VFAST Transactions on Mathematics
http://vfast.org/journals/index.php/VTM@ 2023, ISSN(e): 2309-0022, ISSN(p): 2411-6343
Volume 11, Number 2, July-December, 2023 pp: 16-27

An Algorithm Coded with R to Generate GN_2-designs in Circular Blocks

Javid Shabbir[1*], Jamshaid ul Hassan[2], Mahmood Ul Hassan[3], Khadija Noreen[2], Sajid Hussain[2], Rashid Ahmed[2]

[1]Department of Statistics, University of Wah, Wah Cantt; [2]Department of Statistics, The Islamia University of Bahawalpur, Pakistan; [3]Department of Statistics, Stockholm University, Stockholm, Sweden

Keywords: *Neighbor balanced designs, Neighbor effect, Economical design, Generalized neighbor design.* ***Subject Classification***: *05B05, 62K10, 62K05.*

Journal Info:
Submitted: April 15, 2023
Accepted: June 29, 2023
Published: Sep. 01, 2023

Abstract Minimal neighbor balanced designs are economical, therefore, these are preferred by the experimenters to minimize the bias due to neighbor effects. Minimal circular balanced neighbor designs cannot be constructed for almost every case of v even, where v is number of the treatments to be compared. For v even, the circular GN_2-designs in which each treatment appears exactly once as neighbors with all other treatments except the one with which it appears twice, are considered the better alternate to the minimal balanced neighbor designs. In this article, an algorithm is developed to generate the circular GN_2-designs for v even which can be converted directly into minimal circular balanced and strongly balanced neighbor designs. This algorithm is also coded with R.

***Correspondence Author Email Address:**
javidshabbir@gmail.com

1 Introduction

Effect of the neighboring unit treatment is called neighbor effect. Treatments comparisons are less affected by neighbor effects with the use of balanced neighbor designs (BNDs). If every treatment appears once with all other treatments (excluding itself) as neighbors then it is called minimal BND. If including itself then it is minimal strongly BND. Minimal circular BNDs (MCBNDs) and minimal circular strongly BNDs (MCSBNDs) can be generated only for v (number of treatments) odd. Method of cyclic shifts (Rule II) provides a circular design for v even where every treatment appears once as neighbor (either left or right) with all other treatments except a treatment with which it appears twice, are called circular Quasi Rees neighbor designs or circular GN_2 (CGN2) designs which are considered as the good alternate to the MCBNDs.

[17] and [24] applied MCBNDs in plant breeding and serology respectively. According to [6], neighbor effects are often shown in experiments of agriculture, forestry and horticulture where treatments on neighboring plots effect the response on a given plot. Neighbor effect may arise in agriculture experiments due to nature and layout of plots. BNDs minimize bias due to neighbor effects, see [5, 15]. [6] presented BNDs, using border plots. [4, 13] constructed some BNDs in circular binary blocks. [9] presented some classes of BNDs in non-binary circular blocks. [1, 2, 25, 26] are some more references for BNDs in circular binary blocks. Generalized neighbor designs (GNDs) are used when minimal BNDs could not be constructed. [7, 16, 21] constructed some different classes of GNDs. [14] presented some series of GN_2 and GN_3 designs. [3, 10, 12, 18, 19, 23, 27] constructed circular GNDs. [8] presented some classes of partially BNDs in non-binary circular blocks. Among GNDs, GN_2-designs are considered as the better alternate of minimal BNDs. [22] developed a simple method to construct minimal BNDs and GN_2-designs in linear blocks.

The constructions of MCBNDs or MCGNDs through method of cyclic shifts require the division of constructors into i groups, each of size k with the condition that the sum of each group is divisible by v. Such division is a tedious work and time consuming. To overcome this problem, [20] developed R codes which are useful only for Rule I. The division of constructors into i groups is still a great problem while using Rule II which should be solved. To resolve this problem, R-coded algorithm is developed in this article, to generate the GN_2-designs using Rule II. It is an innovational research work which produces required CGN_2-designs by just putting the values of i (number of groups) and k (block size) which can directly be converted into MCBNDs and MCSBNDs. Source codes of the algorithm are publicly available at https://github.com/scenic555/CGN2-Designs.

2 Method of Cyclic Shifts

[11] introduced a method which is described here for CGN_2-designs, CSBNDs and CBNDs.

