Evaluating **algorithm efficiency for optimizing experimental designs with correlated data**

Lazarus K. Mramba\*,1, and Salvador A. Gezan2

1 University of Florida, Department of Medicine, Gainesville, FL32610, USA

2 University of Florida, School of Forest Resources and Conservation, Gainesville, FL32611, USA

Received zzz, revised zzz, accepted zzz

The search for efficient methods and procedures to optimize experimental designs is a vital process in field trials that is often challenged by computational bottlenecks. Most existing methods ignore the presence of some form of correlations in the data to simplify the optimization process at the design stage. This study explores several algorithms for improving experimental designs using a linear mixed models statistical framework adjusting for both spatial and genetic correlations based on A- and D-optimality criteria. Relative design efficiencies are estimated for an array of algorithms including pairwise swap, genetic neighborhood, and simulated annealing and evaluated with varying levels of heritabilities, spatial and genetic correlations. Initial randomized complete block designs were generated using a stochastic procedure and can also be imported directly from other programs such as CycDesign. Results showed that at a spatial correlation of 0.6 and a heritability of 0.3, under the A-optimality criterion, both simulated annealing and simple pairwise algorithms achieved the highest design efficiencies of 7.4% among genetically unrelated individuals. In contrast, results under D-optimality criterion indicated that simulated annealing had the lowest design efficiency. The simple pairwise algorithm consistently maintained highest design efficiencies in all evaluated conditions. Design efficiencies for experiments with full-sib families decreased with increasing. The number of successful swaps appeared to decrease with increasing heritability and were highest for both simulated annealing and simple pairwise algorithms, and lowest for genetic neighbor algorithm.

*Key words:* Genetic relationships; Greedy algorithms; Pairwise swap; Simulated annealing; Spatial correlations.

# 1 Introduction

Experimental designs for field trials are a critical part of any biological research setting, where several decisions are considered in order to select the most appropriate arrangement of the experimental units. Construction of experimental designs requires treatments to be replicated, randomized and apply some blocking to reduce heterogeneity as strategies to make sure that results of an experiment are unbiased, optimal, and allow to perform appropriate inferences to a larger population (Welham et al., 2015). For plant breeding, field trials are an important component that help to evaluate and select the genotypes (or treatments) with superior performance to be used as future parents or commercial varieties (Piepho et al., 2008). Breeding trials are often characterized by testing a large number of genetic entries but with limited replication. The effect of these entries is often estimated by fitting a linear mixed model (LMM) that considers them as a random effect, and that incorporates genetic relationships by the definition of a variance-covariance matrix obtained based on pedigree information.

Several proposed experimental designs exist, and these can be generated using available statistical software. However, the process of generating an optimal or near-optimal design, that maximizes the amount of information extracted with the minimum resources, is often ignored due to intensive computational requirements, particularly for experimental designs with large number of treatments. Some authors have presented efficient procedures to construct experimental designs for breeding trials, including incomplete blocks, row-column and augmented designs (e.g. John and Williams, 1995; Williams, John and Whitaker, 2006). However, these are restricted to the assumption of fixed treatments effects, and therefore ignore the information provided by the genetic relationships. At the same time, it has been shown that modelling field spatial correlations (e.g., by incorporating an autoregressive variance structure) results in more efficient designs than assuming that residuals are independent and identically distributed (Gezan, White and Huber, 2010; Butler, Smith and Cullis, 2014). Here, the framework of mixed models is advantageous over traditional linear models since they allow for specification of appropriate variance-covariance structures for both factors (i.e., genetic entries) and residuals (i.e., spatial correlation), providing greater flexibility and more efficient statistical analyses.

To improve experimental designs, an optimality criterion is used in the implemented search algorithm. *A*- and *D*- information based criterion are the most widely used procedures in field experiments to generate designs (Chernoff, 1953; Cullis et al., 1989; Cullis, Smith and Coombes, 2006) and most recently used by Butler et al., (2014) and Mramba et al., (2018) to design experiments with correlated observations as these procedure are useful in the process of selecting the optimal design (Kuhfeld, 2010). *A*-optimality criterion seeks to minimize the average variance of random treatment effects and can be expressed as: , where **M**(Ω) is the inverse of an information matrix of the treatment (or genetic) effects from a given design layout . *D*-optimality was introduced by (Wald, 1943) and minimizes the determinant of **M**(Ω) which can be interpreted as minimizing the generalized variance of the treatment effects (Kuhfeld, 2010) by choosing designs which minimize the volume of the joint confidence ellipsoid (Das, 2002) and is given by , for |**M**(Ω)| ≠ 0.

Often, search algorithms involve interchanging pairs of treatments and re-evaluating the efficiency of the new design compared to the previous. Some of the computer search algorithms available include: 1) pairwise swap procedure, and its variants where a single or multiple pairs of treatments are swapped at a time (John and Williams, 1995), and simulated annealing (SA) where a cooling strategy is employed (Kirkpatrick, Gelatt and Vecchi, 1983), among others. Most of the applications of these algorithms focus on the analysis of data and very little has been done on their applications to improve the designs of such experiments, yet, estimated parameters from improved designs can be obtained with increased precision if variability of treatment effects is minimized (Mramba et al., 2018). Although there exist other standard software tools such as CycDesigN, GenStat, SAS procedures and DiGGer package in R (R Core Team, 2018), the focus of this paper is to evaluate performance of different algorithms based on a linear mixed model framework which optimally accounts for possible sources of variation in an experiment at the design stage. Specifically, these proposed procedures will model both the spatial correlations and genetic relatedness structures as evidenced in most plant breeding experimental designs.

The main objective of this study is to evaluate the efficiency of diverse search algorithms to generate improved randomized complete block (RCB) designs applying *A*- or *D*-optimality criteria, while accounting for both spatial and genetic correlations using linear mixed models with applications in plant breeding trials. This will be done by initially generating experimental layouts through a random process and later applying an array of proposed search algorithms to improve the initial experimental layouts. Several varying field conditions that include a range of heritabilities, genetic relatedness structures and spatial correlations are evaluated.

