

Augmented p-rep designs

Emlyn Williams^{*,1}, Hans-Peter Piepho², and David Whitaker³

¹ Statistical Consulting Unit, The Australian National University, Canberra 0200, Australia

² Bioinformatics Department, Institute for Crop Science, University of Hohenheim, 70593 Stuttgart, Germany

³ Ranfurly, RD2, South Island, New Zealand

Received 2 May 2010, revised 14 September 2010, accepted 27 September 2010

Early generation variety trials are very important in plant and tree breeding programs. Typically many entries are tested, often with very little resources available. Unreplicated trials using control plots are popular and it is common to repeat the trials at a number of locations. An alternative is to use p-rep designs, where a proportion of the test entries are replicated at each location; this can obviate the need for control plots. α -Designs are commonly used for replicated variety trials and we show how these can be adapted to produce efficient p-rep designs.

Key words: α -Designs; Augmented designs; p-rep designs.

1 Introduction

Federer (1961) proposed augmented designs for trials where entries cannot be replicated, but controls are available for local error control. The basic approach is to choose some replicated blocked design for the controls and then to augment the blocks with unreplicated entries. Using a classical incomplete block analysis of variance, the controls then allow for an estimate of error and computation of adjusted means correcting for incomplete block effects. Extensions have been proposed for blocking in rows and columns (Federer *et al.*, 1975; Federer, 2002; Williams and John, 2003). This kind of design has the desirable feature that entries need not be replicated, but suffers from the fact that a considerable proportion of plots may need to be devoted to controls.

An attractive alternative to the use of controls is to replicate a certain proportion of the entries of interest for local error control. Recently, Smith *et al.* (2006) and Cullis *et al.* (2006) have proposed such designs, which they denoted as partially replicated (p-rep) designs. These designs are particularly useful when trials are repeated across locations. When there are l locations, one can replicate $1/l$ -th of entries at a location such that each entry is tested with two replicates in one of the l locations.

This paper considers a combination of both approaches. The key idea is to use Federer's augmented designs, replacing the controls with partially replicated entries. Thus, for each location one identifies the subset of entries to be partially replicated on two plots each and determines a resolvable incomplete block design with two replicates for this subset. The blocks of this design are then augmented by the unreplicated entries.

For example, a breeder collaborating with the second author (Andres Gordillo, AgReliant Genetics, personal communication) routinely performs trials evaluating the testcross performance of doubled haploid lines derived from different biparental breeding crosses of maize within a heterotic group, where the test crosses are made with testers from an opposite heterotic group. The

*Corresponding author: e-mail: Emlyn.Williams@anu.edu.au, Phone: +61-2-6125-0562, Fax: +61-2-6125-4984

number of entries per trial typically ranges between 100 and 150, and the trials are conducted at six locations. So far, the trial design used at a location has been an unreplicated design with between 8.3 and 12.5% of plots allocated to check varieties. In order to save resources and improve its efficiency, this breeder is now trying augmented p-rep designs for six locations with block size eight.

The main challenge in finding good augmented p-rep designs is how best to do the augmentation. Care needs to be taken that the number of pairwise concurrences, *i.e.* the number of times two entries concur (appear in the same block), remains as balanced as possible across pairs of entries. We show how α -designs (John and Williams, 1995) can be adapted to produce p-rep designs where at each location there is a two-replicate resolvable block design for a proportion of the entries and overall pairs of entries concur no more than once.

2 Methods to construct augmented p-rep designs

2.1 Construction using α -designs

It will be assumed that a trial is to be replicated across several locations. The following notation will be used:

v = No. of entries.

l = No. of locations.

m = Block size of the resolvable incomplete block design for the replicated entries.

s = Number of blocks per replicate at a location.

k = The final size of incomplete blocks (including replicated entries).

Suppose we have an α -design with $v = k's$ entries and $r = 2l$ replicates where $k' = lm$. This can be generated from a $k' \times r$ α -array (John and Williams, 1995). The k' rows of the α -array generate k' subsets of the entries and at each location, m subsets will be duplicated.

Without loss of generality, we can assume that the first two columns of the α -array apply to location 1, the next two for location 2 and so on. Furthermore, we will assume that the first m rows of the α -array correspond to the subsets of entries to be duplicated at location 1, the next m for location 2 and so on. By fixing these $m \times 2$ rectangles down the diagonal direction of the α -array, we can now drop out half of the remaining $k - m$ elements from each column of the α -array. This leaves an array with missing elements, which we call a partial α -array or p- α -array; we use this to generate a p-rep design with block size

$$k = (l+1)m/2 \quad (1)$$

in a manner similar to the generation of α -designs (John and Williams, 1995).