2.1 Construction of CBNDs and CSBNDs, using method of cyclic shifts (Rule I)

Let $S_j = [q_{j1}, q_{j2}, \cdots, q_{j(k-1)}]$ be sets of shifts and $S^* = [q_{j1}, q_{j2}, \cdots, q_{j(k-1)}, (q_{j1} + q_{j2} + \cdots + q_{j(k-1)}) \bmod v, v - q_{j1}, v - q_{j2}, \cdots, v - q_{j(k-1)}, v - (q_{j1} + q_{j2} + \cdots + q_{j(k-1)}) \bmod v]$.

- If each of $1, 2, \cdots, v-1$ appears once in S^* for $1 \leq q_{ij} \leq v-1$ then it is MCBND.
- If each of $0, 1, 2, \cdots, v-1$ appears once in S^* for $0 \leq q_{ij} \leq v-1$ then it is MCSBND.

Example 1. $S_1 = [16, 3, 2, 8]$, $S_2 = [6, 7, 4]$ *produce MCSBND for* $v = 17$, $k_1 = 5$, $k_2 = 4$.

Proof. $S^* = [16, 3, 2, 8, \mathbf{12}, 6, 7, 4, 0, 1, 14, 15, 9, 5, 11, 10, 13]$, each of $0, 1, \cdots, 16$ appears exactly once. So $S_1 = [16, 3, 2, 8]$, $S_2 = [6, 7, 4]$ produce MCSBND for $v = 17$, $k_1 = 5$ and $k_2 = 4$.

Consider $0, 1, \cdots, v-1$ as first row of v blocks taken for S_1. Generate 2nd row by adding 16 (mod 17) to its corresponding first row values. Generate 3rd row by adding 3 (mod 17) to its corresponding 2nd row values, and similarly add 2 and 8 for 3rd and 4th row, see Table 1.

Take more v blocks for S_2 and obtain the arrays in the similar way of S_1, see Table 2.

Table 1. Arrays obtained from $S_1 = [16, 3, 2, 8]$

Blocks																
1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
16	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	0	1
4	5	6	7	8	9	10	11	12	13	14	15	16	0	1	2	3
12	13	14	15	16	0	1	2	3	4	5	6	7	8	9	10	11

Table 2. Arrays obtained from $S_2 = [6, 7, 4]t$

Blocks																
18	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**	**27**	**28**	**29**	**30**	**31**	**32**	**33**	**34**
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
6	7	8	9	10	11	12	13	14	15	16	0	1	2	3	4	5
13	14	15	16	0	1	2	3	4	5	6	7	8	9	10	11	12
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Table 1 and Table 2 jointly produce the MCSBND for $v = 17$, $k_1 = 5$, $k_2 = 4$. □

Example 2. $S_1 = [16, 3, 2, 8]$, $S_2 = [6, 7]$ *produce MCBND for* $v = 17$, $k_1 = 5$, $k_2 = 3$.

Proof. $S^* = [16, 3, 2, 8, \mathbf{12}, 6, 7, \mathbf{13}, 1, 14, 15, 9, 5, 11, 10, 4]$, each of $1, 2, \cdots, 16$ appears exactly once. So $S_1 = [16, 3, 2, 8]$, $S_2 = [6, 7]$ produce MCBND for $v = 17$, $k_1 = 5$ and $k_2 = 3$. □

2.2 Construction of CGN_2-designs, using the Rule II

Let $S_j = [q_{j1}, q_{j2}, \cdots, q_{j(k-1)}]$ be $(i-1)$ sets and $S_i = [q_{i1}, q_{i2}, \cdots, q_{i(k-2)}]t$ and $S^* = [q_{j1}, q_{j2}, \cdots, q_{j(k-1)}, (q_{j1} + q_{j2} + \cdots + q_{j(k-1)}) \bmod (v-1), (v-1) - q_{j1}, (v-1) - q_{j2}, \cdots, (v-1) - q_{j(k-1)}, (v-1) - (q_{j1} + q_{j2} + \cdots + q_{j(k-1)}) \bmod (v-1), q_{i1}, q_{i2}, \cdots, q_{i(k-2)}]$. If each of $1, 2, \cdots, v-1$ appears once in S^* for $1 \leq q_{ij} \leq v-2$ then it is CGN_2-design.

