Int. J. Anal. Appl. (2024), 22:150

International Journal of Analysis and Applications

Finding Robust Response Surface Designs With Blocking Using a Model-Weighted A-Optimality Criterion

Peang-or Yeesa[1,*], Sudarat Nidsunkid[2]

[1] *Faculty of Science and Technology, Rajamangala University of Technology Srivijaya, Nakhon Si Thammarat, Thailand*

[2] *Department of Statistics, Faculty of Science, Kasetsart University, Bangkok, Thailand*

[*] *Corresponding author:* peangor.y@rmutsv.ac.th

Abstract. This paper proposes a new approach to finding robust response surface designs that can accommodate potential model misspecifications. To achieve this, experimental designs that are robust across all potential models were considered prior to data collection. Blocking effects were combined into all possible models, and the set of all reduced models was obtained using the weak heredity principle. The objective of this study was to propose the use of the geometric mean of A-optimalities as a new weighted A-optimality criterion for finding robust response surface designs. Both a genetic algorithm (GA) and an exchange algorithm (EA) were employed to optimize the weighted A-optimality criterion and compared with the widely used central composite design. The weighted A-optimal designs generated by GA and EA in this study had higher A_w and A-efficiencies than CCD, and the A_w-optimal designs generated by the GA were as or more efficient than the EA.

1. Introduction

Response surface designs are a crucial type of experimental designs used in the development, improvement, and optimization of industrial processes. Response surface methodology (RSM) combines statistical and mathematical techniques for three main objectives: fitting a response surface model on a specific region of interest, determining response optimization, and selecting operating conditions to reach specific requirements or customer needs. RSM focuses on approximating complex unknown functions with a lower-order polynomial, such as a first-order, an interaction, or a second-order model.

In cases where it is impractical to collect data for all factor level combinations under identical conditions, forming blocks is recommended to reduce variability. A common approach is to

Received: Jul. 16, 2024.

2020 *Mathematics Subject Classification.* 62K05, 62K10, 62K20.

Key words and phrases. response surface designs; weighted A-optimality; genetic algorithm; geometric mean.

https://doi.org/10.28924/2291-8639-22-2024-150

ISSN: 2291-8639

construct a small exact response surface design by assuming a second-order model. The model for k design variables and b blocks can be represented by equation (1.1),

$$y = \beta_0 + \sum_{i=1}^{k} \beta_i x_i + \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} \beta_{ij} x_i x_j + \sum_{i=1}^{k} \beta_{ii} x_i^2 + \sum_{l=1}^{b-1} \delta_l + \epsilon. \tag{1.1}$$

In this context, the k design variables are $x_1, x_2, ..., x_k$, y represent the observed response, while β's denotes the parameter coefficients to be estimated. δ_l stands for the l^{th} block effect, and ϵ is a normally distributed error term with an expected value of zero and variance σ^2.

When considering a model, there are various design choices available. Selecting an appropriate design is crucial, and different design criteria can be used for this purpose. Design optimality criteria primarily focus on the optimal characteristics of the $\mathbf{X}^{\mathbf{T}}\mathbf{X}$ matrix, where $\mathbf{X}$ is the model matrix [2]. Alphabetic optimality criteria are single-value measures that assess various desirable variance properties. These criteria serve to compare designs and construct optimal ones [1]. Four commonly used alphabetic design evaluation criteria include D, A, G, and IV optimality criteria. D and A optimality criteria emphasize parameter estimation, whilst G and IV optimality criteria focus on prediction variance. The use of the information matrix underscores the importance assumption regarding the adequacy of the empirical model in design evaluation.

Optimality criteria (D, A, G, and IV) are typically derived from a single model. However, it's common that the original design might not be efficient for the actual model used after data collection. Therefore, seeking a design that performs well across a set of possible reduced models becomes essential. This concept is known as model robustness. Therefore, this paper aims to find a response surface design that works well across the set of reduced models by evaluating it with a good optimality criterion.

Numerous strategies have emerged for evaluating a set of potential reduced models. Chipman [6] introduced two classes of reduced models based on weak heredity (WH) and strong heredity (SH) principles. A model can be represented by vector Δ, where '1'denotes the inclusion of a term and '0'indicates its absence. Notations Δ_i, Δ_{ij}, and Δ_{ii} signify the indicator function values of the i^{th} first-order effect, the ij^{th} interaction effect, and the ii^{th} second-order effect, respectively.

Under weak heredity (WH), the presence of the $\beta_{ij}x_i x_j$ term necessitates the inclusion of either the $\beta_i x_i$ or $\beta_j x_j$ term (or both) in the model. Similarly, if the $\beta_{ii}x_i^2$ term is included, the $\beta_i x_i$ term must also be present. For $k = 2$, the second-order model (without blocks) comprises 6 parameters (β_0, $\beta_1 x_1$, $\beta_2 x_2$, $\beta_{12} x_1 x_2$, $\beta_{11} x_1^2$, $\beta_{22} x_2^2$), yielding 17 WH reduced models corresponding to vectors $\Delta =$ (Δ_0, Δ_1, Δ_2, Δ_{12}, Δ_{11}, Δ_{22}). For $k = 3$, there exist 185 reduced models, with the second-order model featuring 10 parameters (without blocks).

Borkowski, Turk, and Chomtee [3] introduced weighted D-, A-, G-, and IV- efficiencies for response surface designs. These efficiencies are derived by assigning prior probabilities to various potential models, following model heredity principles. Their findings indicate that design optimality criteria may exhibit sensitivity to deviations from the full second-order model using the arithmetic mean. Chairojwattana, Chaimongkol, and Borkowski [5] devised a genetic algorithm to generate designs optimizing the weighted D- and G-optimality criteria for second-order response surface designs. They computed the weighted average of efficiency values across all models using the arithmetic mean, with weights determined by prior probability assignments to model effects. Limmun, Borkowski, and Chomtee [10] proposed a weighted A-optimality criterion for mixture designs, while another Limmun, Chomtee, and Borkowski [11] developed a weighted IV-optimal criterion for mixture designs, both using the arithmetic mean as the criterion.

Yeesa, Srisuradetchai, and Borkowski [21] introduced the weighted G-optimality criterion in 2019, which optimizes designs for two- and three-variable hypercube designs with blocks, utilizing the geometric mean as a criterion. Their resulting designs exhibit higher G-efficiencies compared to traditional G-optimal designs when the true model comprises first-order or interaction models. In 2020, Yeesa, Srisuradetchai, and Borkowski [22] proposed a weighted D-optimality criterion derived from all possible models to formulate robust designs for response surface designs with blocking factors, also employing the geometric mean as a criterion. The results indicate that the weighted D-optimality criterion offers another viable option for researchers. Notably, it shows that using the D-optimality criterion for second-order models, when the true model is indeed second-order, is unnecessary, as the corresponding D-efficiencies do not significantly deviate from those of robust designs obtained from the weighted D-optimality criterion. Limmun, Chomtee, and Borkowski [12] propose the weighted optimality criterion, utilizing the geometric mean to compute robust mixture designs within irregularly shaped polyhedral regions, as dictated by constraints on mixture component proportions. The results show that the proposed designs, based on G- and/or IV-efficiency, are robust to model misspecification.

In this study, we utilize A-optimality to formulate a weighted criterion aimed at generating designs resilient to various potential models. Specifically, we employ the weighted A-optimality criterion to assess designs. The objective of the weighted A-optimality criterion is to maximize the weighted A-efficiencies, denoted as A_w, within the design space across a set of reduced models. These weights are determined by experimenters, with one common approach being their assignment based on the parameters of each model. In cases where experimental runs cannot be conducted under identical conditions, it becomes necessary to establish blocks. When observations can be grouped into blocks of homogeneous units, the choice of blocking scheme is contingent upon the experiment's nature. Blocks introduce additional parameters into the model, considered as nuisance parameters. However, effective blocking of experimental designs can yield favorable features in experimental runs.

Consequently, the novelty of our research lies in the introduction of weighted A-optimality across the set of reduced models in experiments involving a blocking factor, using the geometric mean as a criterion. Furthermore, we implement both a genetic algorithm and an exchange algorithm to generate optimal designs, which are then compared to central composite designs.