**2 Material and methods**

## 2.1 Statistical model for randomized complete block designs

A linear mixed model (LMM) can be expressed as **y = X**𝛃 **+ Zg + e**, where **y** is a vector of observed phenotypes (responses); **X** is an incidence matrix of fixed block effects; **β** is a vector of fixed block effects; **Z** is an incidence matrix of random treatment effects; **g** is a vector of random treatment effects, with **g** ~ MVN(**0**, **G**), where **G** = σg2**A** for genetically correlated observations with **A** being a numerator relationship matrix calculated from pedigree information to account for additive genetic relatedness between individuals or calculated from molecular markers, and **G** is a variance-covariance matrix for genetic relationships. For instance, **G** = σg2**I** for genetically unrelated individuals. The vector **e** represents residual errors, with, **e** ~ MVN(**0**, **R**), where **R** is a variance-covariance matrix for modelling correlated errors. Most often, **R** is modelled with an autoregressive error structure of order 1 (Gilmour et al., 2009) using the equation **R** = σe2**∑x**(ρx)⊗∑y(ρy), where Var(eij) = σe2 and Cov(eij, ei’j’) = σe2ρx|dx|ρy|dy|, and where |dx| = |xi – xi’| and |dx| = |yj – yj’| are the row and column coordinates absolute distances, respectively; ⊗ is a Kronecker product; and **∑x** and ∑y are matrices with autocorrelation parameters ρxand ρyfor rows and columns, respectively. If residuals are assumed to be independent and identically distributed then **R** = σe2**I.** To obtain the variance-covariance matrix of random treatment effects, linear mixed model normal equations are solved as described by Henderson (1950) and computations implemented as discussed by (Harville, 1997; Hooks et al., 2009) to obtain an expression given by

(2)

from which the trace and determinant of the matrix **M**(Ω) are calculated based on *A*- and *D*-optimality criteria, respectively (See methods details (Mramba et al., 2018)).

## 2.2 Algorithms

The proposed algorithms comprise of 1) simple pairwise (SP), that only swaps a pair of treatments at a time; 2) variants of pairwise procedure that swaps a group of α treatments at a time, and is identified as greedy pairwise (GP); 3) a genetic neighbourhood (GN), that takes into consideration the genetic relatedness of the direct neighbouring experimental units; and 4) simulated annealing (SA), that also swaps a pair of treatments at a time, but also accepts poor designs with a given probability which diminishes with time.

For any of these algorithms, the procedure involves randomly generating *m* initial experimental layouts, denoted as Ω*i*. For each layout, the variance-covariance matrix of the treatment effects **M**(Ω) is obtained, and its criterion value is calculated (as trace or determinant). Next, from the *m* designs, the single “best” initial (non-improved) experimental layout with the best criterion value is selected. After this, a given optimization algorithm is applied for *p* iterations. For all implemented algorithms, the output is a list of objects including the improved experimental layout, a vector with criterion values and iterations of the sequentially accepted (successful) designs, and a vector of all criterion values from all iterations, whether the swap was successful or not. Following is a detailed description of the implemented algorithms.

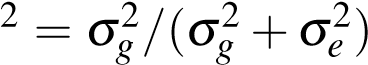
For the SP algorithm, the following steps are undertaken after selecting the best initial with criteria value : 1) randomly interchange a single pair of treatments within a randomly selected block to produce a new layout, ; 2) re-calculate a new criterion value ; 3) if , accept as the new layout; and 4) repeat steps 1 to 3 for a total of *p* iterations and produce the output.

Greedy algorithms (GP) are more aggressive variants of the simple pairwise algorithm (SP) that allow multiple treatments to be randomly interchanged within a block. In order to evaluate a spectrum of alternative implementations, this algorithm was implemented by varying the number of treatments to be swapped simultaneously, denoted as GPα, where α refers to the number of treatments swapped. The algorithm allows specification of any even number of treatments to be swapped at a time. Tested procedure were denoted as GP4, GP14 and GP98 for randomly swapping 4, 14 and 98 treatments simultaneously on each iteration within a randomly selected block, respectively. Numbers 14 and 98 were chosen as a proportion (or 50 %) of the treatments to be swapped at a time, in an experiment with 30 and 196 treatments, respectively, whereas 4 was chosen as a close value to 2 to detect any small changes in improvement of the design when a single pair or double pairs of treatments are swapped in each iteration. Steps 1 to 4 apply as described under the SP procedure.

The GN algorithm is defined as a method that makes use of genetic relatedness of the eight neighbouring experimental units found in a 3 × 3 matrix using information provided by a numerator relationship matrix (**A**) of the corresponding genotypes. Steps for this algorithm are: 1) randomly generate *m* initial designs and select the best (Ωi) with the smallest trace, ; 2) randomly select a treatment from Ωi; 3) identify the genetic correlation coefficients from the numerator relationship matrix for all experimental units within the nearest neighborhood of ; 4) if there exists a pairwise genetic relationship of 0.25 or higher between and any other treatment for *l* ≠ *k* within the neighbouring matrix; 5) replace either one of the treatments with a another treatment that is at a distance of more than a unit (row or column) away; 6) if there are no treatments further than a unit away even though these neighbours are genetically correlated, randomly interchange with ; 7) calculate the new criterion value, , based on the new design layout Ωj; 8) if , accept , otherwise reject Ωj; and 9) repeat steps 2 to 8 for a total of *p* iterations. Note that if all the experimental units from a neighbourhood are genetically unrelated, then the SP is applied.

SA is a probabilistic meta-heuristic and stochastic optimization procedure that prevents the search from getting trapped in a local optima by accepting some solutions with a set probability and lowering the temperature with time to make sure that poorer solutions are accepted with lower probabilities. The SA algorithm implemented in this study is described as follows: 1) randomly interchange a pair of treatments within a randomly selected block to produce a new layout, and re-calculate a criterion value, τ*j*; 2) if τ*i >* τ*j*, accept as the new layout with probability 1.0; else do the following, 3) calculate △ =τ*j* −τ*i* and set a cooling temperature *Tc*[*i*]= 1*/i*, for each *i*-th iteration, and calculate *v* =exp(−△*/Tc*[*i*]); 4) draw a random value *u* from a uniform distribution, and if *u < v* accept ; and 5) repeat steps 1 to 4 for a total of *p* iterations.