2.3 Logic behind the Rule II to Produce CGN_2-designs

If the smallest block size is greater than four then CGN_2-designs (which are convertible into CBNDs and CSBNDs) can be constructed as:

- Let $A = [1, 2, \cdots, m]$, if sum of A is not factor of $(v-1)$, for it replace one or two values with their complements, where complement of 'a' is '(v-1)-a' and $m = (v-2)/2$.
- For equal block sizes, divide adjusted A in $(i-1)$ groups with each of k values and one of $k-2$ values so that sum of every group is factor of $(v-1)$.
- For unequal block sizes, divide adjusted A in i groups with each of k_1 values, one group each of $k_2, k_3, \cdots, k_h - 1$ values and one of $k_h - 2$ values so that sum of every group is factor of $(v-1)$.
- Delete one value from all groups except one which contains $k_h - 2$ values which remains unchanged. These obtained sets produce CGN_2-design.

Example 3. $S_1 = [16, 3, 2, 8]$, $S_2 = [6, 7, 4]t$ *produce* CGN_2*-design for* $v = 18$ *with* $k = 5$.

Table 3. Arrays obtained from $S_1 = [16, 3, 2, 8]$

Blocks																
1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
16	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	0	1
4	5	6	7	8	9	10	11	12	13	14	15	16	0	1	2	3
12	13	14	15	16	0	1	2	3	4	5	6	7	8	9	10	11

Table 4. Arrays obtained from $S_2 = [6, 7, 4]t$

Blocks																
18	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**	**27**	**28**	**29**	**30**	**31**	**32**	**33**	**34**
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
6	7	8	9	10	11	12	13	14	15	16	0	1	2	3	4	5
13	14	15	16	0	1	2	3	4	5	6	7	8	9	10	11	12
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
17	17	17	17	17	17	17	17	17	17	17	17	17	17	17	17	17

Proof. $S^* = [16, 3, 2, 8, \mathbf{12}, 6, 7, 4, 1, 14, 15, 9, 5, 11, 10, 13]$, each of $1, 2, \cdots, 16$ appears exactly once. So $S_1 = [16, 3, 2, 8]$ and $S_2 = [6, 7, 4]$ produce CGN_2-design for $v = 18$ with $k = 5$. □

Consider $0, 1, \cdots, v-2$ as first row of $(v-1)$ blocks taken for S_1. Generate 2nd row by adding 16 (mod 17) to its corresponding first row values. Generate 3rd row by adding 3 (mod 17) to its corresponding 2nd row values, and similarly add 2 and 8 for 3rd and 4th row, see Table 3.

Get the arrays from S_2 in the next $(v-1)$ blocks in the similar way of S_1. Add one more row with $(v-1)$ as its each unit value, see Table 4.

Table 3 and Table 4 jointly produce the CGN_2-design for $v = 18$ and $k = 5$.

3 Algorithm to generate CGN$_2$–designs

Here, an algorithm is developed to generate CGN_2–designs which can be converted directly into MCBNDs and MCSBNDs.

3.1 Generation of sets of shifts for CGN$_2$–designs

To generate sets of shifts for CGN_2-designs, elements of C will be divided into required number of groups so that sum of every group will be multiple of $(v-1)$. Delete the smallest value from each group except the last group with $k_h - 2$ values, which will remain same. Obtained sets produce proposed CGN_2-designs in blocks of:

- Equal sizes for $v = 2ik - 2$, $k > 4$, $i > 1$. Divide C (chosen from Section 3.2) in $(i-1)$ groups of size k and one of remaining $(k-2)$ values. Let remaining three values are 2, 5, 6 for $k = 5$, then this set of shifts will be written as $[2, 5, 6]t$.

