Proceedings of the 2007 Winter Simulation Conference
S. G. Henderson, B. Biller, M.-H. Hsieh, J. Shortle, J. D. Tew, and R. R. Barton, eds.

SUBSET SELECTION AND OPTIMIZATION FOR SELECTING BINOMIAL SYSTEMS APPLIED TO SUPERSATURATED DESIGN GENERATION

Ning Zheng
Theodore T. Allen

Department of Industrial, Welding and Systems Engineering
210 Baker Systems, The Ohio State University
Columbus, OH 43210, U.S.A.

ABSTRACT

The problem of finding the binomial population with the highest success probability is considered when the number of binomial populations is large. A new rigorous indifference zone subset selection procedure for binomial populations is proposed with the proof of the corresponding least favorable configuration. For cases involving large numbers of binomial populations, a simulation optimization method combining the proposed subset selection procedure with an elitist Genetic Algorithm (GA) is proposed to find the highest-mean solution. Convergence of the proposed GA frame work are established under general assumptions. The problem of deriving supersaturated screening designs is described and used to illustrate the application of all methods. Computational comparisons are also presented for the problem of generating supersaturated experimental designs.

1 INTRODUCTION

In some situations of interest, the goal of optimization is to find the binomial system with the highest success probability. This can occur, for example, when one is trying to find the system offering the highest probability of success or "yield" evaluated using simulation. In this paper, simulation optimization methods are proposed for this goal both for cases in which the number of systems of interest is relatively small and for cases in which the number of systems is very large. The general search methods proposed here for the latter cases progress through successive relatively small subset selection problems. Therefore, the paper contains contributions relevant to both subset selection and general stochastic optimization. The problem that we focus on here to illustrate all proposed methods is the derivation of supersaturated design of experiments matrices considered in Allen and Bernshteyn (2003). Yet, the methods here could conceivably be applied to situations as diverse as experimenting to find the best baseball batter and maximizing patient satisfaction by improved staffing in hospital emergency rooms.

1.1 Supersaturated Experimental Designs

An experimental design is an $n \times m$ matrix specifying the combinations of settings for m factor inputs relevant to performing n tests. For example, the first run might be associated with setting all system inputs to a chosen low level and measuring outputs. Experimental designs for screening are useful for studying complicated systems to determine input-output relationships using a small number of experimental tests. Supersaturated experimental designs and associated analysis methods are relevant for cases in which the experimental budget is so small that the number of runs, n, is even less than the number of factors, m.

Selecting the best supersaturated design matrices is a combinatorial optimization problem with a large number of possible solutions. For a screening design with p controllable factors of interest and two possible levels for each factor, the number of possible screening designs with n runs is 2^{np}.

There are two types of optimization criteria used in the literature to define optimality design matrices, deterministic and probabilistic criteria. Deterministic criteria consider some surrogate objectives such as D and G efficiency. Algorithms so-called "exchange algorithms" are widely used for this type of criteria (Li and Wu 1997).

Allen and Bernshteyn (2003) proposed several probabilistic and relatively exotic criteria for generation of new designs such as the so-called "p_{COV}" or coverage probability. p_{COV} is the success probability that the experimental design and analysis method correctly identifies a subset of the factors of interest that covers all factors whose changes truly affect responses of interest. These criteria are exotic because they required some form of numerical method or simulation to be conducted for their evaluation.

The optimization method Allen and Bernshteyn (2003) used for generating their designs was the so-called confidence interval based elitist genetic algorithms (CIEGA,

Bernshteyn (2001)). However, CIEGA is not associated with any rigorous convergence results and computational comparisons of CIEGA with alternatives are limited. In this paper, we propose a new method for generating experimental designs that maximize p_{COV} together with proven convergence results and a thorough computational comparison with relevant alternatives.

1.2 Screening Design Optimization

Exchange Algorithms (EA) and their variants (e.g., Fedorov 1969, Fedorov 1972, Mitchell and Miller 1970 and Donev and Atkinson 1988) are widely used to generate experimental designs. The basic idea of EAs is to start with an initial design (random and/or specified) and update it by exploring all pair-wise exchanges between the rows (or columns) in that design and candidate settings/runs from a list of choices. The method greedily selects the exchange which most improves the criterion value of interest. Then, the process is repeated using the updated design until no improvement can be achieved. EAs are designed for specific optimality criteria with particular structures such that fast update calculations are available to make exhaustive local searches possible. The relevance of exchange algorithms for stochastic optimization design of experiments problems has not yet been established due to the unavailability of fast update methods.

The proposed methods are based on a series of subset selections, i.e., attacking a large optimization problem by comparing N solutions at a time. A special case of the proposed scheme is elitist Genetic Algorithm (GA). Elitist GAs are widely-used general methods, believed to be the most promising GA for noisy functions since relatively few objective function evaluations are used to discriminate bad solutions (De Jong 1975, Aizawa and Wah 1994).

1.3 The Proposed Scheme

Let m be the number of dimensions in the optimization space. A candidate solution can be written as $\boldsymbol{x} = (x_1, x_2, \ldots, x_m)$. A "generation" of N solutions being considered at iteration, t $\boldsymbol{x}_t \equiv (\boldsymbol{x}_{t,1}, \boldsymbol{x}_{t,2}, \ldots, \boldsymbol{x}_{t,N})$ where $\boldsymbol{x}_{t,i}$ are individual solutions. This generation of solutions can also be called a population. A general GA framework is as follows.

Subset Selection: Copy an elitist subset of solutions in the current population to the next population.

In-Fill: Many methods could be used to fill-in the population including embedded search methods such as simulated annealing. Also, one could apply genetic cross-over to generate new solutions by exchanging individual settings x_i between two solutions in current generation.

Immigration/Mutation: Randomly generate new solutions to next generation.

In this paper, we focus on identifying the elitist subset to keep the best solution. In this case the above scheme reduces to a method called a sequential subset selection elitist genetic algorithm (SSSEGA). However, all of the rigorous results that follow are more general than elitist GAs and apply to many population based search methods conforming to the abovementioned scheme.