## 2.3 Evaluation of algorithms

The above algorithms were evaluated under varying experimental conditions to assess their effectiveness to improve a RCB design. For the greedy algorithm, a total of 4, 14 and 98 random pairs of swaps were performed simultaneously within a given random block. These are identified as GP4, GP14 and GP98. Conditions considered include narrow-sense heritabilities, *h*2, of 0.1, 0.3, and 0.6, where *h*; unrelated individuals (Independent), half-sib and full-sib families; and a spatial correlation of ρ = 0*.*6. Every combination of conditions was repeated λ = 10 times for *p* = 5*,*000 iterations. All implementation and evaluation of algorithms was done using *R* (R Core Team, 2018)*.*

The following scenarios were considered: and represent RCB designs with 30 genotypes generated using *A*- and *D*-optimality criteria, respectively. In these layouts, the designs had six blocks each of dimensions five rows by six columns. Here, pedigree from half-sib families consisted of five male parents each with six individuals, and full-sib families consisted on a half-diallel with five parents for a total of 10 families each with three individuals. Scenario, represents an RCB design with 196 genotypes generated using *A*-optimality criterion with four blocks of dimensions 14 rows by 14 columns per block. Pedigree files for half-sib families had 32 known parents each with six offspring, whereas full-sib families had 30 parents with several half-diallels for a total of 68 families each with approximately three offspring.

Note that, GP98 was implemented only for scenario to swap 50 % of the total genotypes at every single iteration, and GP14 represents swapping about 50 % for and scenarios. A motivating example was evaluated for all algorithms to investigate the level of design efficiencies and rates of convergence that can be obtained for a specified condition with all algorithms having to improve the same initial RCB experimental design. This was done using *A*-optimality criterion for an experiment with 30 genotypes, 6 blocks of sizes 5 rows and 6 columns, and comprised of half-sib families with five male parents each with six individuals, for *h*2 = 0*.*1, ρ= 0*.*6, and a nugget effects of 0.1. Initially, *m* = 1*,*000 designs were randomly generated and the best one selected for optimization. All the proposed algorithms were made to improve this initial design by going through *p* = 20*,*000 iterations. Traces from both successful and unsuccessful swaps were observed together with the time taken for each algorithm. A swap was defined to be successful if the resulting design had a smaller trace than the previous as this translates to a reduction in average variance of the treatment effects (Das, 2002). The motivating example was run from a 64-bit windows operating system Intel(R) Core(TM) i7-4720HQ CPU@2.60GHz, RAM = 8.0GB.

In order to evaluate the improvement of a design, a relative overall design efficiency (ODE %), that quantifies how efficient the improved design is relative to an initially non-improved design was calculated as:

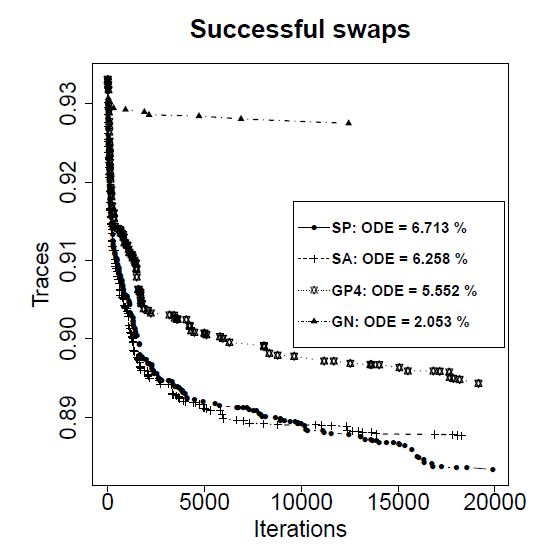
*where i* = 1, 2, 3, ---; where is the total number of conditions to be evaluated and with condition *i* is replicated *j* times, for *j* = 1, 2, 3, --- , and are averages of *m* initial traces and log(determinants), respectively, for *i*-th condition and *j*-th replicate, and are the smallest trace and log(determinant), respectively, obtained from an improved design. Estimates of ODE% over the λ = 10 replicates per condition were summarized.