2. Materials and Methods

2.1. **Weighted A-optimality Criterion.** In this paper, our focus is on the A-optimality criterion, which revolves around minimizing the individual variances of the estimated model coefficients. Its aim is to minimize the total (or mean) of the variances of the estimated coefficients, achieved by minimizing the sum of the diagonal elements of $(\mathbf{X^T X})^{-1}$. This, in turn, enhances precision in estimating model parameters. Accordingly, a design is considered A-optimal when it minimizes the trace of $(\mathbf{X^T X})^{-1}$.

The measure commonly used to assess the efficiency of any proposed design and its corresponding model based on A-optimality is referred to as "A-efficiency". The A-efficiency of an N-point design is determined and denoted as

A-efficiency = $\frac{100p}{trace\left[N\left(\mathbf{X^T X}\right)^{-1}\right]}$, where p is the number of model parameters.

Let the initial or 'full'model be the second-order model with blocks as specified in equation (1.1). Given these conditions, models with larger parameter counts are allotted greater weights. Here, M represents the count of reduced models derived from a given full model. A set of model weights $\{w_1, w_2, ..., w_M\}$ is defined, where w_i represents the weight assigned to the i^{th} model, and $\sum_{i=1}^{M} w_i = 1$.

The weight assigned to the i^{th} reduced model is calculated as $w_i = \frac{p(i)}{D \times m(p(i))}$, where $p(i)$ denotes the number of parameters excluding blocks in model i, $m(p(i))$ represents the count of models with $p(i)$ parameters, and $D = \sum_{p=1}^{\binom{k+2}{2}} p$. These weights are utilized in computing the weighted A-optimality criterion (A_w). For $k = 2$, the second-order model (excluding blocks) consists of 6 parameters with $D = \sum_{p=1}^{6} p = 21$. For $k = 3$, the second-order model (excluding blocks) contains 10 parameters with $D = \sum_{p=1}^{10} p = 55$.

The study generated robust designs in hypercube for 10 to 21 design points with 2 and 3 design variables and 2 and 3 blocks, respectively. The proposed scheme for weighting the criteria was based on giving more weight to a model with a larger number of parameters.

In this article, the geometric mean is utilized to calculate the A_w-optimality criterion. Consider Ξ as the set comprising all potential exact designs within the design space χ. The A_w-optimality criterion aims to seek a design ξ^* satisfying:

$$\xi^* = arg \min_{\xi \in \Xi} \left(\prod_{i=1}^{M} \left[trace[\mathbf{M_i}^{-1}(\xi)] \right]^{w_i} \right), \tag{2.1}$$

where $\mathbf{M_i}(\xi) = \mathbf{X}_{(\mathbf{i})}^{\mathbf{T}} \mathbf{X}_{(\mathbf{i})} \big/ N$ is a moment matrix for model i and N is the design size, $\mathbf{X}_{(\mathbf{i})}$ is the model matrix. Consequently, the corresponding weighted A-efficiency is defined as:

$$A_w = \prod_{i=1}^{M} A_i^{w_i}, \tag{2.2}$$

where $A_i = \frac{100p}{trace\left[N\left(\mathbf{X}_{(\mathbf{i})}^{\mathbf{T}} \mathbf{X}_{(\mathbf{i})}\right)^{-1}\right]}$, which A_i is termed the A-efficiency of the i^{th} reduced model.

The consideration of the geometric mean arises from the necessity for the design to exhibit robustness to model reduction and accommodate all parameters for reduced models. Conversely, the weighted optimality criterion relying on the arithmetic mean fails to ensure the accommodation of all reduced models. This misalignment contradicts the overarching aim of identifying a model-robust design.

2.2. **Genetic Algorithms.** A genetic algorithm (GA), a computational optimization method inspired by Darwinian Evolutionary Theory, particularly survival of the fittest, was first described by Holland in 1975 [9]. It operates by iteratively refining a population of potential solutions towards an optimal state by mimicking natural selection processes. This is achieved through genetic operators including selection, crossover, mutation, and inversion, which drive the evolution of the population towards better solutions based on an objective function evaluating each individual's fitness. The effectiveness of a GA is influenced by several factors such as the choice of crossover operator, crossover and mutation rates, and the size of the population. While early GAs utilized binary encoding, the adoption of real-value encoding has proven advantageous in many applications due to its simplicity and computational efficiency. Real-value encoding is particularly suited for optimizing multi-parameter problems, as it enables the representation of continuous variables. Although numerous methodologies have been employed to find optimal designs, the GA has emerged as a prominent choice in contemporary times. GAs exhibit a capacity to yield designs approaching optimality, particularly when utilizing real-number encoding and appropriately defined genetic operators. Consequently, a GA is capable of exploring the entirety of a continuous design space to generate designs that are notably close to optimal.

GAs have been utilized in design optimization. For example, Shahraki and Noorossana [19] proposed a combined genetic algorithm and reliability analysis enhanced by the design of experiments for parameter optimization, demonstrating the method's effectiveness with a numerical example. Mahachaichanakul and Srisuradetchai [13] employed GAs to develop robust response surface designs against missing data. Similarly, Yeesa, Srisuradetchai, and Borkowski [21] used GAs to produce optimal response surface designs with blocking factors, applying the weighted G-optimality criterion, and extended this approach in 2020 [22] to designs based on the weighted D-optimality criterion. Comparative research in 2020 showed that GAs yield designs superior to another algorithm. Additionally, Limmun, Chomtee, and Borkowski [12] proposed GA-based designs using weighted optimality criteria to compute robust mixture designs within irregularly shaped polyhedral regions.

The GA generates a precise N-point, k-variable response surface design, incorporating various blocking structures. Each chromosome represents an $N \times k$ matrix, detailing the N design points across k factors. The aim is to identify an $N \times k$ matrix that optimizes a design optimality criterion. In this context, a gene constitutes a row within the chromosome (design), while a genetic variable can pertain to any design variable within a gene (or row). Denoting the j^{th} genetic design variable in the i^{th} row of a chromosome as x_{ij}, the k-dimensional hypercube design region $[-1,1]^k$ gives directions for the potential values for each $x_{ij} \in [-1,1]^k$. An objective function, denoted as F, serves as a measure of a chromosome's fitness, reflecting the quality of the solution we aim to optimize. F takes a chromosome as input to create the objective function value as output, where higher values indicate superior fitness. The approach for generating designs using a GA is outlined below:

Step 1: Initiation Process

At the start of each generation, a population denoted by S chromosomes is established, with S being an odd number. Begin by randomly generating S chromosomes to represent the population of design matrices for a hypercube design region. Subsequently, calculate the objective function for each chromosome.

Step 2: Selection Process

Following the generation of the initial population of S chromosomes, the best chromosome is identified. This "**elite chromosome**" is distinguished by having the highest objective function F value and influences the subsequent generation of chromosomes. To produce the next generation of offspring chromosomes, select $(S-1)/2$ pairs randomly from the remaining $S-1$ non-elite chromosomes (referred to as parent chromosomes) prior to the reproduction process.

Step 3: Reproduction Process

Reproduction induces evolutionary changes in certain characteristics of the chromosomes, giving rise to the next generation. Upon completion of the reproduction process, we obtain $S-1$ offspring chromosomes, which are derived from the $S-1$ parent chromosomes. If the best offspring chromosomes exhibit a higher objective function value than the elite chromosome, the offspring chromosome with the highest objective function value replaces the elite chromosome as the new elite. Consequently, the elite chromosome and the $S-1$ offspring become the progenitors, ensuring the continuation of the next generation comprising S chromosomes. The reproduction process can be adjusted based on the researcher's preferences and the nature of the desired solutions. Nonetheless, it embodies the same principles as biological population genetics. Each reproduction operator undergoes a probability test on each row of P and Q, representing the two parent designs paired during reproduction. Let P_a denote the a^{th} row of P and Q_b denote the b^{th} row of Q. A reproduction operator is applied if it passes a probability test. A probability test is passed (PTIP) occurs when a randomly generated value, u, is less than or equal to a specified value, α_t, where u follows a uniform distribution [0,1], and the α_t values are predetermined by the experimenter. In this study, seven operators are applied to each parental pair in the reproduction process, following a predetermined order.