- Two different sizes for $v = 2ik_1 + 2k_2 - 2$, $k_1 > k_2 > 4$. Divide C (chosen from Section 3.2) in i groups of size k_1 and last of remaining $(k_2 - 2)$ values.
- Two different sizes for $v = 2ik_1 + 4k_2 - 2$, $k_1 > k_2 > 4$. Divide C (chosen from Section 3.2) in i groups of size k_1, one of size k_2 and last of remaining $(k_2 - 2)$ values.
- Three different sizes for $v = 2ik_1 + 2k_2 + 2k_3 - 2$, $k_1 > k_2 > k_3 > 4$. Divide C (chosen from Section 3.2) in i groups of size k_1, one of size k_2 and last of remaining $(k_3 - 2)$ values.
- Three different sizes for $v = 2ik_1 + 4k_2 + 2k_3 - 2$, $k_1 > k_2 > k_3 > 4$. Divide C (chosen from Section 3.2) in i groups of size k_1, two of size k_2 and last of remaining $(k_3 - 2)$ values.
- Three different sizes for $v = 2ik_1 + 2k_2 + 4k_3 - 2$, $k_1 > k_2 > k_3 > 4$. Divide C (chosen from Section 3.2) in i groups of size k_1, one of size k_2, one of size k_3 and last of remaining $(k_3 - 2)$ values.
- Three different sizes for $v = 2ik_1 + 4k_2 + 4k_3 - 2$, $k_1 > k_2 > k_3 > 4$. Divide C (chosen from Section 3.2) in i groups of size k_1, two of size k_2, one of size k_3 and last of remaining $(k_3 - 2)$ values.

3.2 Constructors which produce the sets of shifts for CGN$_2$–designs

Elements of the following 'C' produce the CGN_2, where $m = (v - 2)/2$. If m (mod 8):

- $\equiv 0$, then $C = [1, 2, \cdots, m, 2m - j + 1$. Delete value equal to $j]$ with $j = m/8$, $j \geq 1$.
- $\equiv 1$, then $C = [1, 2, \cdots, m - 1, m + 1, 2m - 3j$. Delete value equal to $3j + 1]$ with $j = (m - 1)/8$, $j \geq 1$.
- $\equiv 2$, then $C = [1, 2, \cdots, m - 1, m + 1, 2m - 5j - 1$. Delete value equal to $5j + 2]$ with $j = (m - 2)/8$, $j \geq 1$.
- $\equiv 3$, then $C = [1, 2, \cdots, m, m + j + 1$. Delete value equal to $m - j]$ with $j = (m - 3)/8$, $j \geq 0$.
- $\equiv 4$, then $C = [1, 2, \cdots, m - 1, m + 1, 2m - j$. Delete the value equal to $j + 1]$ with $j = (m - 4)/8$, $j \geq 0$.
- $\equiv 5$, then $C = [1, 2, \cdots, m, 2m - 3j - 1$. Delete the value equal to $3j + 2]$ with $j = (m - 5)/8$, $j \geq 0$.
- $\equiv 6$, then $C = [1, 2, \cdots, m, 2m - 5j - 3$. Delete the value equal to $5j + 4]$ with $j = (m - 6)/8$, $j \geq 0$.
- $\equiv 7$, then $C = [1, 2, \cdots, m - 1, m + 1, m + j + 1$. Delete the value equal to $m - j]$ with $j = (m - 7)/8$, $j \geq 0$.

4 CGN$_2$-designs

In this section, some CGN_2-designs are presented in blocks of equal, two and three different sizes. In all our proposed designs, the complete set of parameters is: v, b, $n_1 = v - 2$, $n_2 = 1$, $\lambda'_1 = 1$, $\lambda'_2 = 2$. See snap shots in Appendix to run the program.

4.1 CGN$_2$-designs in blocks of equal sizes

Here, CGN_2-designs are constructed for $v = 2ik - 2$. See Table 5

4.2 CGN$_2$-designs in blocks of two different sizes

Here, CGN_2-designs are constructed for $v = 2ik_1 + 2k_2 - 2$ from i sets for k_1 and either one set for k_2, or two sets for k_2.

4.2.1 CGN_2-designs in block sizes k_1 and k_2, using one set for k_2

See Table 6

4.2.2 CGN_2-designs in block sizes k_1 and k_2, using two sets for k_2

See Table 7

Table 5. CGN_2-designs in blocks of equal sizes

v	c	k	Sets of Shifts
34	0	9	[16,31,13,4,5,6,8,7]+[10,11,3,12,14,15,1]t
52	1	9	[1,2,3,4,5,6,7,14]+[41,12,11,13,8,15,17,16]+ [19,18,21,22,23,24,26]t
40	2	7	[12,2,3,4,5,6]+[8,9,22,11,1,13]+[15,16,10,18,19]t
48	3	5	[24,11,3,4]+[6,7,8,9]+[2,10,20,14]+[16,12,18,22]+[19,13,15]t
42	4	11	[21,2,38,5,4,6,7,8,9,10]+[12,11,14,15,16,17,18,19,1]t
28	5	5	[13,12,4,3]+[1,7,8,2]+[10,11,6]t
30	6	8	[14,2,3,13,5,6,7]+[20,11,10,12,4,1]t
48	7	5	[24,11,3,4]+[6,7,8,9]+[2,10,20,14]+[16,12,18,22]+[19,13,15]t