**3. Results**

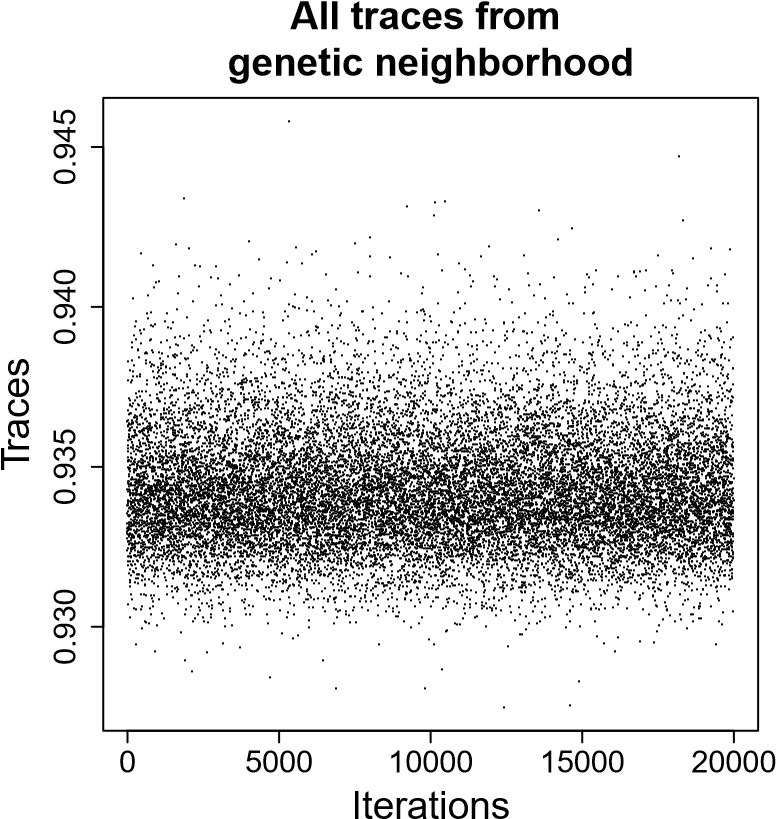
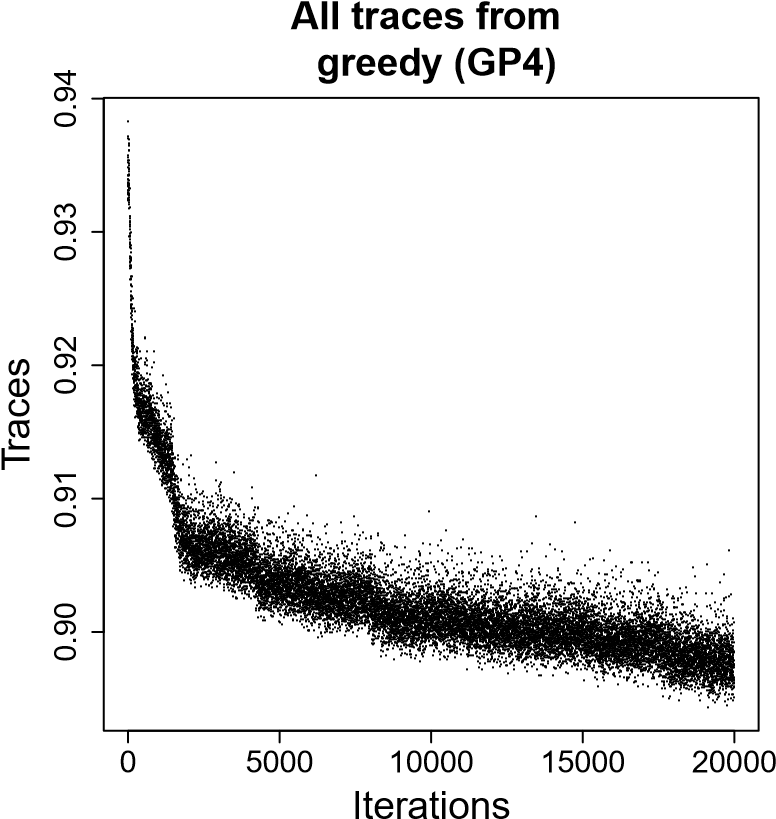
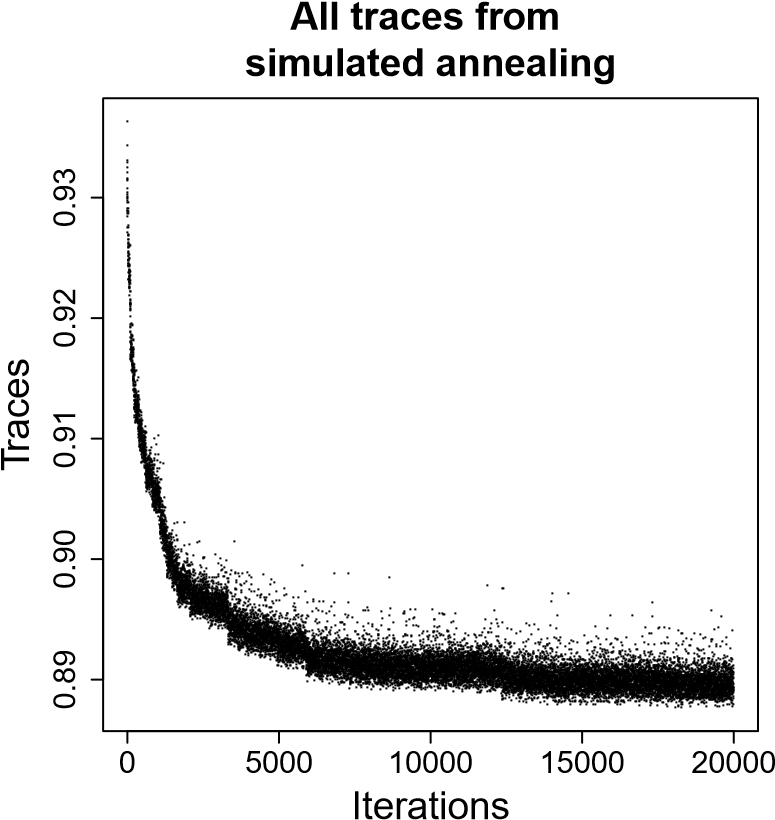
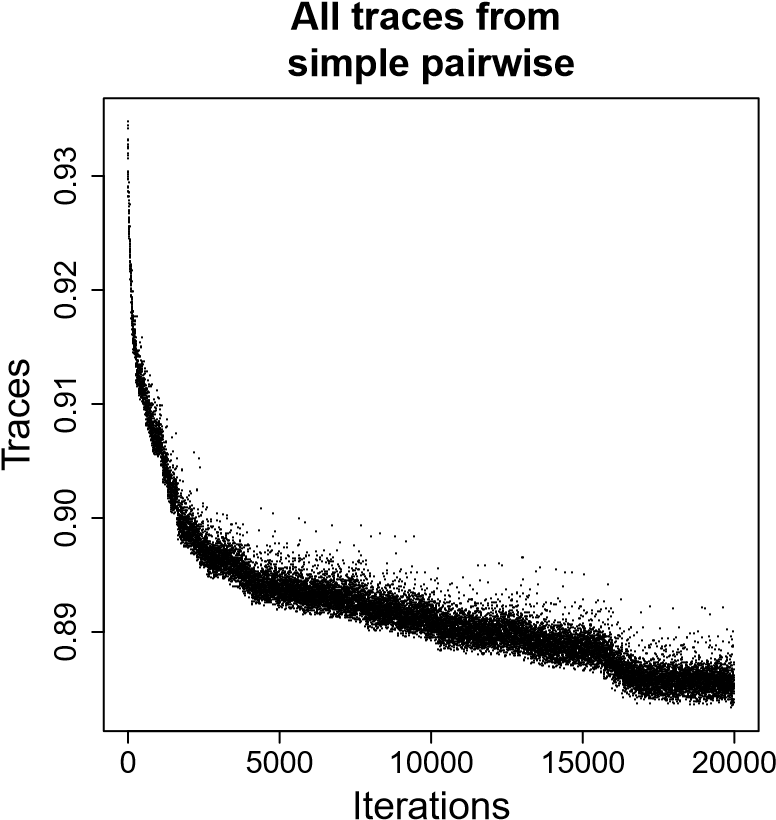
Results from the motivating example that was conducted for an RCB design with *h*2 = 0*.*1 and ρ= 0*.*6 based on with a nugget effect of 0.1 for an experiment with 6 blocks of 5 rows by 6 columns, are displayed in Figure 1 which plots traces obtained from successful swaps and their overall design efficiencies and Figure 2 showing the rate of convergence by plotting all the 20,000 traces obtained for each algorithm. From this illustration, the results indicate that simple pairwise (SP) algorithm had the highest design efficiency of 6.713% with the highest number of successful swaps = 192 and took about 5.8 minutes for the 20,000 iterations. This was closely followed by the simulated annealing (SA) algorithm that had an ODE = 6.258% with 139 successful swaps and took about 5.8 minutes. GP4 algorithm had an ODE = 5.552% with 104 successful swaps and also took about 5.8 minutes and genetic neighbourhood (GN) algorithm recorded the lowest ODE = 2.053% with 12 successful swaps and took about 6.1 minutes.