1) **Swap Rows (sr) Gene Operator:** If a PTIP occurs for row P_a of parent P, the operator exchanges P_a with a random row Q_b of parent Q. The range of α_{sr} values is 0.002 to 0.02.

2) **Swap Cut Point (scp) Gene Operator:** If a PTIP occurs for row P_a of parent P, the operator exchanges the last two decimal digits of the k genetic design variables of P_a with the last 2 decimal digits of the k genetic design variables for a random row Q_b of parent Q. The range of α_{scp} values is 0.005 to 0.02.

3) **Swap Block (sb) Gene Operator:** If a PTIP occurs for a row in block b (in either parent P or parent Q), the operator exchanges the row in block b with a random row from another block. The remaining operators are applied to the genetic variables in the rows of either parent P or parent Q. The range of α_{sb} values is 0.002 to 0.02.

4) **Swap Coordinates (sc) Gene Operator:** If a PTIP occurs for x_{ij} of parent P, the operator exchanges x_{ij} of parent P with a random x_{kl} of parent Q. The range of α_{sc} values is 0.002 to 0.02.

5) **Zero (z) Gene Operator:** If a PTIP occurs for x_{ij}, then x_{ij} is changed to 0. The range of α_z values is 0.01 to 0.05.

6) **Extreme (e) Gene Operator:** If a PTIP occurs for x_{ij}, x_{ij} is randomly set to either 1 or -1. The range of α_e values is 0.01 to 0.10.

7) **Creep (c) Operator:** If a PTIP occurs for x_{ij}, a random variate from $N(0, \sigma^2)$ is added to x_{ij} to create a new value x^*_{ij}. The variance σ^2 is determined by the researchers. The aim is to gradually change the value in each generation. If $x^*_{ij} > 1$ or $x^*_{ij} < -1$, it will be set to 1 or -1, respectively. The range of α_c values is 0.025 to 0.10.

Step 4: Convergence Check

The GA will stop if the objective function for the best chromosome in the new generation remains unchanged over numerous generations, indicating that further improvement is unlikely to be achieved.

2.3. **Exchange Algorithms.** The Exchange Algorithm (EA) operates by selecting points from a candidate set to construct a design matrix $\mathbf{X}^*$ that optimizes an optimality criterion. Initially, one or more points are exchanged between a randomly generated starting design and points from the candidate set. This iterative exchange process continues until no further enhancement in the optimality criterion value is achieved, indicating the discovery of the best possible design (Meyer and Nachtsheim, [14]). Variations of EAs have been developed by several researchers, including Fedorov [8], Wynn [20] , Mitchell [15] and [16], and Cook and Nachtsheim [7]. A notable modification by Mitchell [15] involves a more flexible method that modifies the basic single-point exchange algorithm to allow the replacement of multiple points in the original design during each iteration.

The procedure for generating designs to optimize the Aw-optimality criterion via an EA proceeds as follows:

Step 1: Begin by specifying the number of design variables k and the number of blocks b for N design points. Create a candidate set C comprising N_c points, and then randomly generate a starting design matrix of size $N \times k$. In this study, $N_c = 21^k$ for $k = 2$ and $k = 3$. That is, for $k = 2$, select $21^2 = 441$ candidate design points for each (x_1, x_2) with $C = \{-1, -0.9, \ldots, 0.9, 1\} \times \{-1, -0.9, \ldots, 0.9, 1\}$. For $k = 3$, select $21^3 = 9,261$ candidate design points.

Step 2: Replace a point in the starting design with a point from C and compute the corresponding A_w value. Repeat this process for all $N \times N_c$ exchanges. Retain the exchange and the design that produce the largest A_w value. This design becomes the new best design.

Step 3: Continue iterating Step 2 until no further improvement is observed in the A_w value.
Step 4: Iterate Steps 1 to 3 for 20 starting designs. Keep the best design obtained from these 20 starting designs.

2.4. **Central Composite Design.** Central Composite Design (CCD) is a widely used experimental design in response surface methodology (RSM) that facilitates the construction of second-order models without requiring exhaustive experimentation of all factor combinations. This design is instrumental in optimizing processes with multiple input variables. The CCD includes factorial points (n_f), axial points (n_a), and center points (n_c), totaling $N = n_f + n_a + n_c$ design points to estimate the parameters in the second-order model. The region of interest for a CCD depends on the selection of the axial point distance, typically ranging from ±1.0 to $\pm\sqrt{k}$. In this study, a cuboidal design region is considered, with a face-centered cube design having the axial points coded as $(\pm 1, 0, 0, ..., 0)$, $(0, \pm 1, 0, ..., 0)$, . . . , $(0, 0, ..., \pm 1)$.

To manage uncontrolled variability and improve experimental precision, blocking can be an essential addition to CCD. The general approach to blocking in CCD is to use two blocks, with the factorial points in one block and the axial points in the other block. If there are no center points, one of the pure quadratic effects becomes non-estimable. This issue can be addressed by adding center points, which allow all the quadratic effects to be estimable. For three blocks, the factorial points are divided between the first two blocks, and the axial points are in the last block. For further details and review, see Montgomery [17], which provides comprehensive coverage of experimental design techniques, including CCD and blocking. Box, Hunter, and Hunter [4] explain the principles of experimental design and the application of CCD with blocking. Myers, Montgomery, and Anderson-Cook [18] offer in-depth insights into response surface methodology and the practical aspects of CCD and blocking.

3. Results and Discussion

In this research, computer-generated designs using GA and EA are compared to CCDs. The results for A_w- and A-efficiencies are presented in Tables 1 to 4. For GA-generated and EA-generated designs, the A_w and A columns represent the A_w- and A-efficiencies of GA or EA designs that maximize A_w-efficiency for all WH reduced models. The A_{full} columns represent the A-efficiencies of GA or EA designs that maximize A-efficiency only for the full second-order model with blocks, as described in Equation (1.1). For CCDs, the A_w-efficiency is calculated by weighting the efficiencies of all reduced models, and the A-efficiency is calculated from the full second-order model with blocks. This context includes cases where the sample size in each block is feasible for a CCD.

For $k = 2$ variables and $b = 2$ blocks, n_1 and n_2 represent the sample sizes in the first and second blocks, respectively. For a CCD, the first and second blocks are the factorial block and the axial block. The varying number of center points in the blocks of the CCD results in different efficiency values. The maximum A_w-efficiency values of a CCD with blocks occur when $N = 12$, with 6 design points in each block, the factorial block contains 4 factorial points and 2 center points, while the axial block contains 4 axial points and 2 center points. The corresponding A_w-efficiency is 31.7871 and A-efficiency is 28.9256.

A comparison of A_w-efficiencies between GA designs, EA designs, and CCDs revealed that A_w-efficiencies of GA and EA designs are always greater than those of CCDs. For example, for $N = 16$, with sample sizes of 8 and 8 in the two blocks, the A_w-efficiencies of the GA design and EA design are 32.8087 and 32.7045, respectively, which are significantly higher than the CCD's A_w-efficiency of 28.8825. Comparing GA with EA, GA was able to find designs with higher A_w values than EA in almost all cases, except for three cases where the A_w values were equal: $N = 11(5,6)$, $N = 11(7,4)$, and $N = 15(7,8)$.

Regarding A-efficiencies, comparisons are made between five designs: GA designs for "all models," GA designs for "full model only," EA designs for "all models," EA designs for "full model only," and CCDs. The A-efficiencies of CCDs are less than those of GA and EA designs in all cases. Comparing GA with EA, the A-efficiencies values for "full model only" (A_{full}) of GA designs are always greater than those of EA designs, except where $N = 11$. In two cases presented, GA and EA designs for "all models" and "full model only" reach the same design, meaning the A-efficiency values for "all models" and "full model only" generated by GA and EA are the same.

