	34.3904	32.6506	35.9025	33.1951	29.6227	29.6891
		12	(3,3,3,3)	35.5543	33.2052	36.6957	35.1373	30.4610	30.6839	35.5461	33.3092	36.7112	35.1513	30.4629	30.6883
		13	(3,3,3,4)	35.5181	35.4428	38.0984	38.0226	31.1666	31.1680	35.5121	35.4798	38.0798	38.0273	31.1734	31.1815
		14	(3,3,4,4)	35.1509	35.0827	37.3672	37.2561	31.4241	31.4325	34.8690	34.8218	36.8275	36.7839	31.6513	31.6607
		15	(3,4,4,4)	35.1278	34.9845	37.1339	36.8742	31.6662	31.7255	34.9730	34.9443	37.2766	36.8545	31.6542	31.7296
		16	(4,4,4,4)	35.2993	35.1939	37.8703	37.4186	31.7452	31.8402	35.3049	35.1982	37.8858	37.4237	31.7456	31.8491
		17	(4,4,4,5)	35.5308	35.4935	38.5721	38.5201	31.9370	31.9506	35.5574	35.5353	38.6006	38.0430	31.9412	32.0220
		18	(4,4,5,5)	36.0036	35.8990	39.3347	39.2181	32.1524	32.1789	35.9647	35.9334	39.2875	39.2526	32.1805	32.1884

Table 6. Summary of $\mathbf{D}_N$ -efficiencies of the "all models" and "full model only" designs having k = 3 variables and b = 2 blocks for a first-order model, interaction model, and second-order model.

				EA						GA					
				FOM		INT		SOM		FOM		INT		SOM	
k	*b*	*N*		All models	Full model only	All models	Full model only	All models	Full model only	All models	Full model only	All models	Full model only	All models	Full model only
3	2	11	(5,6)	65.6560	65.4735	66.9441	66.3697	41.6184	41.6663	66.2231	65.3883	68.1986	66.0825	41.0993	41.6721
		12	(6,6)	65.3052	65.3052	66.5030	66.5030	42.1235	42.1235	65.3064	65.2554	66.5043	66.4792	42.1279	42.1348
		13	(6,7)	64.1127	64.0260	62.9689	62.8619	43.0104	43.0223	64.0728	63.7236	62.8969	62.7178	43.0462	43.1165
		14	(7,7)	64.9174	64.8318	63.9515	63.7522	43.1592	43.1686	64.9070	60.9477	63.9327	59.4993	43.1841	43.4424
		15	(7,8)	65.5188	65.3767	65.0632	64.9091	43.3036	43.3093	65.4707	65.3680	65.0055	64.8934	43.3293	43.3407
		16	(8,8)	65.7976	64.9951	65.7313	63.9174	43.0484	43.1733	65.8623	65.0351	65.8291	63.9875	43.0748	43.1924
		17	(8,9)	66.7618	64.6088	66.5468	63.0253	42.9037	43.2504	66.6493	64.6757	67.2171	63.1342	42.9584	43.2857
		18	(9,9)	67.5481	64.4086	67.9130	62.3645	42.9017	43.4297	67.6246	64.4191	68.4274	62.4006	43.1979	43.4564
		19	(9,10)	68.0320	65.2857	69.3543	63.8900	42.9720	43.6777	67.8893	65.3451	69.2534	63.9834	43.0922	43.6995
		20	(10,10)	67.0066	66.1520	67.0238	65.0845	43.6216	43.7963	67.0017	65.8811	67.0099	64.9516	43.6394	43.8042

Table 7. Summary of $\mathbf{D}_N$ -efficiencies of the "all models" and "full model only" designs having k = 3 variables and b = 3 blocks for a first-order model, interaction model, and second-order model.