In general, many variants of elitist GAs have been proposed by changing the specifics of one or more of the three "operators": subset selection, cross-over, and immigration. For example, Aggarwal et al. (1997) proposed an optimized crossover mechanism for the independent set problem, and Poland et al. (2001) used Exchange Algorithms as the mutation operator. Although most of the GA literature is focused on solving deterministic optimization problems, recent research has attempted to use GA to solve stochastic simulation optimization problems such as Hedlund and Mollaghasemi (2001). However, to our knowledge no long run convergence results are available for GAs for stochastic problems have been established.

In this paper, we specifically focus problems involving finding the binomial population with the highest success probability motivated by our design to generate supersaturated designs. For cases in which only a small number of candidate solutions are considered at one time, we propose subset selection methods that are generalizations of the Levin and Robbins (1981) indifference zone procedure. Our procedure permits a subset of predetermined size to be selected while offering standard indifference zone guarantees. We further propose a population based search incorporating our subset selection procedure and present related rigorous convergence results. Finally, we compare the proposed optimization method with two alternatives of generating supersaturated designs from the literature.

The rest of this paper is organized as follows. Section 2 describes the framework of the proposed optimization algorithm. In section 3 we discuss in details the algorithm designed and applied to find the optimal supersaturated designs. Section 4 presents computational comparisons between the proposed method and alternatives. Section 5 gives conclusions and future research.

2 THE PROPOSED ALGORITHM

2.1 The Proposed Method

In this paper, we consider noisy problems such that the best solution previously identified could be discarded by chance at any iteration of the search method. To address this issue, the following proposed subset selection operator guarantees, with a probability no less than P_t, to keep at least a single solution whose objective value (e.g., success probability) is within δ_t to that of the real best solution in

each generation t. P_t and δ_t are allocated such that

$$\prod_{t=0}^{\infty} P_t = P^*, \text{ and} \quad (1)$$
$$\sum_{t=0}^{\infty} \delta_t = \Delta. \quad (2)$$

In the long run, this schema guarantees to find a solution which is within Δ to the global optima with probability P^* (The proof will be shown in section 2.2). This gives SSSEGA a guarantee to find "good enough" solutions, thus has important practical meaning for discrete stochastic optimization problems. Denoting $\boldsymbol{x}_{t,i}$ as the i^{th} solution in t^{th} generation, the framework of SSSEGA is as follows.

Initialize

$t \leftarrow 1$; Predetermine Δ and P^*.

Randomly generate N solutions $(\boldsymbol{x}_{t,1}, \boldsymbol{x}_{t,2}, \ldots, \boldsymbol{x}_{t,N})$.

Repeat

Calculate δ_t and P_t from series satisfying equations (2) and (1). In this paper, computational results derived from the following series with predetermined parameters ε, o, u, and s:

$$\delta_t = \Delta(1-s)s^t, \text{ and} \quad (3)$$
$$P_t = 1 - \frac{u}{(t+o)^{1+\varepsilon}}, \quad (4)$$

where $0 < s < 1$, $o < u < 1$ and $0 < \varepsilon < 1$.

Subset Selection

Identify the subset $\boldsymbol{B}$ of size $b \geq 1$ containing at least one solution having objective value within δ_t to the real best solution in current generation t, with probability no less than P_t. Copy those solutions in $\boldsymbol{B}$ to the next generation.

Crossover

Create c solutions by swapping the genes between any two solutions in current generation t.

Immigration

Randomly generate $N-c-b$ designs; $t \leftarrow t+1$.

Until {the selected stopping rule is satisfied}.

2.2 Convergence Theorem

In this section, we prove that the SSSEGA procedure will converge to a solution having an objective value within Δ to the global optimum with probability greater than P^* in the long run. The proof also assumes that the solution space is finite such that there is a positive probability that acceptable solutions may be randomly hit by immigration and/or infill. The proof of the long run convergence is analogous to existing proofs for simulated annealing based methods in Andradóttir (1999). The following notations are used in the lemmas and theorem. The convergence theorem is based on Ittiwattana (2002). However, in this paper, no "batch for normality" is required for convergence due to the proposed subset selection procedure for binomial population.

$\boldsymbol{X}_t \equiv$ the state of a generation t. Suppose we have N solutions in each generation, then $\boldsymbol{X}_t = (\boldsymbol{X}_{t,1}, \boldsymbol{X}_{t,2}, \ldots, \boldsymbol{X}_{t,N})$, where $\boldsymbol{X}_{t,i}$ is a state of i^{th} solution.

$\boldsymbol{x}_t \equiv$ a specific generation t, or in other words, a realization of $\boldsymbol{X}_t$. Suppose we have N solutions in each generation, then $\boldsymbol{x}_t = (\boldsymbol{x}_{t,1}, \boldsymbol{x}_{t,2}, \ldots, \boldsymbol{x}_{t,N})$, where $\boldsymbol{x}_{t,i}$ is a specific i^{th} solution vector.

$\boldsymbol{E} \equiv$ the state space of the generations, i.e., all possible combinations of solution vectors that can form feasible generations for the GA.

$d(\boldsymbol{x}) \equiv$ the objective value difference between solution $\boldsymbol{x}$ and global optima.

$A_\Delta \equiv$ the optimal state space. $A_\Delta = \{\boldsymbol{x} \in \boldsymbol{E} : d(\boldsymbol{x}) < \Delta\}$.

$A_\Delta^c \equiv$ the completion of A_Δ. $A_\Delta^c = \boldsymbol{E} \setminus A_\Delta$.

$A_{\Delta,t} \equiv$ the state space of the best solution in the generation t. $A_{\Delta,t} = \{\boldsymbol{x}_t \in \boldsymbol{E} : d(\boldsymbol{x}_t) < \Delta_t\}$, where Δ_t is the indifference parameter at generation t. Δ_t is defined as $\Delta_t = \sum_{j=1}^{t} \delta_j$ where δ_j is given by (3) such that $\Delta_\infty = \Delta$.