Example 1. Consider an α -design for $r = 4$ replicates and $v = 20$ entries in incomplete blocks of size $k' = 4$, *i.e.* $s = 5$. A reduced α -array (John and Williams, 1995, Section 4.4.2) is

0	0	0	0
0	1	2	3
0	2	4	1
0	3	1	4

(2)

This array generates the unrandomized α -design (John and Williams, 1995, Section 4.4.1) given in Table 1.

We associate replicates 1 and 2 with location 1 and replicates 3 and 4 with location 2. Then to duplicate entries 1–10 at location 1 and entries 11–20 at location 2 we fix the elements of (2) enclosed in boxes. We then remove one element from the remaining two elements in each column of (2), *e.g.* the numbers in bold, to give the p-rep design in Table 2.

Table 1 Unrandomized α -design for 20 entries in blocks of size 4.

Rep 1	1	2	3	4	5
	6	7	8	9	10
	11	12	13	14	15
	16	17	18	19	20
Rep 2	1	2	3	4	5
	7	8	9	10	6
	13	14	15	11	12
	19	20	16	17	18
Rep 3	1	2	3	4	5
	8	9	10	6	7
	15	11	12	13	14
	17	18	19	20	16
Rep 4	1	2	3	4	5
	9	10	6	7	8
	12	13	14	15	11
	20	16	17	18	19

Table 2 Unrandomized p-rep design for 20 entries in blocks of size 3.

Location 1	1	2	3	4	5	1	2	3	4	5
	6	7	8	9	10	7	8	9	10	6
	11	12	13	14	15	19	20	16	17	18
Location 2	1	2	3	4	5	9	10	6	7	8
	15	11	12	13	14	12	13	14	15	11
	17	18	19	20	16	20	16	17	18	19

2.2 Choosing p- α -arrays

If possible we want to ensure that entries appear together within incomplete blocks at most once. Designs with pairwise concurrences of zero or one are called (0,1)-designs. It is conjectured that for fixed basic parameters, *i.e.* r , s , k' and m , designs that maximize the average efficiency factor, E of the design will come from the class of (0,1)-designs (John and Williams, 1982). A necessary, though not sufficient, condition for the existence of an $\alpha(0,1)$ -design is that $k' \leq s$ (John and Williams, 1995, Section 4.5). In the construction of p-rep designs, if we start with an $\alpha(0,1)$ -design, then any resulting p-rep design will necessarily be a (0,1)-design. When $k' \leq s$ the software package CycDesign (Whitaker *et al.*, 2009) will usually generate an $\alpha(0,1)$ -design which can be used to obtain a suitable p-rep design.

When $k' > s$ the best we can do is start with an $\alpha(0,1,2)$ -design. Then more care is required when removing elements from an α -array in order to achieve a p-rep (0,1)-design. An effective way of proceeding is by calculating differences on the elements of the α -array as discussed by John and

Williams (1995, Section 4.4.3). Essentially if a set of row (or column) differences (modulo s) contain repeated elements then this results in pairwise concurrences of two in the design and we want to eliminate these. The method is best described using an example.

Example 2. Consider the parameters $k' = 4$, $s = 3$, $r = 4$ and $m = 2$ leading to a design with $v = 12$ entries. From CycDesign, we obtain the following α -array (written in a reduced form):

$$\begin{array}{|c|c|c|c|} \hline 0 & 0 & 0 & \mathbf{0} \\ \hline 0 & 2 & \mathbf{1} & 0 \\ \hline 0 & \mathbf{0} & 1 & 2 \\ \hline \mathbf{0} & 1 & 2 & 1 \\ \hline \end{array} \quad (3)$$

Since $k' > s$, this will generate an $\alpha(0,1,2)$ -design. As with (2), we will assume (without loss of generality) the 2×2 boxed elements down the diagonal are fixed and are the components of the α -array that generate the two-replicate designs at each of the two locations. Taking differences with columns (modulo 3) we have.

Column 3–1	Column 4–1	Column 3–2	Column 4–2
0	0	0	0
1	0	2	1
1	2	1	2
2	1	1	0

Whenever there is a number repeated in a column of the above difference table, this means that there are pairs of entries that concur in blocks twice, *i.e.* there is a 2 in the concurrence matrix of the starting $\alpha(0,1,2)$ -design. But by targeted deletion of elements in the α -array, it is usually possible to eliminate repeat elements in the columns of the difference table and hence concurrences of two in the resulting p-rep design. For example deleting the elements in bold in the α -array (3) will remove the bolded elements in the difference array and so the p-rep design will be a (0,1)-design. Note that there is no need to consider differences between columns 1 and 2 since we have assumed that the first two columns relate to location 1; hence the only thing to ensure there is that elements are deleted evenly over the non-boxed rows of the α -array. Similarly for columns 3 and 4.