For $k = 2$ variables and $b = 3$ blocks, n_1, n_2, and n_3 represent the number of design points in the first, second, and third blocks, respectively. The A_w- and A-efficiencies for GA and EA designs are always greater than those of CCDs. Comparing GA with EA, the A_w- and A-efficiencies of GA designs are always greater than those of EA designs.

For $k = 3$ variables and $b = 2$ blocks, in the same format as $k = 2$, the A_w-and A-efficiencies for GA and EA designs are always greater than those of CCDs. Comparisons show that GA and EA designs are more efficient than CCDs when the factorial block has no center point ($n_1 = 8$). Comparing GA with EA, the A_w-and A-efficiencies of GA designs are always greater than those of EA designs, except where $N = 20$, with block sizes of 10 and 10, where both GA and EA reach the same design.

For $k = 3$ variables and $b = 3$ blocks, the A_w-and A-efficiencies for GA and EA designs are always greater than those of CCDs. In the comparison of A_w-efficiencies between GA and EA designs, it was revealed that A_w-efficiencies of GA designs are always greater than those of EA designs. For A-efficiencies, when comparing GA designs and EA designs for "all models," it was found that GA designs generally have higher A-efficiencies than EA designs, except where $N = 18(5,5,8)$ and $N = 20(5,6,9)$, where A-efficiencies of EA designs are greater. This is because, for designs generated for "all models," the criterion used to find the best design is the weighted A-optimal criterion, meaning designs with the highest A_w values might have lower A-efficiency than designs with lower A_w values. For designs generated for "full model only," GA designs are greater than EA designs in all cases.

Table 1: Summary of A_w, A, and A_{full}-efficiencies from GA designs, EA designs, and CCDs with $k = 2$ variables and $b = 2$ blocks

N	n_1, n_2	GA all models		GA full model only	EA all models		EA full model only	CCD	
		A_w	A	A_{full}	A_w	A	A_{full}	A_w	A
10	(5, 5)	32.1315	29.2955	29.3815	32.1131	29.3127	29.3555	31.5871	27.9627
11	(5, 6)	32.3984	29.9210	29.9210	32.3984	29.9210	29.9210	30.7960	27.1019
11	(7, 4)	33.0968	30.2430	30.2430	33.0968	30.2430	30.2430	30.9918	29.3706
12	(6, 6)	32.6587	29.5161	29.5532	32.6507	29.4633	29.5444	31.7871	28.9256
13	(6, 7)	31.9823	29.1326	29.5351	31.9208	29.1585	29.5178	30.7917	27.6390
13	(7, 6)	33.1517	29.4994	29.7768	33.1328	29.4247	29.7629	31.2133	28.5556
14	(7, 7)	32.7340	28.9596	29.5054	32.6679	28.9097	29.4762	30.4709	27.4510
15	(7, 8)	32.5652	28.7465	29.3930	32.5652	28.7465	29.3630	29.5399	26.2632
15	(8, 7)	32.8358	29.2078	29.7679	32.8163	29.1838	29.7330	29.6159	26.6294
16	(8, 8)	32.8087	29.1265	29.7868	32.7045	29.7378	29.7655	28.8825	25.6098

Table 2: Summary of A_w, A, and A_{full}-efficiencies from GA designs, EA designs, and CCDs with $k = 2$ variables and $b = 3$ blocks

N	n_1, n_2, n_3	GA all models		GA full model only	EA all models		EA full model only	CCD	
		A_w	A	A_{full}	A_w	A	A_{full}	A_w	A
12	(3, 4, 5)	21.7519	22.1024	22.1363	21.7402	22.0541	22.1175	17.1182	17.0213
12	(4, 4, 4)	24.0143	23.9541	23.9573	24.0061	23.9266	23.9521	20.9577	21.0526
13	(4, 4, 5)	23.4888	23.4902	23.6925	23.4834	23.4610	23.6868	19.9085	19.9339
13	(5, 4, 4)	24.5612	24.4675	24.4862	24.5485	24.4564	24.4787	21.8450	21.7056
14	(4, 5, 5)	23.1628	23.4723	23.4889	23.1443	23.4350	23.4686	19.4109	19.3278
14	(5, 5, 4)	24.2516	23.9448	24.1331	24.2190	23.8794	24.1293	21.2325	21.4446
15	(4, 5, 6)	22.8404	23.2460	23.2614	22.7436	23.1151	23.2489	18.4498	18.3055
15	(5, 5, 5)	24.0663	23.9767	24.0356	24.0537	23.9639	24.0239	20.6974	20.3876
16	(5, 5, 6)	23.8691	23.9173	23.9350	23.8599	23.8895	23.9069	19.8215	19.4411
16	(6, 5, 5)	24.7984	24.6829	24.7064	24.7868	24.6771	24.6868	21.1478	20.5613

Table 3: Summary of A_w, A, and A_{full}-efficiencies from GA designs, EA designs, and CCDs with $k = 3$ variables and $b = 2$ blocks

N	n_1, n_2	GA all models		GA full model only	EA all models		EA full model only	CCD	
		A_w	A	A_{full}	A_w	A	A_{full}	A_w	A
15	(8, 7)	33.4864	29.6429	29.6611	33.4264	29.6347	29.6347	13.3539	4.7210
16	(8, 8)	33.2353	29.5396	29.5678	33.1618	29.3961	29.5228	15.1019	7.0831
17	(8, 9)	33.1723	29.9770	30.0625	33.0568	29.4628	29.7903	15.7369	8.3500
17	(9, 8)	33.5636	30.2369	30.2628	33.5285	30.2204	30.2204	22.8904	18.6298
18	(9, 9)	33.6356	30.3831	30.2888	33.5367	30.4067	30.2513	21.9673	17.8716
18	(10, 8)	34.0503	30.9246	30.9287	34.0258	30.9155	30.9155	26.4062	22.4494
19	(9, 10)	33.3103	30.4309	30.4815	33.2639	30.4323	30.4364	21.0702	17.1280
19	(10, 9)	33.9885	31.0876	31.0915	33.9740	31.0709	31.0825	25.3153	21.4151
20	(9, 11)	33.1536	30.2144	30.8286	32.9851	29.8188	30.6743	20.2152	16.4172
20	(10, 10)	33.9332	31.4286	31.4286	33.9332	31.4286	31.4286	24.2766	20.4515

Table 4: Summary of A_w, A, and A_{full}-efficiencies from GA designs, EA designs, and CCDs with $k = 3$ variables and $b = 3$ blocks

N	n_1, n_2, n_3	GA all models		GA full model only	EA all models		EA full model only	CCD	
		A_w	A	A_{full}	A_w	A	A_{full}	A_w	A
17	(4, 5, 8)	23.6658	24.0893	24.0997	23.5296	23.9061	23.9296	16.6702	14.8526
17	(5, 5, 7)	25.1583	24.9904	25.1142	24.8136	24.4061	24.8713	22.1866	21.0835
18	(5, 5, 8)	24.8541	24.7944	24.9388	24.7182	24.8517	24.8995	23.9476	22.2953
18	(6, 6, 6)	26.2173	25.7226	25.8568	26.1728	25.6996	25.7799	24.7563	23.4604
19	(5, 5, 9)	24.4083	24.5774	24.6824	24.2882	24.4902	24.5605	20.2037	19.1187
19	(6, 6, 7)	26.0094	25.7303	25.8985	25.8870	25.5616	25.6891	23.7665	22.4069
20	(5, 6, 9)	24.5030	24.8335	24.9214	24.4523	24.8520	24.8719	20.3075	19.2601
20	(6, 6, 8)	25.8170	25.8315	25.8913	25.6811	25.7078	25.7668	22.7991	21.4098
21	(6, 6, 9)	25.6054	25.7126	25.7575	25.4254	25.6556	25.7486	21.8760	20.4785
21	(7, 7, 7)	26.5984	26.5065	26.5065	26.5344	26.3531	26.3637	23.8124	21.9805

Based on the results of the comparison between GA designs, EA designs, and CCDs, GA consistently identified designs with the highest A_w-efficiency. Consequently, this research presents the design points obtained through GA in Tables 5 to 8. These tables show the design points identified by GA using the A_w-optimality criterion, which can be very useful for practitioners looking to implement experimental designs for data collection. For example, in Table 5 for $k = 2$,

$b = 2$, $N = 10$, and the block sizes are 5 and 5, the corresponding A_w-efficiencies of the robust design ("all models" design) are equal to 32.1315. The design points in the 1^{st} block are denoted by superscript 1, i.e., $(1,1)^1$, $(-0.13,-0.14)^1$, $(-1,1)^1$, $(1,-0.09)^1$, and $(-0.12,-1)^1$, while design points in the 2^{nd} block are given as superscript 2, i.e., $(0.13,1)^2$, $(1,-1)^2$, $(-1,-1)^2$, $(0.1,0.11)^2$, and $(-1,0.06)^2$. In Table 6, for $k = 2$, $b = 3$, $N = 12$, and the block sizes are 3, 4, and 5, the corresponding A_w-efficiencies are equal to 21.7519. The design points in the 1^{st} block are denoted by superscript 1, design points in the 2^{nd} block are given as superscript 2, and design points in the 3^{rd} block are given as superscript 3.