Means and standard errors (S.E.) of overall design efficiency (ODE %) for the three scenarios, that is, , and for all algorithms are presented in Tables 1, 2 and 3, respectively. Figure 3 display visible trends of ODEs by genetic relatedness and heritability levels whereas Figure 4 shows the average number of successful swaps out of 5,000 (that is, swaps that were accepted due to the resulting design having a smaller criterion value than the previous) for each algorithm. Results indicate that for all experiments conducted based on and scenarios, simulated annealing (SA) and simple pairwise (SP) algorithms achieved the highest ODE means in all evaluated conditions followed by GP4 (for ) or GP98 (for ) and lowest for genetic neighbourhood (GN). Also, the overall highest ODEs were achieved when *h*2 = 0*.*3 among genetically unrelated individuals for all algorithms. Among full-sib families, highest ODEs were achieved when *h*2 = 0*.*1 and decreased with increasing heritability for all algorithms evaluated under and scenarios. SA recorded the highest average ODE = 7.403 % (S.E. = 0.063) followed by SP with average ODE = 7.398 % (S.E. = 0.066) all obtained when *h*2 = 0*.*3 among genetically unrelated individuals. Algorithms SA, SP, GP4 and GP14 evaluated with half-sib families under had highest ODEs obtained for treatments with lowest heritability of 0.1, whereas GN achieved its highest ODE when *h*2 = 0*.*3 for the same family.

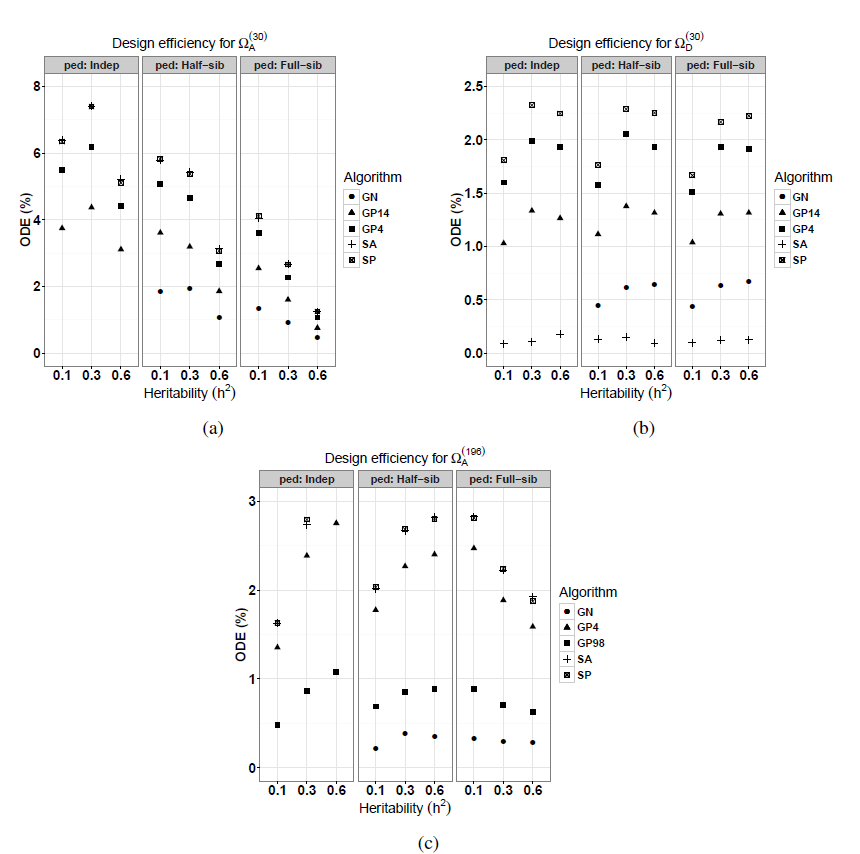
Based on scenario, the best performing algorithm with highest average ODE among all conditions was SP, closely followed by GP4, GP14, GN and SA which recorded the lowest average ODE. Under this scenario, the overall highest ODEs were observed among genetically unrelated individuals for SP, GP4, and GP14 when *h*2 = 0*.*3. Among half-sib families, highest ODEs occurred when *h*2 = 0*.*3 but no clear trend among full-sib families. Both and took, on average, about 2 to 3 minutes to improve a given initial experimental design for *p* = 5*,*000 iterations whereas required about 25 to 40 minutes for the same number of iterations. Figure 4 indicate that the number of successful swaps decrease with increasing heritability especially for and scenarios with small difference in numbers between SA and SP algorithms but presents with large differences under scenario. The number of successful swaps out of 5,000 appeared to be highest for SA and SP under *A*-optimality criterion. From scenario, the number of successful swaps were highest for SA which recorded above 2,500 out of the 5,000 swaps but this was not realized in terms of improving the design efficiency under this criterion.