The process of operating on differences within an α -array in order to generate a p-rep (0,1) design can be easily automated. Note that for equal block sizes k there are some combinations of m and l for which p-rep designs cannot be constructed as seen from Table 3. This is because from (1) either

Table 3 Augmented block size k for different values of l and m .

l	k			
	$m = 2$	$m = 3$	$m = 4$	$m = 5$
2	3	–	6	–
3	4	6	8	10
4	5	–	10	–
5	6	9	12	15
6	7	–	14	–
7	8	12	16	20
8	9	–	18	–

m has to be even or l needs to be odd. There is, however, no need for incomplete block sizes to be the same in a p-rep design; block sizes differing by one plot are very common in variety trial design (John and Williams, 1995, Section 4.9) and the same flexibility is possible with the construction of p-rep designs.

2.3 Criteria of design optimality

We propose to assess efficiency E based on a linear model with main effects for blocks and entries

$$y_{ij} = \mu + \tau_i + \beta_j + \varepsilon_{ij}, \quad (4)$$

where y_{ij} is the response of the i -th entry ($i = 1, 2, \dots, v$) in the j -th block ($j = 1, 2, \dots, 2sl$) and error terms ε_{ij} are uncorrelated random variables each with mean zero and variance σ^2 . The overall efficiency factor is defined as

$$E = (v - 1) / \sum_{i=1}^{v-1} e_i^{-1}$$

where e_i ($i = 1, \dots, v$) are the canonical efficiency factors (John and Williams, 1995, p.31).

Notice that blocks in our design are split among several locations. Model (4) with main effects only ignores entry–location interaction, which constitutes a simplification. If location is a random factor, entry–location interaction is a random effect. If all entries were unreplicated for all entries in all locations, the interaction variance would be completely confounded with error, so the approach would be exact. With partial replication, the approach constitutes an approximation. A more appropriate efficiency assessment would specify an *a priori* estimate of the ratio of error and interaction variance and employ a model with effects for entries, locations, blocks within locations, and interaction, taking all effects except interaction and error fixed. Here, we consider the case where efficiency is based on a model with just main effects for blocks and entries. This has several advantages: (i) there is no need to specify *a priori* values of variance components; (ii) there is no need to specify a variance–covariance model for interaction (which might be complex; see Denis *et al.*, 1997, and Piepho, 1997, 1998); (iii) a computer search is much less demanding.

A further argument for setting interaction to zero at the design stage is as follows. When interaction is large relative to trial error variance then the design per location becomes less important because standard errors of pairwise differences are dominated by the interaction variance, and BLUE of entry means are essentially equivalent to simple means of adjusted means per trial. Conversely, the design matters most when the interaction variance is small relative to the error variance. Thus, we are optimizing the design for the worst-case scenario as regard the design problem, which occurs when the interaction variance is negligible relative to the error variance. Of course, when it comes to analysis of an experiment, we would add the interaction back and analyze accordingly. Also, at the analysis stage, small interaction would constitute the best-case scenario.

Choice of a suitable p-rep design can be conducted as a two-stage process. First, an optimal or near-optimal α -array is obtained for the desired parameters using, say CycDesign. For $k' > s$ this will be an $\alpha(0,1,2)$ -array. In the second stage elements are dropped from the α -array in a way that maximizes E for the final p-rep design. The optimization process can be carried out using updating techniques on the information matrix of the design as described by John and Whitaker (2000), because our design is binary. This second stage will be incorporated into the next version of CycDesign.

As discussed, the $m \times 2$ rectangles down the diagonal direction of the α -array generate the resolvable two-replicate designs at each location. It is important to ensure that these designs are connected (John and Williams, 1995, Section 1.8). A simple criterion for connectedness is that if an $m \times 2$ generating array is written in a reduced form, say

$$\begin{array}{cc} 0 & 0 \\ 0 & a_1 \\ 0 & a_2 \\ & \vdots \\ 0 & a_{m-1} \end{array}$$

then at least one element a_i should be co-prime with s , or co-prime with another element a_j ($i, j = 1, 2, \dots, m-1$). It is usually possible to ensure that the two-replicate designs at each location are connected since one can re-order rows and/or columns of an α -array to achieve this; such re-ordering does not change the properties of the resulting α -design.