Table 5: Design generated by Genetic Algorithm for A_w-optimality criterion with $k = 2$ variables and $b = 2$ blocks

N	n_i	A_w	Design Points
10	(5, 5)	32.1315	$(1,1)^1$, $(-0.13,-0.14)^1$, $(-1,1)^1$, $(1,-0.09)^1$, $(-0.12,-1)^1$, $(0.13,1)^2$, $(1,-1)^2$, $(-1,-1)^2$, $(0.1,0.11)^2$, $(-1,0.06)^2$
11	(5, 6)	32.3984	$(0,1)^1$, $(0,0)^1$, $(0,-1)^1$, $(1,0)^1$, $(-1,0)^1$, $(0,0)^2$, $(0,0)^2$, $(1,-1)^2$, $(1,1)^2$, $(-1,-1)^2$, $(-1,1)^2$
11	(7, 4)	33.0968	$(0,0)^1$, $(-1,1)^1$, $(0,0)^1$, $(1,-1)^1$, $(0,0)^1$, $(-1,-1)^1$, $(1,1)^1$, $(0,-1)^2$, $(1,0)^2$, $(-1,0)^2$, $(0,1)^2$
12	(6, 6)	32.6587	$(-1,0.15)^1$, $(0,1)^1$, $(-1,-1)^1$, $(1,0)^1$, $(0.17,-1)^1$, $(0.04,0.05)^1$, $(-0.08,-0.06)^2$, $(-1,-1)^2$, $(-0.08,-0.1)^2$, $(-1,1)^2$, $(1,1)^2$, $(1,-1)^2$
13	(6, 7)	31.9823	$(-0.05,1)^1$, $(1,0)^1$, $(0.03,0.04)^1$, $(-1,0.26)^1$, $(0.12,-1)^1$, $(-1,-1)^1$, $(1,1)^2$, $(-1,1)^2$, $(-1,-0.06)^2$, $(1,-1)^2$, $(-0.03,-0.1)^2$, $(-1,-1)^2$, $(-0.03,-0.1)^2$
13	(7, 6)	33.1517	$(-0.01,0)^1$, $(0.06,-1)^1$, $(1,1)^1$, $(-1,1)^1$, $(-1,-1)^1$, $(1,-0.05)^1$, $(-0.01,0)^1$, $(1,-1)^2$, $(0.01,0)^2$, $(-1,-1)^2$, $(-0.06,1)^2$, $(-1,0.06)^2$, $(1,1)^2$
14	(7, 7)	32.7340	$(-0.05,0.05)^1$, $(-1,1)^1$, $(-0.05,0.05)^1$, $(1,1)^1$, $(-0.05,0.05)^1$, $(1,-1)^1$, $(-1,-1)^1$, $(-1,-1)^2$, $(1,1)^2$, $(-1,0)^2$, $(0.11,-1)^2$, $(-1,1)^2$, $(0,1)^2$, $(1,-0.11)^2$
15	(7, 8)	32.5652	$(0,0)^1$, $(0,0)^1$, $(1,-1)^1$, $(-1,-1)^1$, $(1,1)^1$, $(0,0)^1$, $(-1,1)^1$, $(1,1)^2$, $(0,1)^2$,$(-1,-1)^2$, $(1,-1)^2$, $(1,0)^2$, $(0,-1)^2$, $(-1,0)^2$, $(-1,1)^2$
15	(8, 7)	32.8358	$(-0.06,0)^1$, $(-0.06,0)^1$, $(-1,1)^1$, $(-1,-1)^1$, $(0.01,-1)^1$, $(1,-1)^1$, $(-0.06,0)^1$, $(1,1)^1$, $(0,-1)^2$, $(0.15,1)^2$, $(-1,-1)^2$, $(-1,1)^2$, $(-1,-0.05)^2$, $(1,0.05)^2$, $(1,-1)^2$
16	(8, 8)	32.8087	$(-0.04,0)^1$, $(1,0)^1$, $(1,1)^1$, $(-0.04,0)^1$, $(-1,-1)^1$, $(-0.04,0)^1$, $(1,-1)^1$, $(-1,1)^1$, $(0.04,-1)^2$, $(1,0)^2$, $(0.04,1)^2$, $(1,-1)^2$, $(-1,1)^2$, $(-1,-1)^2$, $(1,1)^2$, $(-1,0)^2$

Table 6: Design generated by Genetic Algorithm for A_w-optimality criterion with $k = 2$ variables and $b = 3$ blocks

N	n_i	A_w	Design Points
12	(3, 4, 5)	21.7519	$(-0.02,-1)^1$, $(1,0.36)^1$, $(-0.89,0.5)^1$, $(-1,-0.1)^2$, $(1,-1)^2$, $(-0.06,0)^2$, $(1,1)^2$, $(0.06,-0.07)^3$, $(1,-0.2)^3$, $(0.06,1)^3$, $(-1,-1)^3$, $(-1,1)^3$
12	(4, 4, 4)	24.0143	$(-1,0)^1$, $(0,-1)^1$, $(1,0)^1$, $(0,1)^1$, $(1,-1)^2$, $(0,1)^2$, $(-1,-1)^2$,$(0,0.06)^2$, $(0,-1)^3$, $(1,1)^3$, $(0,-0.06)^3$, $(-1,1)^3$
13	(4, 4, 5)	23.4888	$(1,-1)^1$, $(-1,-0.11)^1$, $(0.29,1)^1$, $(0.02,-0.05)^1$, $(0.11,-0.11)^2$, $(-1,1)^2$, $(-0.02,-1)^2$, $(1,1)^2$, $(-1,-1)^3$, $(1,0.06)^3$, $(1,-1)^3$, $(-0.02,0.05)^3$, $(-0.25,1)^3$
13	(5, 4, 4)	24.5612	$(-1,1)^1$, $(1,0)^1$, $(0.12,0)^1$, $(0.12,0)^1$, $(-1,-1)^1$, $(-1,0)^2$, $(-0.13,0)^2$,$(1,1)^2$, $(1,-1)^2$, $(-0.06,1)^3$, $(-1,0)^3$, $(-0.06,-1)^3$, $(1,0)^3$
14	(4, 5, 5)	23.1628	$(-0.06,1)^1$, $(-0.14,-1)^1$, $(1,-0.15)^1$, $(-1,0.05)^1$, $(0.06,0.16)^2$, $(-1,-1)^2$, $(1,-1)^2$, $(-1,1)^2$, $(0.06,0.15)^2$, $(-1,1)^3$, $(0.01,-0.05)^3$, $(1,1)^3$, $(0.2,-1)^3$, $(-1,-0.17)^3$
14	(5, 5, 4)	24.2516	$(-0.05,-1)^1$, $(-0.03,0.16)^1$, $(1,-1)^1$, $(-1,0.06)^1$, $(1,1)^1$, $(1,-0.1)^2$, $(0.05,-0.1)^2$, $(1,1)^2$, $(-1,1)^2$, $(-0.14,-1)^2$, $(-1,-1)^3$, $(0.27,0.05)^3$, $(-0.07,1)^3$, $(1,-1)^3$
15	(4, 5, 6)	22.8404	$(1,-0.1)^1$, $(-0.03,1)^1$, $(-0.17,-1)^1$, $(-1,0.05)^1$, $(0.15,-1)^2$, $(-1,1)^2$, $(0.08,0)^2$, $(1,1)^2$, $(-1,-0.21)^2$, $(-0.01,0.1)^3$, $(1,1)^3$, $(-0.01,0.1)^3$, $(1,-1)^3$, $(-1,-1)^3$, $(-1,1)^3$
15	(5, 5, 5)	24.0663	$(-1,-1)^1$, $(-0.01,0)^1$, $(-0.01,-0.05)^1$, $(1,-0.05)^1$, $(-0.79,1)^1$, $(-1,-1)^2$, $(1,-1)^2$, $(0.14,1)^2$, $(-1,0.1)^2$, $(-0.02,0.11)^2$, $(-1,-0.05)^3$, $(-0.02,-0.16)^3$, $(-1,1)^3$, $(1,1)^3$, $(0.07,-1)^3$
16	(5, 5, 6)	23.8691	$(-0.08,1)^1$, $(1,1)^1$, $(-0.06,-0.06)^1$, $(-1,-0.1)^1$, $(1,-1)^1$, $(-1,1)^2$, $(1,1)^2$, $(-0.08,-1)^2$, $(-0.01,-0.01)^2$, $(1,0)^2$, $(-1,-1)^3$, $(1,0.1)^3$, $(-1,1)^3$, $(0.06,0.05)^3$, $(0.14,1)^3$, $(1,-1)^3$
16	(6, 5, 5)	24.7984	$(1,-1)^1$, $(1,0.05)^1$, $(0.03,0.15)^1$, $(0.03,0.15)^1$, $(-1,-1)^1$, $(-0.12,1)^1$, $(1,1)^2$, $(-0.12,-1)^2$, $(0.04,-0.17)^2$, $(-1,1)^2$, $(1,-0.15)^2$, $(-1,0)^3$, $(1,-1)^3$, $(1,1)^3$, $(0.03,0.1)^3$, $(0,-1)^3$