Note that in our algorithm, the solutions chosen to be copied into the next generation only depend upon the solutions in current generation and the random mechanism. Therefore, $P\{\boldsymbol{X}_t \in A_{\Delta,i} | \boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_t\} = P\{\boldsymbol{X}_{t+1} \in A_{\Delta,i} | \boldsymbol{X}_t\}$, which is the Markovian property, and each forward evolution of one generation is one stage transition. We define the $t-m$ stage transition matrix to an arbitrary state "A" as

$$P^{(m,t)}(\boldsymbol{x}_m, A) = P\{\boldsymbol{X}_t \in A | \boldsymbol{X}_m = \boldsymbol{x}_m\}. \quad (5)$$

We assume that the probability that the next generation $\boldsymbol{x}_i$ will enter A_Δ (i.e., contain a good solution), given that the generation was not in the A_Δ, is greater than or equal to p_{i+1}. This assumption holds when $\boldsymbol{E}$ is finite. With these definitions, we have the following lemma that follows Rudolph (1996) results for deterministic problems.

Lemma 1 *If* $P^{(i,i+1)}(\boldsymbol{x}_i, A_{\Delta,i+1}) \geq p_{i+1} > 0$ *for all* $\boldsymbol{x}_i \in A_{\Delta,i}^c, \forall i = 1, \ldots, t$ *and* $P^{(i,i+1)}(\boldsymbol{x}_i, A_{\Delta,i+1}) \geq P_{i+1}^* > 0$ *for* $\boldsymbol{x}_i \in A_{\Delta,i}, \forall i = 1, \ldots, t$, *then for* $t \geq m+1$, *we have*

$$P^{(i,i+1)}(\boldsymbol{x}_i, A_{\Delta,i+1}) \geq [1 - \prod_{i=m+1}^{t} (1-p_i)] \prod_{i=m+1}^{t} P_i^*. \quad (6)$$

Proof. The result holds for $t = m+1$ because $p_{m+1} < p_{m+1}P_{m+1}^*$ and $P_{m+1} < p_{m+1}P_{m+1}^*$ giving $P^{(m,t)}(\boldsymbol{x}_m, A_{\Delta,t}) = P^{(m,m+1)}(\boldsymbol{x}_m, A_{\Delta,m+1}) \geq p_{m+1}P_{m+1}^* = [1-(1-p_{m+1})]P_{m+1}^*$, which is independent of the choice of $\boldsymbol{x}_m$. Next, we prove by induction that if (6) is true for t, then it is true for t + 1. From the independence of probabilities in Markov chains:

$$P^{(m,t+1)}(\boldsymbol{x}_m, A_{\Delta,t+1}) = \sum_{\boldsymbol{y}_{m+1} \in \boldsymbol{E}} P^{(m+1,t+1)}(\boldsymbol{y}_{m+1}, A_{\Delta,t+1}) P^{(m,m+1)}(\boldsymbol{x}_m, \boldsymbol{y}_{m+1}).$$

Since A consists of $A_{\Delta,t+1}$ and $A^c_{\Delta,t+1}$,

$$P^{(m,t+1)}(\boldsymbol{x}_m, A_{\Delta,t+1}) = \sum_{\boldsymbol{y}_{m+1}\in A_{\Delta,m+1}} P^{(m+1,t+1)}(\boldsymbol{y}_{m+1}, A_{\Delta,t+1}) P^{(m,m+1)}(\boldsymbol{x}_m, \boldsymbol{y}_{m+1}) + \sum_{\boldsymbol{y}_{m+1}\in A^c_{\Delta,m+1}} P^{(m+1,t+1)}(\boldsymbol{y}_{m+1}, A_{\Delta,t+1}) P^{(m,m+1)}(\boldsymbol{x}_m, \boldsymbol{y}_{m+1}). \tag{7}$$

Note that for $\boldsymbol{y}_{m+1} \in A_{\Delta,m+1}$ we have directly from our assumptions and independence:

$$P^{(m+1,t+1)}(\boldsymbol{y}_{m+1}, A_{\Delta,t+1}) \geq \prod_{i=m+2}^{t+1} P_i^*. \tag{8}$$

Inserting this into equation (7) and rearranging, we have:

$$P^{(m,t+1)}(\boldsymbol{x}_m, A_{\Delta,t+1}) \geq \prod_{i=m+2}^{t+1} P_i^* \sum_{\boldsymbol{y}_{m+1}\in A_{\Delta,m+1}} P^{(m,m+1)}(\boldsymbol{x}_m, \boldsymbol{y}_{m+1}) + \sum_{\boldsymbol{y}_{m+1}\in A^c_{\Delta,m+1}} P^{(m+1,t+1)}(\boldsymbol{y}_{m+1}, A_{\Delta,t+1}) P^{(m,m+1)}(\boldsymbol{x}_m, \boldsymbol{y}_{m+1}). \tag{9}$$

If Lemma 1 is true for t, then

$$P^{(m+1,t+1)}(\boldsymbol{y}_{m+1}, A_{\Delta,t+1}) \geq [1 - \prod_{i=m+2}^{t+1}(1-p_i)] \prod_{i=m+2}^{t+1} P_i^*.$$