				EA						GA					
				FOM		INT		SOM		FOM		INT		SOM	
k	b	N		All models	Full model only	All models	Full model only	All models	Full model only	All models	Full model only	All models	Full model only	All models	Full model only
3	3	12	(4,4,4)	50.4961	49.6227	54.0767	53.0097	36.2050	36.5300	50.5354	49.5135	54.0454	53.0482	36.2527	36.5520
		13	(4,4,5)	49.0352	48.9430	51.6922	51.6590	38.2677	38.2888	49.0672	48.8100	51.7618	51.3131	38.2682	38.3253
		14	(4,5,5)	50.1578	50.1578	53.4958	53.4958	38.6039	38.6039	50.2145	49.9632	53.5262	53.1264	38.5891	38.6363
		15	(5,5,5)	51.2576	50.1544	54.4945	52.8524	38.8155	39.0469	51.3358	50.1700	54.7352	52.9346	38.8307	39.0924
		16	(5,5,6)	51.2571	51.1749	55.2485	55.1203	39.1810	39.2018	51.1989	51.1635	55.1525	55.0977	39.2053	39.2118
		17	(5,6,6)	51.3682	50.3556	55.8527	53.3218	39.1514	39.2786	51.4423	50.4333	56.0163	53.4644	39.1849	39.3093
		18	(6,6,6)	51.7243	51.1298	56.8056	55.5703	39.2259	39.2485	51.8104	51.7664	56.9718	56.9095	39.2550	39.2622
		19	(6,6,7)	52.0195	50.5110	57.7519	54.2435	39.2442	39.5382	52.0294	50.5234	57.7736	54.2452	39.2531	39.5514
		20	(6,7,7)	52.8406	49.9526	58.3980	53.3348	39.1203	39.5954	52.8326	50.5143	58.3907	53.7663	39.1552	39.6011
		21	(7,7,7)	53.4591	49.9804	59.0642	52.1625	39.0981	39.7354	53.4576	50.0112	59.0568	52.1936	39.1160	39.7463

Table 8. Summary of $\mathbf{D}_N$ -efficiencies of the **"**all models**"** and **"**full model only**"** designs having k = 3 variables and b = 4 blocks for a first-order model, interaction model, and second-order model.

				EA						GA					
				FOM		INT		SOM		FOM		INT		SOM	
k	b	N		All models	Full model only	All models	Full model only	All models	Full model only	All models	Full model only	All models	Full model only	All models	Full model only
3	4	13	(3,3,3,4)	38.8204	37.6505	41.5291	41.3500	32.1731	32.5088	39.5544	37.6500	42.7989	41.3162	32.2210	32.5239
		14	(3,3,4,4)	40.3330	37.9658	43.8197	42.3023	32.9742	33.0313	40.1186	38.1870	43.6834	42.6792	33.1077	33.1405
		15	(3,4,4,4)	40.0763	38.7343	45.1438	43.0287	33.2981	33.7047	40.1973	38.7956	44.4618	43.1025	33.6177	33.7277
		16	(4,4,4,4)	40.2027	39.3495	46.5870	43.5209	33.7907	34.4044	40.2531	39.3799	46.6239	43.5423	33.8397	34.4316
		17	(4,4,4,5)	40.9673	38.3802	47.8787	43.1813	34.1343	34.6087	40.9785	39.4357	47.8910	43.0890	34.1510	34.7060
		18	(4,4,5,5)	40.8019	39.5144	46.5129	44.2007	34.7388	35.0960	41.2621	39.6185	47.8480	44.4039	34.7029	35.1483
		19	(4,5,5,5)	41.4162	40.4536	47.7441	45.6817	35.2346	35.3951	41.4666	40.5061	47.9739	45.9082	35.2816	35.4552
		20	(5,5,5,5)	40.8877	40.8478	47.5606	47.4701	35.5162	35.5175	40.8879	40.8516	47.5527	47.4480	35.5160	35.5301
		21	(5,5,5,6)	41.1485	41.1401	47.0077	46.9355	35.8498	35.8528	41.1447	41.1330	46.9919	46.9371	35.8597	35.8638
		22	(5,5,6,6)	41.1525	40.7412	48.2304	47.0885	35.3963	35.8294	41.1544	40.7389	48.3499	47.1024	35.4426	35.8340

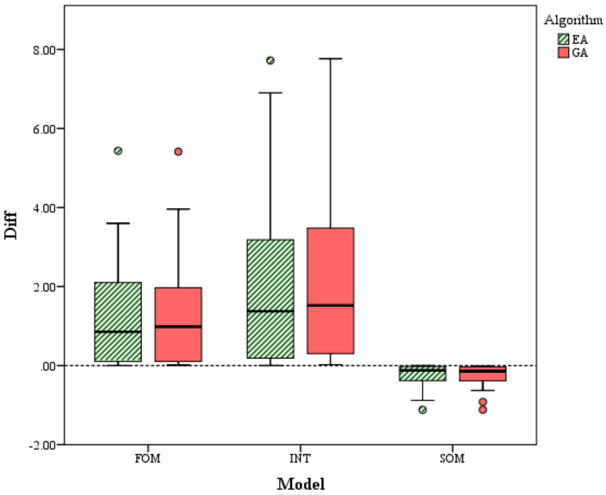

Figure 3. Boxplots for comparing differences in D_N-efficiencies for "all model" and "full model only" designs.