Table 6. CGN_2-designs in blocks of two different sizes, using one set for k_2

v	c	k_1	k_2	Sets of Shifts
34	0	10	8	[16,31,4,3,5,6,7,8,9]+[11,12,13,14,15,1]t
20	1	6	5	[5,3,8,15,1]+[7,2,10]t
22	2	7	5	[11,2,3,6,5,1]+[8,9,4]t
24	3	7	6	[11,2,3,4,13,6]+[8,9,5,1]t
26	4	8	6	[13,23,9,4,5,6,7]+[3,10,11,1]t
28	5	8	7	[1,2,3,4,22,7,6]+[8,10,11,12,13]t
30	6	9	7	[8,13,1,3,6,5,7,14]+[29,10,4,12,2]t
32	7	9	8	[10,2,3,4,5,6,7,16]+[1,8,11,13,12,17]t

Table 7. CGN_2-designs in blocks of two different sizes, using two sets for k_2

v	c	k_1	k_2	Sets of Shifts
34	0	8	5	[1,31,3,4,5,6,7,9]+[10,11,14,15]+[8,12,13]t
36	1	7	6	[18,2,3,4,5,10]+[15,6,8,12,13]+[14,9,1,11]t
38	2	8	6	[12,3,4,5,6,7,9]+[10,11,8,13,17]+[25,16,14,19]t
40	3	9	6	[19,18,16,22,6,5,14,8]+[1,2,3,11,15]+[4,13,12,10]t
42	4	8	7	[1,2,38,15,5,6,7]+[18,10,11,12,13,14]+[16,17,19,9,21]t
44	5	9	7	[16,19,12,14,13,15,18,17]+[3,4,6,7,21,35]+[1,2,9,11,20]t
46	6	10	7	[22,16,3,4,5,6,7,8,9]+[1,31,13,11,15,17]+[18,19,20,21,12]t
32	7	7	5	[1,2,3,4,5,6]+[9,17,11,13]+[7,8,16]t

Table 8. CGN_2-designs in blocks of three different sizes

v	c	k_1	k_2	k_3	Sets of Shifts
34	0	7	6	5	[1,31,12,4,5,6]+[16,9,15,3,10]+[14,11,8]t
36	1	8	6	5	[1,15,3,4,5,6,28]+[2,9,16,12,13]+[11,10,14]t
38	2	9	6	5	[19,2,3,4,5,17,8,7]+[10,11,25,14,13]+[16,6,15]t
40	3	9	7	5	[19,2,18,4,5,7,8,6]+[10,11,12,13,1,15]+[22,3,14]t
42	4	9	8	5	[1,2,38,5,4,6,7,8]+[21,12,19,14,15,9,16]+[18,13,10]t
44	5	9	8	6	[1,16,3,4,5,6,7,35]+[18,11,13,21,14,15,20]+[10,19,2,12]t
46	6	9	8	7	[1,2,3,4,5,6,7,8]+[10,11,22,31,13,16,15]+[18,19,20,21,12]t
48	7	10	8	7	[24,18,3,4,5,6,7,8,9]+[17,19,13,1,16,15,11]+[12,20,26,22,24]t