Therefore, rearranging (9) we have

$$\begin{aligned} P^{(m,t+1)}(\boldsymbol{x}_m, A_{\Delta,t+1}) &\geq \prod_{i=m+2}^{t+1} P_i^* \{ \sum_{\substack{\boldsymbol{y}_{m+1}\in \\ A_{\Delta,m+1}}} P^{(m,m+1)}(\boldsymbol{x}_m, \boldsymbol{y}_{m+1}) \\ &\quad + [1 - \prod_{i=m+2}^{t+1}(1-p_i)] \sum_{\boldsymbol{y}_{m+1}\in A^c_{\Delta,m+1}} P^{(m,m+1)}(\boldsymbol{x}_m, \boldsymbol{y}_{m+1}) \} \\ &= \prod_{i=m+2}^{t+1} P_i^* \{ P^{(m,m+1)}(\boldsymbol{x}_m, A_{\Delta,t+1}) \\ &\quad + [1 - \prod_{i=m+2}^{t+1}(1-p_i)] P^{(m,m+1)}(\boldsymbol{x}_m, A^c_{\Delta,t+1}) \} \\ &= \prod_{i=m+2}^{t+1} P_i^* [P^{(m,m+1)}(\boldsymbol{x}_m, A_{\Delta,t+1}) + P^{(m,m+1)}(\boldsymbol{x}_m, A^c_{\Delta,t+1}) \\ &\quad - \prod_{i=m+2}^{t+1} P^{(m,m+1)}(\boldsymbol{x}_m, A^c_{\Delta,t+1})(1-p_i)] \\ &= \prod_{i=m+2}^{t+1} P_i^* \left[1 - \prod_{i=m+2}^{t+1} P^{(m,m+1)}(\boldsymbol{x}_m, A^c_{\Delta,t+1})(1-p_i) \right]. \end{aligned} \tag{10}$$

$P^{(m,m+1)}(\boldsymbol{x}_m, A_{\Delta,t+1}) \geq p_{m+1}$ by assumption, (10) gives:

$$\begin{aligned} P^{(m,t+1)}(\boldsymbol{x}_m, A_{\Delta,t+1}) &\geq \prod_{i=m+2}^{t+1} P_i^* \left[1 - \prod_{i=m+1}^{t+1}(1-p_i) \right] \\ &\geq \prod_{i=m+1}^{t+1} P_i^* \left[1 - \prod_{i=m+1}^{t+1}(1-p_i) \right]. \end{aligned}$$

Therefore equation (6) in Lemma 1 is proven by induction.

Next we define $P\{\boldsymbol{X}_m = \boldsymbol{x}_m\}$ as the distribution of the starting generation of the solutions in the in the initial generation, m. We have the following lemma.

Lemma 2 *Assuming that the conditions of Lemma 1 are satisfied for specific Δ, and the series of P_t satisfies $P^* = \prod_{i=m+1}^{\infty} P_t$, we then have $\lim_{t\to\infty} P\{\mathbf{X}_t \in A_{\Delta,t}\} \geq P^*$.*

Proof.

$$\begin{aligned} P\{\boldsymbol{X}_t \in A_{\Delta,t}\} &= \sum_{\boldsymbol{E}} P^{(m,t)}(\boldsymbol{x}_m, A_{\Delta,t}) P\{\boldsymbol{X}_m = \boldsymbol{x}_m\} \\ &\geq [1 - \prod_{i=m+1}^{t}(1-p_i)] \prod_{i=m+1}^{t} P_i \sum_{\boldsymbol{E}} P\{\boldsymbol{X}_m = \boldsymbol{x}_m\} \\ &= [1 - \prod_{i=m+1}^{t}(1-p_i)] \prod_{i=m+1}^{t} P_i. \square \end{aligned}$$

The last line follows from Lemma 1 and is independent of the starting point distribution. Therefore,

$$\begin{aligned} \lim_{t\to\infty} P\{\boldsymbol{X}_t \in A_{\Delta,t}\} &\geq [1 - \prod_{i=m+1}^{\infty}(1-p_i)] \prod_{i=m+1}^{\infty} P_i \\ &= \prod_{i=m+1}^{\infty} P_i = P^*. \end{aligned}$$

Now we have the following convergence theorem.

Theorem 1 *The following are sufficient conditions for the SSSEGA procedure to converge to a solution having an objective value within Δ of the global optimum solution with probability greater than P^* in the long run: $N - c - b > 0$ and $b > 0$ (where N is the total number of populations in each generation, and c and b are solutions generated by crossover operator and mutation operator, respectively), and δ_t and P_t satisfy conditions (2), (1), (3) and (4).*

Proof. The first condition of Lemma 1 is satisfied for at least some values of $p_i > 0$ because $N - c - b > 0$, and the second condition is guaranteed due to the subset selection procedure in each generation in SSSEGA. The conditions of Lemma 2 are satisfied because of equation (1). Thus long run convergence of SSSEGA is guaranteed. $\square$

Note that Theorem 1 is general such that it could apply to simulation optimization in which the outputs associated with alternative solutions could come from custom distributions. However, the problem of maximizing the success probabilities considered next offers perhaps the easiest opportunity

to satisfy the needed conditions without assumptions such as approximate normality.

3 APPLICATION TO BINOMIAL OUTPUTS

In section 2, we have described SSSEGA procedure and offered conditions under which the method could converge to an acceptable solution largely regardless of the distributions of the outputs associated with each candidate solution. In this section, we focus on the problem of maximizing success probability, i.e., selecting the binomial system with the highest success probability. Motivated by the need to satisfy the conditions of Theorem 1, an open sequential indifference zone subset selection method called the "generalized Levin-Robbins" (GLR) procedure is proposed for the Subset Selection operator. The rigorous guarantees associated with the GLR procedure are then proven. Also, the insertion of the GLR method into abovementioned optimization scheme and SSSEGA is briefly discussed. Finally, the application of GLR and SSSEGA to the problem of generating supersaturated designs from Allen and Bernshteyn (2003) is described as an application for all methods.

3.1 The Generalized Levin-Robbins (GLR) Procedure

Bechhofer et al. (1995) reviewed procedures to deal with the subset selection and indifference zone problems for binomial populations. The goals of the procedures are generally to select either the best one with indifference zone parameter δ, or a subset with random size that contains the best one among binomial populations. The framework proposed in this paper and the associated convergence proof conditions place two type of requirements on the subset selection method: (1) it needs to terminate with a subset of a user-specified size, b, and (2) there must be at least one solution in the subset having an objective value within a user-specified difference, δ, as compared with the *real* best one among the solutions in the population. To our knowledge, there is no previously proposed subset selection procedure designed for *binomial* populations, although similar procedures exist for *normal* populations (e.g., Koenig and Law 1985).