4. Conclusions

Our results show that GA designs are equally or more efficient and robust than EA designs for the D_w-optimality criterion with respect to all tested combinations of k variables and b blocks. The approximately optimal designs for the second-order model with blocks may be inefficient. We cannot pass over the uncertainty of possible reduced models prior to data collection; therefore, the researcher should consider using criteria that can create robust designs across the set of potential models. Our proposed D_w-optimality criterion can be a good alternative. It is not necessary to use the D_N-optimality criterion assuming a second-order model, because even if a second-order model is the correct model, its D_N-efficiency will be very close to the corresponding D_N-efficiency of the robust design generated using the D_w criterion.

References

Atkinson, A. C., Donev, A. N., & Tobias, R. D. (2007). *Optimum experimental designs with SAS*. Oxford, England: Clarendon.

Borkowski, J. J. (2003). Using a genetic algorithm to generate small exact response surface designs. *Journal of Probability and Statistical Science, 1*(1), 69-92.

Borkowski, J. J., & Valeroso, E. S. (2001). Comparison of design optimality criteria of response surface designs in the hypercube. *Technometrics, 43*(4), 468-477.

Chairojwattana, A., Chaimongkol, S., & Borkowski, J. J. (2017). Using genetic algorithms to generate D_w and G_w-optimal response surface designs in the hypercube. *Thailand Statistician, July 2017, 15*(2), 157-166.

Chipman, H. A. (1996). Bayesian variable selection with related predictors. *The Canadian Journal of Statistics, 24*(1), 17-36.

Chomtee, B., & Borkowski, J. J. (2005). Weighted design optimality criteria for spherical response surface designs. *KMITL Science Journal Issue, Proceedings of the 2nd International Symposium on Mathematical, Statistical and Computer Sciences, 5*, 226-238.

Cook, R. D., & Nachtsheim, C. J. (1980). A comparison of algorithms for constructing exact d-optimal designs. *Technometrics, 22*(3), 315-324.

Doehlert, D. H. (1970). Uniform shell designs. *Journal of the Royal Statistical Society, Series C, 19*(3), 231-239.

Doehlert, D. H., & Klee, V. L. (1972). Experimental designs through level reduction of the d-dimensional cuboctahedron. *Discrete Mathematics, 2*(4), 309-334.

Fedorov, V. V. (1972). *Theory of optimal experiments*. New York, NY: Academic Press

Heredia-Langner, A., Carlyle, W. M., Montgomery, D. C., Borror, C. M., & Runger, G. C. (2003). Genetic algorithms for the construction of D-optimal designs. *Journal of Quality Technology, 35*(1), 28-46.

Holland, J. H. (1975). *Adaptation in natural and artificial system: An introductory analysis with applications to biology control, and artificial intelligence.* Oxford, England: University of Michigan Press.

Limmun, W., Borkowski, J. J., & Chomtee, B. (2018a). Weighted A-optimality criterion for generating robust mixture designs. *Computers and Industrial Engineering, 125*, 348-356. doi:10.1016/j.cie.2018.09.002

Limmun, W., Chomtee, B., & Borkowski, J.J. (2018b). The construction of a model-robust IV-optimal mixture designs using a genetic algorithm. *Mathematical and Computational Application, 23*(2), 25. doi:10.3390/mca23020025

Mahachaichanakul, S., & Srisuradetchai, P. (2019). Applying the median and genetic algorithm to construct *D*- and *G*-optimality robust designs against missing data. *Applied Science and Engineering Process, 12*(1), 3-13.

Mitchell, T. J. (1974a). An algorithm for construction of "D-optimal" experimental designs. *Technometrics, 16*(2), 203-210.

Mitchell, T. J. (1974b). Computer construction of "D-Optimal" for first-order designs. *Technometrics, 16*(2), 211-220.

Myers, R. H., Montgomery, D. C., & Anderson-Cook, C. M. (2016). *Response surface methodology: Process and product optimization using designed experiments* (4th ed.). Hoboken, NJ: John Wiley and Sons.

Roquemore K. G. (1976). Hybrid designs for quadratic response surfaces. *Technometrics, 18*(4), 419-423

Sivanandam, S. N., & Deepa, S. N. (2008). *Introduction to genetic algorithms*. Berlin, Germany: Springer.

Thongsook, S., Borkowski, J. J., & Budsaba, K. (2014). Using a genetic algorithm to generate Ds-optimal designs with bounded D-efficiencies for mixture experiments. *Thailand Statistician*, *12*(2), 191-205.

Wald, A. (1943). On the efficient design of statistical investigations. *The Annals of Mathematical Statistics, 14*(2), 134-140.

Wynn, H. P. (1972). Results in the theory and construction of D-optimum experimental designs. *Journal of Royal Statistical Society, Series B, 34*(2), 133-147.