Table 7: Design generated by Genetic Algorithm for A_w-optimality criterion with $k = 3$ variables and $b = 2$ blocks

N	n_i	A_w	Design Points
15	(8, 7)	33.4864	$(1,-1,-1)^1$, $(-1,-1,1)^1$, $(0.05,1,0.05)^1$, $(-1,1,-1)^1$, $(0.05,0.05,1)^1$, $(1,0.05,0)^1$, $(0.07,0.07,0.1)^1$, $(-1,-1,-1)^1$, $(-1,1,1)^2$, $(1,-1,1)^2$, $(-0.05,-0.05,-1)^2$, $(1,1,1)^2$, $(-1,-0.05,0)^2$, $(1,1,-1)^2$, $(-0.05,-1,-0.05)^2$
16	(8, 8)	33.2353	$(-1,-1,-1)^1$, $(-0.01,-1,0.05)^1$, $(-0.09,0.1,0.1)^1$, $(1,-1,1)^1$, $(-1,1,1)^1$, $(1,1,-1)^1$, $(-0.01,0.03,-1)^1$, $(1,0.09,0.1)^1$, $(0.03,-0.04,1)^2$, $(-1,-0.04,-0.05)^2$, $(0.03,1,-0.05)^2$, $(-1,1,-1)^2$, $(-1,-1,1)^2$, $(1,-0.09,-0.1)^2$, $(1,1,1)^2$, $(1,-1,-1)^2$
17	(8, 9)	33.1723	$(-1,0.01,0)^1$, $(1,1,-1)^1$, $(-1,1,1)^1$, $(-1,-1,-1)^1$, $(-0.02,-0.08,1)^1$, $(0.13,-0.1,0)^1$, $(1,-1,1)^1$, $(-0.01,1,-0.1)^1$, $(-1,1,-1)^2$, $(1,1,0.2)^2$, $(0.03,-1,0.1)^2$, $(-1,0.08,0.1)^2$, $(0.03,0.06,-1)^2$, $(1,0.2,1)^2$, $(0.05,1,1)^2$, $(1,-1,-1)^2$, $(-1,-1,1)^2$
17	(9, 8)	33.5636	$(0.07,-1,-0.05)^1$, $(-1,-1,-1)^1$, $(0.07,-0.02,-1)^1$, $(1,-1,1)^1$, $(0.03,-0.02,1)^1$, $(-1,-0.04,-0.05)^1$, $(-1,1,1)^1$, $(1,1,-1)^1$, $(0.03,1,0)^1$, $(-0.04,0.05,-1)^2$, $(-1,0.04,0.05)^2$, $(1,-1,-1)^2$, $(1,1,1)^2$, $(-1,1,-1)^2$, $(-1,-1,1)^2$, $(1,0.01,0)^2$, $(-0.04,-1,0.05)^2$
18	(9, 9)	33.6356	$(-0.07,-0.07,1)^1$, $(-1,-1,-1)^1$, $(-1,1,1)^1$, $(1,-0.07,-0.1)^1$, $(-0.07,1,-0.1)^1$, $(-0.07,-0.07,-0.2)^1$, $(1,-1,1)^1$, $(-0.08,-0.08,0)^1$, $(1,1,-1)^1$, $(0.19,1,1)^2$, $(0.08,0.08,-1)^2$, $(-1,0.08,0.05)^2$, $(0.08,-1,0.1)^2$, $(1,1,0.2)^2$, $(-1,-1,1)^2$, $(1,-1,-1)^2$, $(1,0.19,1)^2$, $(-1,1,-1)^2$
18	(10, 8)	34.0503	$(-0.01,0,-1)^1$, $(-1,0,0)^1$, $(-0.02,0,1)^1$, $(1,1,-1)^1$, $(-1,1,1)^1$, $(-1,-1,-1)^1$, $(1,0,0)^1$, $(-0.05,1,0.05)^1$, $(-0.05,-1,0.05)^1$, $(1,-1,1)^1$, $(-1,0,-0.05)^2$, $(1,-1,-1)^2$, $(-1,1,-1)^2$, $(0.06,-1,-0.05)^2$, $(0.04,0,1)^2$, $(0.06,1,-0.05)^2$, $(-1,-1,1)^2$, $(1,1,1)^2$
19	(9, 10)	33.3103	$(0.16,-0.16,0)^1$, $(0.05,-0.05,-1)^1$, $(1,1,-1)^1$, $(-1,1,1)^1$, $(1,-1,1)^1$, $(0.03,1,0)^1$, $(0.05,-0.05,1)^1$, $(-1,-0.03,0)^1$, $(-1,-1,-1)^1$, $(1,-1,-1)^2$, $(-1,1,-1)^2$, $(-1,-1,1)^2$, $(-0.03,-1,0)^2$, $(-1,0.04,0)^2$, $(1,0.03,0)^2$, $(-0.08,0.08,-1)^2$, $(1,1,1)^2$, $(-0.04,1,0)^2$, $(-0.08,0.08,1)^2$
19	(10, 9)	33.9885	$(0,-1,0)^1$, $(0,0.04,1)^1$, $(1,0.04,0)^1$, $(-1,-1,-1)^1$, $(0,1,0)^1$, $(1,1,-1)^1$, $(0,0.04,-1)^1$, $(-1,0.04,0)^1$, $(1,-1,1)^1$, $(-1,1,1)^1$, $(1,-1,-1)^2$, $(-1,1,-1)^2$, $(1,1,1)^2$, $(0,1,0)^2$, $(-1,-1,1)^2$, $(0,-0.06,1)^2$, $(-1,-0.06,0)^2$, $(0,-0.06,-1)^2$, $(1,-0.06,0)^2$
20	(9, 11)	33.1536	$(0,-0.07,0)^1$, $(-0.16,-0.11,1)^1$, $(1,-1,1)^1$, $(-1,-1,-1)^1$, $(-0.05,1,-0.1)^1$, $(0.04,0.08,-1)^1$, $(1,-0.04,-0.1)^1$, $(1,1,-0.1)^1$, $(-1,1,1)^1$, $(1,1,-1)^2$, $(1,0.03,1)^2$, $(-1,-1,1)^2$, $(1,-1,-1)^2$, $(0.02,1,1)^2$, $(-1,0.07,0.05)^2$, $(-1,0.07,0)^2$, $(1,1,1)^2$, $(0.03,0.01,-1)^2$, $(0.08,-1,0.05)^2$, $(-1,1,-1)^2$
20	(10, 10)	33.9332	$(0,-1,0)^1$, $(0,-1,0)^1$, $(1,1,1)^1$, $(-1,1,-1)^1$, $(0,0,1)^1$, $(1,-1,-1)^1$, $(1,1,-1)^1$, $(-1,0,0)^1$, $(0,0,1)^1$, $(-1,0,0)^1$, $(-1,1,1)^2$, $(0,1,0)^2$, $(1,0,0)^2$, $(0,1,0)^2$, $(-1,-1,1)^2$, $(0,0,-1)^2$, $(1,-1,1)^2$, $(-1,-1,-1)^2$, $(0,0,-1)^2$, $(1,0,0)^2$