The procedure proposed here is called the generalized Levin-Robbins procedure because it is an extension of the procedure proposed by Levin and Robbins (1981). The procedure from Levin and Robbins (1981) is designed to pick only a single system or solution ($b = 1$) whose success probability is acceptable, i.e., within δ of the best success probability associated with any of the m systems being compared. The following procedure terminates having a subset with b solutions in which at least one solution has acceptable probability.

Let $\boldsymbol{Y}_{(i)}^{(n)}$ denote the ordered accumulated number of successes observed of the m populations, $i = 1,\ldots,m$ after n vector trials, i.e., $\boldsymbol{Y}_{(1)}^{(n)} \geq \boldsymbol{Y}_{(2)}^{(n)} \geq \ldots \geq \boldsymbol{Y}_{(m)}^{(n)}$. The procedure is as follows.

Repeat Evaluate all m solutions one more time.

Until $\{\boldsymbol{Y}_{(b)}^{(n)} - \boldsymbol{Y}_{(b+1)}^{(n)} = r\}$ where r is given by equation (12) below.

On Termination Select the solutions corresponding to the observed successes $\boldsymbol{Y}_{(1)}^{(n)} \geq \boldsymbol{Y}_{(2)}^{(n)} \geq \ldots \geq \boldsymbol{Y}_{(b)}^{(n)}$.

The main difference between the above procedure and the one from Levin and Robbins (1981) is that we keep b solutions instead of 1. Also, the value of r used in our procedure is generally lower than theirs reflecting the improved efficiency associated with the easier goal of keeping b solutions instead of 1. Note that Levin and Robbins (1981) generated many results relevant to the GLR procedure here in recursive proofs associated with their own procedure. The following Lemma 3 and Theorem 2 establish that the above GLR method guarantees the achievement of at least a single acceptable solution (with success probability within δ of the best among m) with probability greater than P^*.

In the following, we assume without loss of generality that the binomial systems are arranged in descending order of success probability such that $p_1 \geq p_2 \geq \ldots \geq p_m$. Also, we denote the probability of selecting a subset containing the true best solution as p_{CS}.

Lemma 3 *There exists a whole number r such that if the GLR method is applied, p_{CS},*

$$p_{CS} \geq \frac{\sum_{\{k_1,k_2,\ldots,k_{b-1}\} \in I^{b-1}\{2,\ldots,m\}} \left(w_1 w_{k_1} \ldots w_{k_{b-1}}\right)^r}{\sum_{\{i_1,i_2,\ldots,i_b\} \in I^b\{1,2,\ldots,m\}} \left(w_{i_1} w_{i_2} \ldots w_{i_b}\right)^r} \geq P^*, \tag{11}$$

where $\{k_1,k_2,\ldots,k_{b-1}\}$ is a set of integers, $I^b\{i,i+1,\ldots,j\}$ denotes the integer set space with b unique integers in each set and each integer belongs to the set $\{i,i+1,\ldots,j\}$, and $w_i = \frac{p_i}{1-p_i}$.

Proof. First, we derive a formula for the probability of selecting any specific subset of interest. Let $g_1 \neq g2 \neq \ldots \neq g_b$ be integers denoting the indices that refer to a specific selection, written $\{g_1 \neq g2 \neq \ldots \neq g_b\}$. Levin and Robbins (1981) proved for any whole number r the GLR procedure selection probability is:

$$P(\text{select subset } \{g_1 \neq g2 \neq \ldots \neq g_b\}) \geq \frac{\left(w_1 w_{k_1} \ldots w_{k_{b-1}}\right)^r}{\sum_{\{i_1,i_2,\ldots,i_b\} \in I^b\{1,2,\ldots,m\}} \left(w_{i_1} w_{i_2} \ldots w_{i_b}\right)^r} \geq P^*.$$

Therefore, the probability of selecting a subset that contains the best population is the summation of all probabilities of selecting any possible subsets $\{1,k_1,k_2,\ldots,k_{b-1}\}$ where

$\{k_1, k_2, \ldots, k_{b-1}\} \in I^{b-1}\{2, 3, \ldots, m\}$. Summing both sides gives the inequality (11). □

Theorem 2 *A sufficient condition for the GLR procedure terminates with at least a single solution with success probability within δ of the best probability among all m solutions with probability greater than P^* is*

$$r = \left\lceil \frac{\log\left[\left(\frac{m}{b} - 1\right)\frac{P^*}{1-P^*}\right]}{\log\left(\frac{w_1}{w_a}\right)} \right\rceil \tag{12}$$

where $\lceil \cdot \rceil$ is the ceiling operation and

$$w_a = \frac{p_0^* - \delta}{1 - p_0^* + \delta}, \quad w_1 = \frac{p_0^*}{1 - p_0^*}, \quad p_0^* = \frac{1+\delta}{2}$$

Proof. Levin and Robbins (1981) effectively established that the least favorable configuration for the GLR procedure is

$$p_1 = p_0, \text{ and } p_2 = p_3 = \cdots = p_m = p_0 - \delta,$$

According to Lemma 3, we have

$$P^* \geq \frac{\sum_{\substack{\{k_1,k_2,\ldots,k_{b-c}\}\in \\ I^{b-c}\{c+1,\ldots,m\}}} \left(w_1 w_{k_1} \ldots w_{k_{b-1}}\right)^r}{\sum_{\substack{\{i_1,\ldots,i_{b-1}\}\in \\ I^{b-c}\{2,\ldots,m\}}} \left(w_1 w_{i_1} \ldots w_{i_{b-c}}\right)^r + \sum_{\substack{\{i_1,\ldots,i_b\}\in \\ I^{b}\{2,\ldots,m\}}} \left(w_{i_1} \ldots w_{i_b}\right)^r}$$

$$= \frac{\binom{m-1}{b-1}\left(w_1^c w_a^{b-c}\right)^r}{\sum_{i=0}^{1}\binom{m-1}{b-1+i}\left(w_1^{c-1} w_a^{b-c+1}\right)^r} = \frac{1}{1 + \frac{m-b}{b}\left(\frac{w_a}{w_1}\right)^r}.$$

Minimizing the right-hand side of the above equation gives $p_0^* = \frac{1+\delta}{2}$ and r in equation (12), which is the tightest bound. □

We also propose the following variant of the GLR method called the generalized Levin Robbins with Elimination (GLRE) for improved computational efficiency.