2.4 Construction using α_2 -designs for m even

John *et al.* (2002) introduced α_2 -designs, which are an extension of α -designs (John and Williams, 1995). An α_2 -design for $v = k^*s^*$ entries can be constructed from a $k^* \times l$ array of 2-ples, where k^* and s^* can be factored into components $k^* = k_1k_2$ and $s^* = s_1s_2$. In this section we assume that m is an even number and $k^* = ml/2$, where m and l are as defined earlier. The factorization required for our purpose is $k_1 = ml/2$, $k_2 = 1$, $s_1 = 2$ and $s_2 = s$. Details of the construction of α_2 -designs from an α_2 -array are given by John *et al.* (2002). Here we will simply give an example.

Example 3. For $k^* = 3$, $s = 7$, $m = 2$ and $l = 3$ the α_2 -array

$$\begin{array}{ccc} 13 & 06 & 16 \\ 16 & 00 & 01 \\ 00 & 05 & 16 \end{array} \quad (5)$$

generates the (unrandomized) resolvable design in Table 4 for 42 entries in three replicates, each with 14 incomplete blocks of size 3.

As with α -designs, the α_2 -array (5) can be used to assess the number of pairwise concurrences in the generated design. An α_2 -array generates an $\alpha_2(0,1)$ -design if the set of column differences, modulo (s_1, s_2) in (5) are all different. For example, the difference between the first elements of columns 1 and 2 in (5) is $0, 6 - 1, 3 = 1, 3$ modulo $(2, 7)$.

We now discuss how to augment the design in Table 4 so that it can be used as a p-rep design for three locations. The α_2 -property of the design means that it is a simple matter to create a resolvable

Table 4 Unrandomized α_2 -design for 42 entries in blocks of size 3.

Rep 1													
11	12	13	14	8	9	10	4	5	6	7	1	2	3
28	22	23	24	25	26	27	21	15	16	17	18	19	20
29	30	31	32	33	34	35	36	37	38	39	40	41	42
Rep 2													
7	1	2	3	4	5	6	14	8	9	10	11	12	13
15	16	17	18	19	20	21	22	23	24	25	26	27	28
34	35	29	30	31	32	33	41	42	36	37	38	39	40
Rep 3													
14	8	9	10	11	12	13	7	1	2	3	4	5	6
16	17	18	19	20	21	15	23	24	25	26	27	28	22
42	36	37	38	39	40	41	35	29	30	31	32	33	34

Table 5 Unrandomized p-rep design for 42 entries in blocks of size 4.

Location 1													
3	4	5	6	7	1	2	10	11	12	13	14	8	9
11	12	13	14	8	9	10	4	5	6	7	1	2	3
28	22	23	24	25	26	27	21	15	16	17	18	19	20
29	30	31	32	33	34	35	36	37	38	39	40	41	42
Location 2													
7	1	2	3	4	5	6	14	8	9	10	11	12	13
24	25	26	27	28	22	23	17	18	19	20	21	15	16
15	16	17	18	19	20	21	22	23	24	25	26	27	28
34	35	29	30	31	32	33	41	42	36	37	38	39	40
Location 3													
14	8	9	10	11	12	13	7	1	2	3	4	5	6
16	17	18	19	20	21	15	23	24	25	26	27	28	22
29	30	31	32	33	34	35	36	37	38	39	40	41	42
42	36	37	38	39	40	41	35	29	30	31	32	33	34

two-replicate design at each location for subsets of entries. For example the design in Table 5 is a possible (unrandomized) p-rep design.

This design can be generated by the augmented α_2 -array

13 ₀₂	06	16
16	00 ₁₂	01
00	05	16 ₀₀

where we have augmented the diagonal elements of (5) with a new set of 2-ples. This new set has been chosen so that:

- (i) The first element of the 2-ple is different to the first element of the existing 2-ple.
- (ii) The α_2 -array using the new set of diagonal elements generates an efficient $\alpha_2(0,1)$ -design.
- (iii) The resulting two-replicate design at each location is an efficient resolvable design for 2s entries in blocks of size 2.

This method extends naturally to other even values of m , although it is unlikely that we would want to use values of m greater than 4. The method, as presented, also requires s to be odd since it is straightforward to show that the component two-replicate designs at each location will be disconnected when s is even. Connected designs for s even can, however, be generated by choosing more than one set of 2-ples for each diagonal element of the α_2 -array.