Table 8: Design generated by Genetic Algorithm for A_w-optimality criterion with $k = 3$ variables and $b = 3$ blocks

N	n_i	A_w	Design Points
17	(4,5,8)	23.6658	$(-0.35,-0.04,-1)^1$, $(1,-1,0)^1$, $(-1,-0.23,1)^1$, $(0.21,1,0.05)^1$, $(1,-0.54,1)^2$, $(0,0.03,0)^2$, $(-1,1,1)^2$, $(1,1,-1)^2$, $(-1,-1,-1)^2$, $(0.02,0.1,0)^3$, $(1,1,1)^3$, $(-1,0.09,-0.05)^3$, $(-0.08,-1,1)^3$, $(-1,-1,0.2)^3$, $(1,-0.18,-1)^3$, $(0.3,-1,-1)^3$, $(-1,1,-1)^3$
17	(5,5,7)	25.1583	$(0.15,-0.12,-1)^1$, $(0.09,-1,0)^1$, $(1,1,0.3)^1$, $(-1,1,-1)^1$, $(-1,0.02,1)^1$, $(-1,-0.02,-0.3)^2$, $(1,1,-1)^2$, $(-0.12,1,1)^2$, $(1,-1,1)^2$, $(-0.07,-0.09,-0.1)^2$, $(-1,-1,-1)^3$, $(1,-1,-1)^3$, $(-0.06,1,-1)^3$, $(-1,-1,1)^3$, $(0.11,0.03,0)^3$, $(-1,1,0.3)^3$, $(1,0.12,1)^3$
18	(5,5,8)	24.8541	$(1,0.07,0)^1$, $(-1,1,-1)^1$, $(0.07,-1,-1)^1$, $(-0.04,0.07,1)^1$, $(-1,-1,0.2)^1$, $(-0.03,-0.04,0)^2$, $(1,-1,1)^2$, $(1,1,-1)^2$, $(-1,-1,-1)^2$, $(-1,1,1)^2$, $(1,-1,-1)^3$, $(1,1,1)^3$, $(-1,-0.02,-1)^3$, $(-1,-1,1)^3$, $(-0.07,1,0)^3$, $(-1,0.05,-0.2)^3$, $(0.04,-1,0)^3$, $(0.01,0.07,-1)^3$
18	(6,6,6)	26.2173	$(0.03,-1,-0.1)^1$, $(0.13,0.05,1)^1$, $(0.12,1,0.3)^1$, $(-1,-0.22,0.05)^1$, $(1,0.11,-1)^1$, $(-1,1,-1)^1$, $(-0.04,0.08,-1)^2$, $(0,0.01,0)^2$, $(1,1,-0.3)^2$, $(-1,1,1)^2$, $(1,-1,1)^2$, $(-1,-1,-1)^2$, $(-1,0.26,-0.2)^3$, $(-1,-1,1)^3$, $(-0.08,1,-1)^3$, $(1,-1,-1)^3$, $(0.02,-0.02,0)^3$, $(1,1,1)^3$
19	(5,5,9)	24.4083	$(1,1,-0.25)^1$, $(-0.15,0.18,1)^1$, $(0.02,-1,-1)^1$, $(-1,0.06,0)^1$, $(1,-1,1)^1$, $(-1,1,-1)^2$, $(-0.17,-1,1)^2$, $(1,1,1)^2$, $(-0.1,-0.1,0.05)^2$, $(1,-1,-1)^2$, $(1,0.14,-1)^3$, $(0.06,1,-1)^3$, $(0,-0.1,-0.1)^3$, $(-1,1,1)^3$, $(0,1,0.15)^3$, $(1,-0.04,1)^3$, $(1,-1,0)^3$, $(-1,-1,1)^3$, $(-1,-1,-1)^3$
19	(6,6,7)	26.0094	$(-0.23,0.03,0)^1$, $(0.03,-1,1)^1$, $(1,1,1)^1$, $(-1,-0.01,0)^1$, $(-0.02,1,-0.4)^1$, $(1,-1,-1)^1$, $(0.03,0.05,0)^2$, $(1,1,-1)^2$, $(-1,-1,-1)^2$, $(0.1,-1,0)^2$, $(-1,1,1)^2$, $(1,-0.24,1)^2$, $(-0.03,-0.13,-1)^3$, $(0.02,1,1)^3$, $(-1,-1,1)^3$, $(1,1,0.1)^3$, $(-1,1,-1)^3$, $(1,-0.03,-1)^3$, $(1,-1,0.1)^3$
20	(5,6,9)	24.5030	$(0.19,-1,-0.1)^1$, $(0.04,1,1)^1$, $(-1,0.27,-1)^1$, $(-1,-1,1)^1$, $(1,0.04,-0.15)^1$, $(-1,-1,-1)^2$, $(-0.05,-0.02,0.15)^2$, $(-1,1,1)^2$, $(1,-1,1)^2$, $(-0.05,-0.02,0.1)^2$, $(1,1,-1)^2$, $(-0.01,-0.12,-1)^3$, $(-0.01,-1,1)^3$, $(-1,-1,0)^3$, $(-1,1,-0.2)^3$, $(1,1,1)^3$, $(-1,-0.08,1)^3$, $(-0.1,1,-1)^3$, $(1,0.06,0.1)^3$, $(1,-1,-1)^3$
20	(6,6,8)	25.8170	$(-0.06,-0.06,-0.2)^1$, $(-0.05,-1,-0.2)^1$, $(-0.27,1,1)^1$, $(1,-0.29,1)^1$, $(1,1,-1)^1$, $(-1,-0.14,-0.05)^1$, $(-1,-1,1)^2$, $(0.06,0.07,0)^2$, $(0.07,0.07,0.05)^2$, $(1,1,1)^2$, $(-1,1,-1)^2$, $(1,-1,-1)^2$, $(-1,0.14,1)^3$, $(1,1,0)^3$, $(1,-0.02,-1)^3$, $(-1,1,0.2)^3$, $(1,-1,0.2)^3$, $(-0.03,0.21,-1)^3$, $(0.08,-1,1)^3$, $(-1,-1,-1)^3$
21	(6,6,9)	25.6054	$(1,-0.09,1)^1$, $(-0.1,1,1)^1$, $(-1,-0.12,0)^1$, $(-0.12,-1,0.05)^1$, $(1,-0.11,-1)^1$, $(-0.1,1,-1)^1$, $(-1,-1,1)^2$, $(1,1,1)^2$, $(0.09,0.09,0)^2$, $(1,-1,-1)^2$, $(-1,1,-1)^2$, $(0.09,0.09,0)^2$, $(-1,0.24,0.05)^3$, $(-1,1,1)^3$, $(-0.04,-0.04,1)^3$, $(-1,-1,-1)^3$, $(0.24,-1,0)^3$, $(1,1,-1)^3$, $(1,1,0)^3$, $(1,-1,1)^3$, $(-0.04,-0.03,-1)^3$
21	(7,7,7)	26.5984	$(-0.04,0.11,1)^1$, $(-0.08,-1,-1)^1$, $(1,-0.12,-0.05)^1$, $(-1,1,-1)^1$, $(-1,-1,1)^1$, $(0,1,0.2)^1$, $(1,-0.12,0)^1$, $(-1,-0.42,-1)^2$, $(-0.03,0.05,0)^2$, $(-0.05,-1,0)^2$, $(-1,1,1)^2$, $(1,1,-1)^2$, $(-0.03,0.05,0.05)^2$, $(1,-1,1)^2$, $(0.14,-1,1)^3$, $(1,-1,-1)^3$, $(0.03,0.27,-1)^3$, $(-1,-1,-0.1)^3$, $(1,1,1)^3$, $(-1,1,-0.11)^3$, $(-1,-0.03,1)^3$