Repeat Evaluate all remaining k solutions one more time.
If $\{\boldsymbol{Y}_{(b)}^{(n)} - \boldsymbol{Y}_{(k)}^{(n)} = r\}$, then
Eliminate the solution corresponding to $\boldsymbol{Y}_{(k)}^{(n)}$.
Until $\{k = b\}$ where r is also given by equation (12).

We conjecture that proof that GLRE offers equivalent guarantees to those associated with GLR will likely be possible and propose it for future study. Leu and Levin (1999) developed rigorous proof for $b = 1$ case.

Table 1 compares the computational performance of GLR and GLRE method for several cases each involving the least favorable configuration (LFC) using simulation. The results show that both methods are fairly conservative in that the simulated success probability is considerably higher than P^*. Also, while not yet associated with rigorous claims, the GLRE method offers a computational advantage over the GLR method that might be considered important. For example, in the $m = 20, b = 5, \delta = 0.1, P^* = 0.9$ case, the GLRE offers the desired guarantee with less than one fourth the number of simulation runs.

Table 1: GLR and GLRE performance in the LFC based on 1000 simulations each (m = 20, δ =0.1).

	b	P^*	r	Estim. p_{CS}	Avg. # rounds	Avg. # evals
GLRE	15	0.80	1	0.850	5.7	103.4
	15	0.90	3	0.978	75.7	1329.5
	15	0.95	5	0.992	225.1	3939.3
	10	0.60	2	0.789	32.2	442.3
	10	0.80	4	0.936	140.1	1925.7
	10	0.90	6	0.994	322.0	4415.8
	5	0.30	1	0.370	4.6	48.0
	5	0.60	4	0.791	116.5	1102.4
	5	0.90	9	0.989	568.7	5470.9
GLR	15	0.80	1	0.897	6.5	129.6
	15	0.90	3	0.997	147.5	2949.1
	15	0.95	5	1.000	443.7	8874.9
	10	0.60	2	0.950	71.2	1424.4
	10	0.80	4	0.998	337.5	6750.4
	10	0.90	6	1.000	765.1	15301.7
	5	0.30	1	0.410	6.3	126.0
	5	0.60	4	0.977	234.7	4693.3
	5	0.90	9	1.000	1256.3	25125.8

3.2 The Derivation of SSSEGA for Binomial Populations

Next, we describe all the features of the SSSEGA applied to supersaturated design generation problems. These features include the usage of the GLR method and a specific type of mutation to ensure the convergence conditions. Also, the usage of Bernoulli cross-over and the GLRE method is described to achieve good computational performance.

Theorem 2 guarantees that inserting the GLR procedure into the scheme for optimization in Section 2 results in the achievement of the conditions needed to prove long run convergence in Theorem 1. These conditions are that subset selection guarantees at least one solution with success probability (objective value) within δ of the best in the generation is achieved with probability greater than P^*. Similarly, selecting from all possible solutions with equal probability guarantees a nonzero probability that an accept-

able solution will be included in the next population, which is needed for the conditions of Lemma 1 and Theorem 1.

Considering that the GLR procedure can be conservative and computationally wasteful, we focus computational results on the more efficient GLRE procedure. However, GLRE is an open procedure. Therefore, the expected number of evaluation rounds may be very large if all the solutions we are testing are very close to each other or even identical. To prevent the procedure stalling, we include in the method a practical stoppage of the subset selection operator when the number of rounds exceeds 5000. Then, the procedure takes all the remaining solutions into the elitist subset. Further, if the number of left solutions exceeds 30% of the generation size, we increase the generation size to ten times of the number of remaining solutions.

Finally, since convergence results do not depend on the specifics of the in-fill phase, we somewhat arbitrarily selected the well-known "Bernoulli crossover" for generating new candidate solutions, which is defined as follows. For any two solutions $\boldsymbol{x}_{t,i}$ and $\boldsymbol{x}_{t,j}$ in current generation t, generate two solutions $\boldsymbol{x}_{t+1,k}$ and $\boldsymbol{x}_{t+1,l}$ in according to the following:

$$\boldsymbol{x}_{t+1,k}(m) = \begin{cases} \boldsymbol{x}_{t,i}(m), & \text{if } U_m \leq \theta \\ \boldsymbol{x}_{t,j}(m), & \text{otherwise} \end{cases}, \text{ and}$$

$$\boldsymbol{x}_{t+1,l}(m) = \begin{cases} \boldsymbol{x}_{t,j}(m), & \text{if } U_m \leq \theta \\ \boldsymbol{x}_{t,i}(m), & \text{otherwise} \end{cases}.$$

Here, $\boldsymbol{x}(m)$ is the m^{th} run in design $\boldsymbol{x}$, and U_m is a random number drawn from uniform distribution $U(0,1)$. θ is a user specified number and $0 < \theta < 1$.

3.3 Application to Supersaturated Design Generation

We shall use the same coding method and simulation framework in Allen and Bernshteyn (2003). In their work, each design is coded by specifying the index (or position) of each of its runs in the set of all possible candidate settings (call this set a *candidate set*). Thus a design is expressed as a vector of indices. The criterion p_{COV} is obtained through simulation. Following a series of assumptions, one can generate assumed true models (which includes all linear terms plus all possible interactions between any two factors) of the system which the screening design will apply to. For each assumed true model, stepwise regression is applied to identify the important factors. If all the important factors are successfully identified, an indicator function is given value 1, otherwise 0. And we call this process an *evaluation* for the design. After simulating a large number of assumed true models and average all the values of the indicator functions, an estimated value of p_{COV} is obtained. Readers are referred to Allen and Bernshteyn (2003) for the details of the simulation.