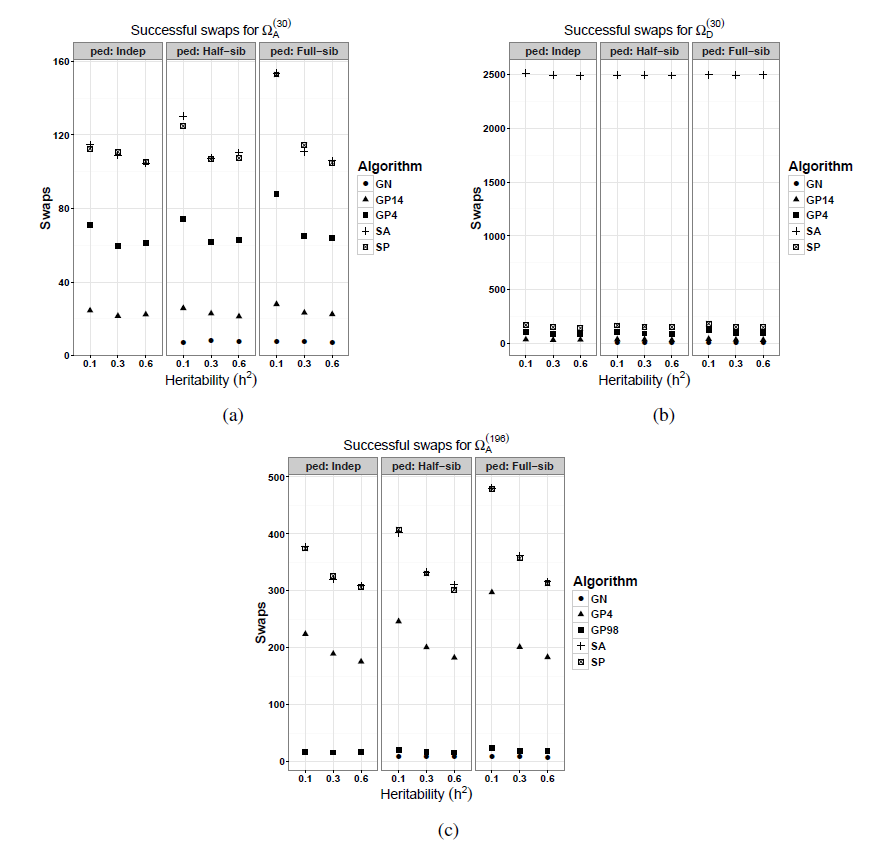
**Figure 1**. A motivating example displaying the traces from successful swaps to convey the rate of convergence for simple pairwise (SP), simulated annealing (SA), greedy pairwise (GP4), and genetic neighborhood (GN) algorithms with their overall design efficiencies (ODE) evaluated for half-sib families with h2 = 0.1, ρ= 0.6 with a nugget error of 0.1 iterated for 20,000.



**Figure 2.** A motivating example illustrating the rates of convergence of algorithms showing all traces obtained from these algorithms evaluated for half-sib families with *h*2 = 0*.*1, ρ= 0*.*6 with a nugget error of 0.1 iterated for 20,000.



**Figure 3.** Overall design efficiency (ODE %) for (a) (b), and (c) scenarios evaluated for simple pairwise (SP), greedy pairwise: GP4, GP14, and GP98, simulated annealing (SA) and genetic neighbourhood (GN) algorithms iterated *p* = 5*,*000 times, with each condition replicated λ= 10 times, with *m* = 100 initially unimproved designs and *s* = 1 selected design.



**Figure 4.** Average swaps for scenarios (a) (b), and (c) based on simple pairwise (SP), greedy pairwise: GP4, GP14, and GP98, simulated annealing (SA) and genetic neighbourhood (GN) algorithms.

**Table 1.** Average (and standard errors) of algorithms overall design efficiencies (ODE %) for RCB experimental designs at a spatial correlation of 0*.*6. Average ODE % from 10 replicates per condition are reported together with standard errors (S.E.) for simple pairwise (SP), greedy pairwise (GP4) and GP14, simulated annealing (SA) and genetic neighbourhood (GN) procedures.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Pedigree | h2 | SP | GP4 | GP14 | SA | GN |
|  | 0.1 | 6.347 (0.060) | 5.501 (0.060) | 3.747 (0.093) | 6.385 (0.072) | - |
| Indep. | 0.3 | 7.398 (0.066) | 6.194 (0.080) | 4.371 (0.053) | 7.403 (0.063) | - |
|  | 0.6 | 5.109 (0.044) | 4.414 (0.057) | 3.110 (0.054) | 5.222 (0.064) | - |
|  |  |  |  |  |  |  |
|  | 0.1 | 5.826 (0.026) | 5.082 (0.055) | 3.610 (0.065) | 5.781 (0.045) | 1.853 (0.042) |
| Half-sib | 0.3 | 5.375 (0.056) | 4.640 (0.082) | 3.192 (0.052) | 5.428 (0.047) | 1.940 (0.088) |
|  | 0.6 | 3.066 (0.028) | 2.663 (0.023) | 1.858 (0.033) | 3.131 (0.028) | 1.064 (0.033) |
|  |  |  |  |  |  |  |
|  | 0.1 | 4.109 (0.030) | 3.611 (0.026) | 2.543 (0.038) | 4.045 (0.027) | 1.343 (0.034) |
| Full-sib | 0.3 | 2.656 (0.029) | 2.265 (0.021) | 1.601 (0.034) | 2.667 (0.032) | 0.920 (0.027) |
|  | 0.6 | 1.247 (0.006) | 1.065 (0.009) | 0.755 (0.012) | 1.247 (0.013) | 0.460 (0.011) |

**Table 2.** Average (and standard errors) of algorithms overall design efficiencies (ODE %) for RCB experimental designs at a spatial correlation of 0*.*6. Average ODEs are reported together with standard errors (S.E.) for simple pairwise (SP), greedy pairwise: GP4 and GP98, simulated annealing (SA) and genetic neighbourhood (GN) procedures.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Pedigree | h2 | SP | GP4 | GP14 | SA | GN |
|  | 0.1 | 1.633 (0.013) | 1.354 (0.018) | 0.481 (0.008) | 1.629 (0.015) | - |
| Indep. | 0.3 | 2.794 (0.020) | 2.387 (0.017) | 0.864 (0.024) | 2.736 (0.034) | - |
|  | 0.6 | 3.232 (0.024) | 2.754 (0.039) | 1.080 (0.028) | 3.270 (0.027) | - |
|  |  |  |  |  |  |  |
|  | 0.1 | 2.032 (0.023) | 1.776 (0.019) | 0.690 (0.018) | 2.016 (0.019) | 0.216 (0.014) |
| Half-sib | 0.3 | 2.684 (0.018) | 2.269 (0.009) | 0.851 (0.024) | 2.670 (0.019) | 0.381 (0.013) |
|  | 0.6 | 2.801 (0.027) | 2.402 (0.029) | 0.890 (0.025) | 2.818 (0.009) | 0.351 (0.020) |
|  |  |  |  |  |  |  |
|  | 0.1 | 2.813 (0.018) | 2.471 (0.022) | 0.888 (0.025) | 2.827 (0.014) | 0.324 (0.015) |
| Full-sib | 0.3 | 2.240 (0.016) | 1.886 (0.023) | 0.702 (0.020) | 2.226 (0.021) | 0.297 (0.011) |
|  | 0.6 | 1.873 (0.011) | 1.588 (0.013) | 0.623 (0.016) | 1.926 (0.013) | 0.280 (0.013) |