3 Discussion

As with Federer's augmented designs, analysis of augmented p-rep designs can be done by a classical incomplete block model per location. This could be complemented by a spatial add-on component (Piepho *et al.*, 2008; Piepho and Williams, 2010). Our design approach ignores such spatial error components, mainly because it is usually difficult to predict if addition of such components will be needed, and if so, what choice should be made among various modeling options. When such information is available, however, one may optimize the allocation of entries to plots accordingly (Williams *et al.*, 2006; Cullis *et al.*, 2006; Butler *et al.*, 2008). In the same vein, our

approach assumes a constant error term across environments. If heterogeneity is expected, and prior information is available, one might consider a weighted analysis (Möhring and Piepho, 2009).

A very simple alternative way to generate an augmented p-rep design across locations is to generate a two-replicate resolvable incomplete block design, such as an α -design, for replicated entries at each location and then augment these successively in a one-by-one fashion with unreplicated entries. It turns out, however, that by a naïve one-by-one augmentation it is almost unavoidable that concurrences greater than one occur. This will be the case even if the augmentation is monitored so as to minimize the number of pairwise concurrences. Application of numerical search strategies such as simulated annealing to explore different possible paths for one-by-one augmentation could be tried, but we expect such strategies to be much more computationally demanding than our proposed methods.

The construction method in this paper can easily be extended to allow for an unequal number of repeated entries at each location. For example suppose m_j is the block size of the repeated entries at location j . Then for the method in Section 2.4

$$k^* = \sum_{j=1}^l m_j/2$$

with final block sizes of $k_j = k^* + m_j/2$ ($j = 1, 2, \dots, l$). Similar results exist for the method in Section 2.1 but some locations will have block sizes differing by one plot, which is usually not a problem (John and Williams, 1995, Section 4.9).

Acknowledgements Hans-Peter Piepho was supported by the GABI GAIN project (grant. no FKZ0315072C).

Conflict of Interest

The authors have declared no conflict of interest.

References

- 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.
- 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.
- Denis, J.-B., Piepho, H. P. and van Eeuwijk, F. A. (1997). Modelling expectation and variance for genotype by environment data. *Heredity* **79**, 162–171.
- Federer, W. T. (1961). Augmented designs with one-way elimination of heterogeneity. *Biometrics* **17**, 447–473.
- Federer, W. T. (2002). Construction and analysis of an augmented lattice square design. *Biometrical Journal* **44**, 241–257.
- Federer, W. T., Nair, R. C. and Raghavarao, D. (1975). Some augmented row-column designs. *Biometrics* **31**, 361–375.
- John, J. A., Ruggiero, K. and Williams, E. R. (2002). α_n -Designs. *Australian and New Zealand Journal of Statistics* **44**, 457–465.
- John, J. A. and Whitaker, D. (2000). Recursive formulae for the average efficiency factor in block and row-column designs. *Journal of the Royal Statistical Society B* **62**, 575–583.
- John, J. A. and Williams, E. R. (1982). Conjectures for optimal block designs. *Journal of the Royal Statistical Society B* **44**, 221–225.
- John, J. A. and Williams, E. R. (1995). *Cyclic and Computer Generated Designs*. Chapman & Hall, London.
- Möhring, J. and Piepho, H. P. (2009). Comparison of weighting in two-stage analyses of series of experiments. *Crop Science* **49**, 1977–1988.

- Piepho, H. P. (1997). Analyzing genotype-environment data by mixed models with multiplicative effects. *Biometrics* **53**, 761–766.
- Piepho, H. P. (1998). Methods for comparing the yield stability of cropping systems – a review. *Journal of Agronomy and Crop Science* **180**, 193–213.
- Piepho, H. P., Richter, C. and Williams, E. R. (2008). Nearest neighbour adjustment and linear variance models in plant breeding trials. *Biometrical Journal* **50**, 164–189.
- Piepho, H. P. and Williams, E. R. (2010). Linear variance models for plant breeding trials. *Plant Breeding* **129**, 1–8.
- Smith, A. B., Lim, P. and Cullis, B. R. (2006). The design and analysis of multi-phase plant breeding experiments. *Journal of Agricultural Science* **144**, 393–409.
- Whitaker, D., Williams, E. R. and John, J. A. (2009). *CycDesign 4.0: A Package for the Computer Generation of Experimental Designs*. CycSoftware Ltd., Naseby, New Zealand.
- Williams, E. R. and John, J. A. (2003). A note on the design of unreplicated trials. *Biometrical Journal* **45**, 751–757.
- Williams, E. R., John, J. A. and Whitaker, D. (2006). Construction of resolvable spatial row-column designs. *Biometrics* **62**, 103–108.