4 COMPUTATIONAL COMPARISONS

In this section, the proposed SSSEGA procedure is compared to Fedorov Exchange Algorithm (FEA) (Fedorov 1969, Fedorov 1972), which is believed to be the best EA "in terms of producing 'good' designs" (Nguyen and Miller 1992), and CIEGA (Bernshteyn 2001) for the generation of supersaturated designs.

4.1 Methods Settings

EA was designed for surrogate deterministic optimization criteria such as D-optimality. To solve the stochastic optimization problem in this work, the determination of a "better" screening design need to be changed instead of D-efficiency. The modification is, when comparing two designs, compare the percentage of successes by evaluating each design k times with different random seeds, where k is an integer (we call the corresponding procedure k-EA).

To balance the trade-off between the broadness of exploration and probability of "correctly" select the better design in each exchange, we study different values of k = 500, 1000 and 5000.

For SSSEGA, although its parameter setting does not affect its long run convergence to the global optimum, it does influence the practical performance. We suggest the settings as in Table 2 because they works well practically. Also notice that different problems may have different "good" parameter settings for SSSEGA because of the different characteristics of the problems. Finding optimal settings under different cases will be part of the future research.

Table 2: SSSEGA parameter settings used.

N	c	b	u	s	ε	o	Δ
100	60	10	20	0.95	0.0001	500	0.1

We test three different supersaturated design problems from Allen and Bernshteyn (2003) with different number of factors and runs, which are listed in Table 3 respectively. Designs with 7 factors, 11 factors and 14 factors are tested to compare the performances of the alternatives under different problem sizes. Different runs (6, 10 and 8, respectively) are tested to compare the performances under different optimal p_{COV}. To examine the short and long run performance of each algorithm, 500000, 2000000 and 8000000 total numbers of evaluations are used as the stopping criteria for each case and each algorithm comparing.

For each problem, we run each procedure for 20 replicates. For each replicate, 20000 evaluations are conducted on the final solution (i.e., the final screening design) to obtain the accurate value of the p_{COV} achieved.

4.2 Computational Results

Table 3 shows the computational results under different total numbers of evaluations. In most cases SSSEGA outperforms other alternatives or has comparable performance in terms of average p_{COV} achieved. For instance, in the 11-factor-and-10-run case, using 0.5 million evaluations, SSSEGA can obtain screening designs with $p_{COV} = 0.713$, which is better than what k-EA achieves using 8 million evaluations (which is 0.627) and comparable to the p_{COV} CIEGA obtains using 2 million evaluations (which is 0.704).

SSSEGA also surpasses other methods in terms of stableness and robustness. In all cases in Table 3, SSSEGA has smaller or comparable standard deviations of the achieved p_{COV}'s against the other two alternative methods, which indicates that SSSEGA is more stable than the other procedures. The out-performance of SSSEGA (under the same setting) in each case also shows its robustness and generality in dealing with different screening designs. The quality of the designs (i.e., p_{COV}) achieved by SSSEGA continuously improve with the total number of evaluations increasing, which supports the long run convergence of SSSEGA. In contrast, The quality of the designs (i.e., p_{COV}) achieved by k-EAs does not necessarily improve with the total number of evaluations increasing, which indicates the incapability of k-EAs to keep the "good" designs they hit.

We use ANOVA followed by Bonferroni's multiple comparisons procedure to statistically justify the performance of SSSEGA against other alternatives.

For each type of alternatives, (i.e., k-EAs and CIEGA), we select the ones with the best performance under each stopping total number of evaluations to compete with SSSEGA. For ANOVA, we use the linear model $p_{COV} = p_0 + p_i$, where i = k-EA, CIEGA or SSSEGA. The results of the statistical analysis show that for total number of evaluations = 0.5 million and the case with 11 factors and 10 runs, SSSEGA achieves significantly better p_{COV} with total α level 0.05.

Table 3: Average p_{COV} (standard deviations) achieved.

Case	Algorithm	Evaluations (million)		
		0.5	2	8
7 factors 6 runs	500-EA	0.707 (0.043)	0.743 (0.027)	0.749 (0.024)
	1000-EA	0.704 (0.039)	0.751 (0.018)	0.765 (0.015)
	5000-EA	0.689 (0.068)	0.730 (0.026)	.751 (0.021)
	CIEGA	0.771 (0.012)	0.778 (0.011)	0.794 (0.009)
	SSSEGA	0.776 (0.009)	0.787 (0.008)	0.790 (0.009)
11 factors 10 runs	500-EA	0.613 (0.029)	0.625 (0.026)	0.613 (0.027)
	1000-EA	0.619 (0.038)	0.626 (0.037)	0.627 (0.030)
	5000-EA	0.635 (0.035)	0.643 (0.032)	0.641 (0.030)
	CIEGA	0.683 (0.013)	0.704 (0.011)	0.731 (0.019)
	SSSEGA	0.713 (0.014)	0.721 (0.014)	0.729 (0.014)
14 factors 8 runs	500-EA	0.306 (0.021)	0.296 (0.025)	0.302 (0.021)
	1000-EA	0.308 (0.021)	0.300 (0.031)	0.303 (0.022)
	5000-EA	0.317 (0.014)	0.310 (0.021)	0.317 (0.016)
	CIEGA	0.344 (0.007)	0.348 (0.004)	0.351 (0.007)
	SSSEGA	0.344 (0.008)	0.351 (0.007)	0.354 (0.008)

5 CONCLUSIONS

In this paper, a framework based on subset selection is proposed for simulation optimization and conditions for its convergence are proven. Then, a specific subset selection for the case of comparing binomial systems and success probabilities is proposed called the generalized Levin-Robbins (GLR) procedure. Guarantees associated with the proposed GLR procedure are provided such that the insertion of the GLR procedure guarantees convergence of the overall optimization framework. A variant of the GLR procedure called generalized Levin-Robbins with Elimination (GLRE) is also proposed and its computational benefits are discussed.