**Table 3.** Average (and standard errors) of algorithms overall design efficiencies (ODE %) for RCB experimental designs at a spatial correlation of 0.6. Each condition was replicated 10 times with their average ODEs reported together with standard errors (S.E.) for simple pairwise (SP), greedy pairwise: GP4 and GP14, simulated annealing (SA) and genetic neighbourhood (GN) procedures.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Pedigree | h2 | SP | GP4 | GP14 | SA | GN |
|  | 0.1 | 1.807 (0.014) | 1.600 (0.014) | 1.029 (0.014) | 0.085 (0.019) | - |
| Indep. | 0.3 | 2.324 (0.017) | 1.993 (0.013) | 1.335 (0.030) | 0.104 (0.025) | - |
|  | 0.6 | 2.247 (0.021) | 1.930 (0.021) | 1.265 (0.032) | 0.178 (0.027) | - |
|  |  |  |  |  |  |  |
|  | 0.1 | 1.766 (0.012) | 1.576 (0.015) | 1.115 (0.015) | 0.130 (0.034) | 0.446 (0.015) |
| Half-sib | 0.3 | 2.287 (0.023) | 2.054 (0.024) | 1.377 (0.022) | 0.150 (0.041) | 0.614 (0.013) |
|  | 0.6 | 2.253 (0.024) | 1.933 (0.020) | 1.315 (0.019) | 0.090 (0.023) | 0.637 (0.023) |
|  |  |  |  |  |  |  |
|  | 0.1 | 1.666 (0.011) | 1.514 (0.013) | 1.037 (0.009) | 0.097 (0.025) | 0.431 (0.010) |
| Full-sib | 0.3 | 2.168 (0.013) | 1.935 (0.025) | 1.307 (0.026) | 0.119 (0.025) | 0.634 (0.017) |
|  | 0.6 | 2.225 (0.027) | 1.913 (0.023) | 1.316 (0.025) | 0.125 (0.019) | 0.669 (0.022) |

# 4. Discussion

Algorithms are used in research for instance, to optimize long term forest planning management using simulated annealing (Borges, Eid and Bergseng, 2014), to estimate the optimum combination of stand paths for a given forest (Seo et al., 2005) and Liu et al., (2006) to optimize spatially constrained harvest scheduling problems in forest planning and management. It has been applied in the current study to assess how well it can be used to improve the efficiency of experimental designs. In this study, evaluation of algorithm efficiency to improve experimental designs has focused on the use of RCB designs in field trials with application in plant breeding. Presence of half-sib or full-sib families in experiments require appropriate modelling of their genetic correlations and similarly, proximity of genotypes within rows and columns needs to be accounted for as those genotypes in close range may share microsite and thus, accounting for spatial correlations within rows and columns is necessary to minimize experimental error bias. Incorporating spatial Correlations in a RCB design has been shown by (Gezan et al., 2010) to produce designs that are nearly as efficient as those generated using a row-column designs with uncorrelated residual errors. The current study examines potential design efficiency levels that can be achieved when simple pairwise (SP), greedy pairwise algorithms denoted as GP4, GP14, and GP98, simulated annealing (SA) and genetic neighbourhood (GN) algorithms in plant breeding programs.

From the motivating example that examined a specific condition where an RCB experiment with half-sib that had *h*2 = 0*.*1 at ρ = 0*.*6 with a nugget error of 0.1 and iterated for 20,000 to improve a design, results indicate that the simple pairwise (SP) algorithm is the best as it managed to improve the initial experiment by reducing the average variance of treatment effects by 6.713 %. SA followed closed with an ODE of 6.258 %, not much different from that of SP. The more aggressive algorithms such as GP4, are unlikely to perform better than SP under the evaluated experimental conditions presented in this study with GN achieving the lowest design improvement levels. Also, GN had only 12 successful swaps which is a much smaller number than SP and SA who recorded 192 and 139 respectively. The rates of convergence as shown in Figure 2 is better for SP, SA and GP4 than for GN as its traces are randomly scattered.

Results from Tables 1, and 2, have shown that simple pairwise and simulated annealing algorithms achieves the highest relative design efficiencies under all experimental conditions for and scenarios with second best algorithm appearing to be greedy pairwise (GP4) followed by GP14 for Ω(*A*30) scenario or GP98 for scenario and last by genetic neighborhood. These results could be attributable to the fact that simple pairwise procedure swaps a single pair of treatments per iteration, thus takes small steps in the search for optimality which makes it more likely to find an optimal condition than greedy algorithms that take large steps. Simulated annealing performed well under *A*-optimality criterion since it has the ability not to be trapped in a local minima by accepting a proportion of bad solutions using an exponential distribution and a cooling schedule. However, simulated annealing algorithm achieved lowest relative design efficiencies for the same number of iterations of 5,000 under scenario as shown in Table 3. It is not very clear why this is so, but it was observed that it accepted too many bad solutions as it tried not to get trapped in a local minima, that did not subsequently maximize the objective function. Large design improvements have been observed among genetically unrelated individuals, which agrees with findings from (Bueno Filho and Gilmour, 2003) although they did not analyse varied levels of spatial correlations. Optimization based on *A*-criterion has revealed from this current study that a substantial decrease in average variance of treatment effects among full-sib families can be realized for treatments with very small narrow-sense heritabilities (*h*2 = 0*.*1). When full-sibs have strong narrow-sense heritabilities such as 0*.*6 at a spatial correlation of 0*.*6, little improvements on the design efficiencies can be achieved. For experimental designs that were evaluated under scenario, the amount of design improvement was, for some conditions, about four times larger than that realized under scenario. This is is because more iterations (*>* 50*,*000) are required for larger experiments to converge to an optimal solution than it would take a smaller experiment. The number of successful swaps displayed in Figure 4 indicate that they decrease with increasing heritability for all families for experiments evaluated under and scenarios for almost all algorithms.