When evaluated in terms of criteria, under the same algorithm, A-efficiencies of designs generated for the "full model only" (A_{full}) of the second-order model with blocks are higher than those of designs generated for the "all models" (A for best A_w designs) across nearly all combinations of k and b. This is because the objective of the "full model only" designs is to specifically optimize the full second-order model with blocks. However, there are instances where A-efficiencies for the "full model only" designs are equal to those for the "all models". For instance, in Table 1, for $k = 2$, $b = 2$, and $N = 11$ with block sizes of 5 and 6, the A-efficiencies of all designs for the "all models" and the "full model only" are 29.9210. Similarly, for $k = 3$, $b = 2$, and $N = 20$ with block sizes of 10 and 10, the A-efficiencies are 31.4286. For $k = 3$, $b = 3$, and $N = 21$ with block sizes of 7, 7, and 7, only for GA design cases, the A-efficiencies of both designs are 26.5065, which were higher than both EA designs for the "all models" and the "full model only". These examples illustrate that the designs from the "all models" and the "full model only" can sometimes be identical.

Overall, A-efficiency values for "all models" (best A_w designs) are close to the optimal A-efficiency for the "full model only". This pattern is consistent for both GAs and EAs. When fitting the full second-order model, using the A_w-optimality criterion results in designs with A-efficiency values nearly matching those of the full second-order model. Additionally, the "all models" designs support all possible reduced models, making them more robust to model misspecification compared to the "full model only" designs.

4. Conclusions

The study demonstrates that computer-generated designs using GA and EA are more efficient than traditional CCDs in terms of both A_w- and A-efficiencies, across various configurations of variables and blocks. GA designs generally outperform EA designs. The findings support the use of GA designs for more robust and efficient experimental setups, providing significant improvements over CCDs, particularly in complex model scenarios.

Our findings indicate that optimal designs for the second-order model may be less efficient than previously believed. Given the uncertainty of potential reduced models prior to data collection, researchers should explore experimental designs that offer robustness across various possible models. The proposed robust design, known as A_w-optimal designs, provides a promising alternative. Even if the actual model is a second-order model, employing an A-optimal design (full model only) is unnecessary because the A-efficiency of the robust design (all models) for the second-order model is nearly identical to that of the full model-only design. Thus, using the all-models design for a second-order model results in little A-efficiency loss. Overall, the proposed approach provides a useful tool for finding robust response surface designs against model misspecification.

Acknowledgment. This work is supported by a research grant from Research and Development Institute, Rajamangala University of Technology Srivijaya, Thailand.

Conflicts of Interest: The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

[1] A.C. Atkinson, A.N. Donev, R.D. Tobias, Optimum Experimental Designs, with SAS, Oxford University Press, Oxford, 2007. https://doi.org/10.1093/oso/9780199296590.001.0001.

[2] J.J. Borkowski, E.S. Valeroso, Comparison of Design Optimality Criteria of Reduced Models for Response Surface Designs in the Hypercube, Technometrics. 43 (2001), 468–477. https://doi.org/10.1198/00401700152672564.

[3] J.J. Borkowski, P. Turke, B. Chomtee, Using Weak and Strong Heredity to Generate Weighted Design Optimality Criteria for Response Surface Designs, J. Stat. Theory Appl. 10 (2011), 468–477.

[4] G.E.P. Box, J.S. Hunter, W.G. Hunter, Statistics for Experimenters: Design, Innovation, and Discovery, 2nd ed, Wiley-Interscience, Hoboken, 2005.

[5] A. Chairojwattana, S. Chaimongkol, J.J. Borkowski, Using Genetic Algorithms to Generate D_w and G_w-Optimal Response Surface Designs in the Hypercube, Thailand Statistician. 15 (2017), 157–166.

[6] H. Chipman, Bayesian Variable Selection With Related Predictors, Canad. J. Stat. 24 (1996), 17–36. https://doi.org/10.2307/3315687.

[7] R.D. Cook, C.J. Nachtrheim, A Comparison of Algorithms for Constructing Exact D-Optimal Designs, Technometrics 22 (1980), 315–324. https://doi.org/10.1080/00401706.1980.10486162.

[8] V.V. Fedorov, Theory of Optimal Experiments, Academic Press, New York, 1972.

[9] J.H. Holland, Adaptation in Natural and Artificial System: An Introductory Analysis With Applications to Biology Control, and Artificial Intelligence, University of Michigan Press, Oxford, 1975.

[10] W. Limmun, J.J. Borkowski, B. Chomtee, Weighted A-Optimality Criterion for Generating Robust Mixture Designs, Comp. Ind. Eng. 125 (2018), 348–356. https://doi.org/10.1016/j.cie.2018.09.002.

[11] W. Limmun, B. Chomtee, J.J. Borkowski, The Construction of a Model-Robust IV-Optimal Mixture Designs Using a Genetic Algorithm, Math. Comp. Appl. 23 (2018), 25. https://doi.org/10.3390/mca23020025.

[12] W. Limmun, B. Chomtee, J.J. Borkowski, Using Geometric Mean to Compute Robust Mixture Designs, Qual. Reliab. Eng. 37 (2021), 3441–3464. https://doi.org/10.1002/qre.2927.

[13] S. Mahachaichanakul, P. Srisuradetchai, Applying the Median and Genetic Algorithm to Construct D- and G-Optimal Robust Designs Against Missing Data, Appl. Sci. Eng. Progress. 12 (2019), 3–13. https://doi.org/10.14416/j.ijast.2018.10.002.

[14] R.K. Meyer, C.J. Nachtsheim, The Coordinate-Exchange Algorithm for Constructing Exact Optimal Experimental Designs, Technometrics 37 (1995), 60–69. https://doi.org/10.1080/00401706.1995.10485889.

[15] T.J. Mitchell, An Algorithm for the Construction of "D-Optimal" Experimental Designs, Technometrics 16 (1974), 203–210. https://doi.org/10.2307/1267940.

[16] T.J. Mitchell, Computer Construction of "D-Optimal" First-Order Designs, Technometrics 16 (1974), 211–220. https://doi.org/10.2307/1267941.

[17] D.C. Montgomery, Design and Analysis of Experiments, John Wiley & Sons, Hoboken, 2013.

[18] R.H. Myers, D.C. Montgomery, C.M. Anderson-Cook, Response Surface Methodology: Process and Product Optimization Using Designed Experiments, John Wiley & Sons, Hoboken, 2016.

[19] A.F. Shahraki, R. Noorossana, A Combined Algorithm for Solving Reliability-Based Robust Design Optimization Problems, J. Math. Comp. Sci. 7 (2013), 54–62.

[20] H.P. Wynn, Results in the Theory and Construction of D-Optimum Experimental Designs, J. R. Stat. Soc. Ser. B: Stat. Methodol. 34 (1972), 133–147. https://doi.org/10.1111/j.2517-6161.1972.tb00896.x.

[21] P. Yeesa, P. Srisuradetchai, J.J. Borkowski, Model-Robust G-Optimal Designs in the Presence of Block Effects, Appl. Sci. Eng. Progress. 12 (2019), 198–208.

[22] P. Yeesa, P. Srisuradetchai, J.J. Borkowski, A Weighted D-Optimality Criterion for Constructing Model-Robust Designs in the Presence of Block Effects, Songklanakarin J. Sci. Technol. 42 (2020), 1259–1273.