Table 9. Conversion of CGN_2-designs given in Table 5 into MCSBNDs

v	k_1	k_2	Sets of Shifts
33	9	8	[16,31,13,4,5,6,7,8]+[10,11,12,3,14,15,1]
51	9	8	[1,2,3,4,5,6,7,14]+[41,12,11,13,8,15,17,16]+[19,18,21,22,23,24,26]
39	7	6	[12,2,3,4,5,6]+[8,9,22,11,1,13]+[15,16,10,18,19]
47	5	4	[24,11,3,4]+[6,7,8,9]+[2,10,20,14]+[16,12,18,22]+[19,13,15]
41	11	10	[21,2,38,5,4,6,7,8,9,10]+[12,11,14,15,16,17,18,19,1]
27	5	4	[13,12,4,3]+[1,7,8,2]+[10,11,6]
29	8	7	[14,2,3,13,5,6,7]+[20,11,10,12,4,1]
47	5	4	[24,11,3,4]+[6,7,8,9]+[2,10,20,14]+[16,12,18,22]+[19,13,15]

4.3 CGN_2-designs in blocks of three different sizes

Here, CGN_2-designs are constructed for $v = 2ik_1 + 2k_2 + 2k_3 - 2$ from i sets for k_1, one set each for k_2 and k_3. See Table 8

5 Conversion of CGN_2-designs into MCSBNDs and MCBNDs

In this Section, conversion of proposed CGN_2-designs into MCSBNDs and MCBNDs is described. In these designs $\lambda' = 1$.

Conversion 1. *CGN_2-designs constructed in Section 4 can directly be converted into MCSBNDs just by considering it Rule II with a change $v^* = v - 1$ and $k_h^* = k_h - 1$, where k_h is the smallest block size.*

Example 4. *(Conversion of CGN_2-design given in Table 5 into MCSBND): See Table 9*

Conversion 2. *CGN_2-designs constructed in Section 4 can be converted directly into MCBNDs with a change $v^* = v - 1$ and $k_h^* = k_h - 2$, just by (i) considering it Rule II, and (ii) deleting one value from the set with $k - 2$ values.*

Example 5. *(Conversion of CGN_2-design given in Table 5 into MCBND): See Table 10*

Table 10. Conversion of CGN_2-designs given in Table 5 into MCBNDs

v	k_1	k_2	Sets of Shifts
33	9	8	[16,31,13,4,5,6,7,8]+[10,11,12,3,14,15]
51	9	8	[1,2,3,4,5,6,7,14]+[41,12,11,13,8,15,17,16]+[19,18,21,22,23,24]
39	7	6	[12,2,3,4,5,6]+[8,9,22,11,1,13]+[15,16,10,18]
47	5	4	[24,11,3,4]+[6,7,8,9]+[2,10,20,14]+[16,12,18,22]+[19,13]
41	11	10	[21,2,38,5,4,6,7,8,9,10]+[12,11,14,15,16,17,18,19]
27	5	4	[13,12,4,3]+[1,7,8,2]+[10,11]
29	8	7	[14,2,3,13,5,6,7]+[20,11,10,12,4]
47	5	4	[24,11,3,4]+[6,7,8,9]+[2,10,20,14]+[16,12,18,22]+[19,13]

Acknowledgement

Authors are thankful to the Reviewers for their valuable suggestions.

Author Contributions

Javid Shabbir: Conceptualization, Methodology. **Sajid Hussain**: Data curation, Writing- Original draft preparation, Software. **Mahmood Ul Hassan**: Visualization, Investigation. **Rashid Ahmed**: Supervision. **Jamshaid ul Hasan**: Software, Validation. **Khadija Noreen**: Writing- Reviewing and Editing.

Compliance with Ethical Standards

It is declare that all authors don't have any conflict of interest. It is also declare that this article does not contain any studies with human participants or animals performed by any of the authors. Furthermore, informed consent was obtained from all individual participants included in the study.

Funding Information

This research received no external funding.

Author Information

ORCID:
Mahmood Ul Hassan: 0000-0003-2889-0263
Sajid Hussain: 0000-0002-3848-0981
Khadija Noreen: 0000-0002-0377-0030
Rashid Ahmed: 0000-0001-9703-7296

References

[1] Ahmed, R. and Akhtar, M. [2011a], 'Designs balanced for neighbor effects in circular blocks of size six', *Journal of Statistical Planning and Inference* **141**, 687–691.

[2] Ahmed, R. and Akhtar, M. [2011b], 'Construction of neighbor balanced designs in linear blocks', *Communications in Statistics-Theory and Methods* **40**, 1–8.

[3] Ahmed, R., Akhtar, M. and Tahir, M. H. [2009], 'Economical generalized neighbor designs of use in serology', *Computational Statistics and Data Analysis* **53**, 4584–4589.