The choice of *A*- or *D*-optimality criteria depends on the objective function to be maximized or minimized. Both criteria are a convex function of the eigenvalues of an information matrix (Das, 2002; Kuhfeld, 2010) since *A*-optimality is a function of the arithmetic mean of the eigenvalues whereas *D*-optimality is a function of the geometric mean of eigenvalues (Kuhfeld, 2010). Thus, an increase in overall design efficiency implies a decrease in the average variances of the random treatment effects. When matrices are sparse, it is not efficient to use the *D*-optimality criterion since the determinants are likely to be zeros and this may cause computational problems. The approach in this study used the natural logarithms of the determinants, although this did not solve the problem for large experiments such as the scenarios. For these reasons, the authors would recommend *A*-optimality procedure. If approximations to the procedure are required, then, a similar approach to that described by (Butler, Eccleston and Cullis, 2008) can be used.

The procedure presented in this study can be easily extended to other complex experimental designs such as non-orthogonal experiments that can be implemented with appropriate statistical models and an optimality criterion of choice. Other variants of the search algorithms can also be implemented. For instance, for genetic neighbourhood procedure, a value different from 0.25 could be chosen to indicate which treatments to be swapped. In the implemented GN procedure, any two neighbouring treatments that had a genetic relationship coefficient of 0.25 (for half-sib) or more were swapped. It is not known whether changing this value to a higher coefficient would increase the efficiency of GN algorithm.

In summary, the potential to improve experimental designs, particularly randomized complete block designs, has been shown, in this study, to be highest when simulated annealing and simple pairwise algorithms are used under *A*-optimality criteria, in which case, they also achieve the highest numbers of successful swaps. Similarly, under *D*-optimality criterion, simple pairwise records highest overall design efficiencies whereas simulated annealing performs poorest with the largest number of accepted swaps. In conclusion, the use of a simple pairwise algorithm based on *A*-optimality criterion under a linear mixed model framework to improve RCB experimental designs is desirable.

# References

Borges, P., Eid, T. and Bergseng, E. (2014) Applying simulated annealing using different methods for the neighborhood search in forest planning problems. *European Journal of Operational Research*, **233**, 700–710.

Bueno Filho, J.S.D.S. and Gilmour, S.G. (2003) Planning incomplete block experiments when treatments are genetically related. *Biometrics*, **59**, 375–381.

Butler, D.G., Eccleston, J.A. and Cullis, B.R. (2008) On an approximate optimality criterion for the design of field experiments under spatial dependence. *Australian and New Zealand Journal of Statistics*, **50**, 295–307.

Butler, D.G., Smith, A.B. and Cullis, B.R. (2014) On the Design of Field Experiments with Correlated Treatment Effects. *Journal of Agricultural, Biological, and Environmental Statistics*, **19**, 539–555.

Chernoff, H. (1953) Locally optimal designs for estimating parameters. *The Annals of Mathematical Statistics*, **24**, 586–602.

Cullis, B.R., Lill, W., Fisher, J., Read, B. and Gleeson, A. (1989) A new procedure for the analysis of early generation variety trials. *Journal of Applied Statistics*, **38**, 361–375.

Cullis, B.R., Smith, A.B. and Coombes, N.E. (2006) On the design of early generation variety trials with correlated data. *Journal of Agricultural, Biological and Environmental Statistics*, **11**, 381–393.

Das, A. (2002) An introduction to optimality criteria and some results on optimal block design. *Design Workshop Lecture Notes* pp. 1–21. Indian Statistical Institute, Kolkata, Theoretical Statistics and Mathematics Unit, New Delhi, India.

Gezan, S.A., White, T.L. and Huber, D.A. (2010) Accounting for spatial variability in breeding trials: A simulation study. *Agronomy*, **102**, 1562–1571.

Gilmour, A.R., Gogel, B.J., Cullis, B.R. and Thompson, R. (2009) ASReml user guide release 3.0. *VSN International Ltd*.

Harville, D.A. (1997) *Matrix Algebra from a Statistician’s Perspective*. New York, Inc, Springer-Verlag, USA.

Henderson, C.R. (1950) The estimation of genetic parameters. *The Annals of Mathematical Statistics*, **21**, 309–310.

Hooks, T., Marx, D., Kachman, S. and Pedersen, J. (2009) Optimality Criteria for Models with Random Effects. *Revista Colombiana de Estadistica*, **32**, 17–31.

John, J.A. and Williams, E.R. (1995) *Cyclic and Computer Generated Designs*, Second. Chapman and Hall, UK.

Kirkpatrick, S., Gelatt, C.D. and Vecchi, M.P. (1983) Optimization by simulated annealing. *Science*, **220**, 671–680.

Kuhfeld, W.F. (2010) *Experimental Design: Effiency, Coding, and Choice Designs*.

Liu, G., Han, S., Zhao, X., Nelson, J.D., Wang, H. and Wang, W. (2006) Optimisation algorithms for spatially constrained forest planning. *Ecological Modelling*, **194**, 421–428.

Mramba, L.K., Peter, G.F., Whitaker, V.M. and Gezan, S.A. (2018) Generating Improved Experimental Designs with Spatially and Genetically Correlated Observations Using Mixed Models. *Agronomy*, **8**.

Piepho, H.P., Möhring, J., Melchinger, A.E. and Büchse, A. (2008) {BLUP} for phenotypic selection in plant breeding and variety testing. *Euphytica*, **161**, 209–228.

R Core Team. (2018) R: A Language and Environment for Statistical Computing.

Seo, J.-H., Vilčko, F., Orois, S.S., Kunth, S., Son, Y.-M. and Gadow, K. Von. (2005) A case study of forest management planning using a new heuristic algorithm. *Tree Physiology*, **25**, 929–938.

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

Welham, S.J., Gezan, S.A., Clark, S.J. and Mead, A. (2015) *Statistical Methods in Biology*. Chapman & Hall, Boca Raton, USA.

Williams, E.R., John, J.A. and Whitaker, D. (2006) Construction of Resolvable Spatial Row-Column Designs. *Biometrics*, **62**, 103–108.