The so-called sequential subset selection elitist genetic algorithm (SSSEGA) is then formally proposed as an example of the overall optimization framework building upon the proposed subset selection methods (GLR and GLRE). The application of SSSEGA to the generation of supersaturated experimental designs is then described and computational results comparing SSSEGA with alternative methods from the experimental design literature are given. The results suggest that SSSEGA offers a comparatively promising approach for generating supersaturated designs from simulation optimization.

Some opportunities for future research were also identified. First, the rigorous properties of the GLRE method could be established formally. Also, additional even more efficient subset selection methods could be designed and characterized for the case of comparing binomial success probabilities. In applying GLRE in the context of SSSEGA, it might be possible to avoid any need for increasing the subset and/or overall population size. Variable sizes generally increase the complexity of the methods and might

not be needed. Finally, although discussed here, genomic applications of supersaturated designs could be important for efficient derivation of transcriptional networks. Using simulation optimization for the derivation of supersaturated designs could offer the ability to tailor methods to specific needs in genomic research both in academia and industry.

ACKNOWLEDGMENTS

We thank Waraphorn Ittiwattana for her many contributions including in the proof-making and literature review.

REFERENCES

Aggarwal, C. C., J. B. Orlin, and R. P. Tai. 1997. Optimalized crossover for the independent set problem. *Operations Research* 45 (2): 226–234.

Aizawa, A. N., and B. W. Wah. 1994. Scheduling of genetic algorithms in a noisy environment. *Evolutionary Computation* 2 (2): 97–122.

Allen, T. T., and M. Bernshteyn. 2003. Supersaturated designs that maximize the probability of finding the active factors. *Technometrics* 45:1–8.

Andradóttir, S. 1999. Accelerating the convergence of random search methods for discrete stochastic optimization. *ACM Transactions on Modeling and Computer Simulation* 9 (4): 349–380.

Bechhofer, R. E., T. J. Santner, and D. Goldsman. 1995. *Design and analysis of experiments for statistical selection, screening and multiple comparisons*. New York: John Wiley & Sons.

Bernshteyn, M. 2001. *Simulation optimization methods that combine multiple comparisons and genetic algorithms with applications in design for computer and supersaturated experiments*. Ph.D. Thesis, The Ohio State University, Columbus, Ohio.

De Jong, K. A. 1975. *An analysis of the behavior of a class of genetic adaptive systems*. Ph.D. thesis, University of Michigan, Ann Arbor.

Donev, A. N., and A. C. Atkinson. 1988. An adjustment algorithm for the construction of exact D-optimum experimental designs. *Technometrics* 30:429–433.

Fedorov, V. V. 1969. *Theory of optimal experiments*. Preprint No. 7 LSM, Moscow State University, Moscow.

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

Hedlund, H. E., and M. Mollaghasemi. 2001. A genetic algorithm and an indifference-zone ranking and selection framework for simulation optimization. In *Proceedings of the 2001 Winter Simulation Conference*, ed. B. A. Peters, J. S. Smith, D. J. Medeiros, and M. W. Rohrer, 417–421.

Ittiwattana, W. 2002. *A method for simulation optimization with applications in robust process design and locating supply chain operations*. Ph.D. Thesis, The Ohio State University, Columbus, Ohio.

Koenig, L. W., and A. M. Law. 1985. A procedure for selecting a subset of size m containing the l best of k independent normal populations, with applications to simulation. *Communications in Statistics: Simulation and Computation* 14:719–734.

Leu, C.-S., and B. Levin. 1999. On the probability of correct selection in the levin-robbins sequential elimination procedure. *Statistica Sinica* 9:879–891.

Levin, B., and H. Robbins. 1981. A genetic algorithm and an indifference-zone ranking and selection framework for simulation optimization. In *Proceedings of the National Academy of Sciences of the United States of America*, Volume 78, 4663–4666.

Li, W. W., and C. F. J. Wu. 1997. Columnwise-pairwise algorithms with applications to the construction of supersaturated designs. *Technometrics* 39:171–179.

Mitchell, T. J., and F. L. Miller. 1970. Use of "design repair" to construct designs for special linear model. Report ORNL-4661, Mathematics Division, Oak Ridge National Laboratory, Oak Ridge.

Nguyen, N.-K., and A. J. Miller. 1992. A review of some exchange algorithms for constructing discrete d-optimal designs. *Computational Statistics & Data Analysis* 14:489–498.

Poland, J., A. Mitterer, K. Knödler, and A. Zell. 2001. Genetic algorithms can improve the construction of d-optimal experimental designs. In *Advances In Fuzzy Systems and Evolutionary Computation*, ed. N. Mastorakis, 227–231. WSES 2001.

Rudolph, G. 1996. Convergence of evolutionary algorithms in general search spaces. In *Proceedings of the Third IEEE Conference on Evolutionary Computation*, 50–54. Piscataway, New Jersey: IEEE Press.

AUTHOR BIOGRAPHIES

NING ZHENG is a Ph.D. candidate in the Department of Industrial, Welding and Systems Engineering at The Ohio State University. His research interests include Online Optimization, Information Analysis and Comprehension, and Database Indexing and Processing for Optimization Queries. His web page can be found via ⟨ningsean.googlepages.com⟩.

THEODORE T. ALLEN is an associate professor in the Department of Industrial, Welding and Systems Engineering at The Ohio State University. His research interests include Experimental Design, Simulation Optimization and Manufacturing Process Engineering. His web page can be found via ⟨www-iwse.eng.ohio-state.edu/ISEFaculty/allen/allen.htm⟩.