[4] Akhtar, M., Ahmed, R. and Yasmin, F. [2010], 'A catalogue of nearest neighbor balanced-designs in circular blocks of size five', *Pakistan Journal of Statistics* **20**(2), 397–405.

[5] Azais, J. M. [1987], 'Design of experiments for studying intergenotypic competition', *Journal of the Royal Statistical Society: Series B* **49**, 334–345.

[6] Azais, J. M., Bailey, R. A. and Monod, H. [1993], 'A catalogue of efficient neighbor designs with border plots', *Biometrics* **49**(4), 1252–61.

[7] Chaure, N. K. and Misra, B. L. [1996], 'On construction of generalized neighbor design', *Sankhya Series B* **58**, 45–253.

[8] Hamad, N. and Hanif, M. [2016], 'Non binary partially neighbor balanced designs for circular blocks', *Communications in Statistics-Theory and Methods* **45**(20), 5961–5965.

[9] Hamad, N., Zafaryab, M. and Hanif, M. [2010], 'Non-binary neighbor balance circular designs for $v = 2n$ and $\lambda = 2$', *Journal of Statistical Planning and Inference* **40**(11), 3013–3016.

[10] Hassan, J., Noreen, K., Rasheed, H. M. K., Ul Hassan, M. and Ahmed, R. [2021], 'Construction of circular quasi rees neighbor designs which can be converted into minimal circular balanced and strongly balanced neighbor designs', *Communications in Statistics-Theory and Methods* **52**(16), 5587–5605.

[11] Iqbal, I. [1991], 'Construction of experimental design using cyclic shifts', *Ph.D. Thesis, University of Kent at Canterbury, U.K* .

[12] Iqbal, I., Tahir, M. H., Aggarwal, M. L., Ali, A. and Ahmed, I. [2012], 'Generalized neighbor designs with block size 3', *Journal of Statistical Planning and Inference* **142**, 626–632.

[13] Iqbal, I., Tahir, M. H. and Ghazali, S. S. A. [2009], 'Circular neighbor-balanced designs using cyclic shifts', *Science in China Series A: Mathematics* **52**(10), 2243–2256.

[14] Kedia, R. G. and Misra, B. L. [2008], 'On construction of generalized neighbor design of use in serology', *Statistics and Probability Letters* **18**, 254–256.

[15] Kunert, J. [2000], 'Randomization of neighbour balanced designs', *Biometrical Journal* **42**(1), 111–118.

[16] Misra, B. L., Bhagwandas and Nutan, S. M. [1991], 'Families of neighbor designs and their analysis', *Communications in Statistics - Simulation and Computation* **20**, 427–436.

[17] Morgan, J. P. [1988], 'Poly cross designs with complete neighbor balance', *Euphytica* **39**, 59–63.

[18] Nadeem, M., Rasheed, M., Tahir, M. H., Noreen, K., Hussain, S. and Ahmed, R. [2021], 'An easy construction of generalized neighbor designs in minimal circular blocks', *Kuwait Journal of Science* **49**(4), 1–11.

[19] Noreen, K., Omer, T., Hassan, J., Rasheed, H. M. K. and Ahmed, R. [2023], 'Some new constructions of minimal efficient circular nearly strongly balanced neighbor designs', *Journal of King Saud University-Science* **35**, 1027–48.

[20] Noreen, K., Rashid, M. S., Shehzad, F., Ul Hassan, M., Noreen, Z., Omer, T. and Ahmed, R. [2022], 'Algorithms to obtain generalized neighbor designs in minimal circular blocks', *Communications in Statistics-Simulation and Computation, In Press* .

[21] Nutan, S. M. [2007], 'Families of proper generalized neighbor designs', *Journal of Statistical Planning and Inference* **137**, 1681–1686.

[22] Ollis, M. A. [2017], 'Some constructions for minimal neighbor designs and gn_2 designs in linear blocks', *Communications in Statistics-Theory and Methods* **46**(4), 1547–1550.

[23] Rasheed, M., Noreen, K., Ahmed, R., Tahir, M. H. and Jamal, F. [2021], 'Some useful classes of minimal weakly balanced neighbor designs in circular blocks of two different sizes', *Communications in Statistics-Theory and Methods* **21**(24), 8822–8839.

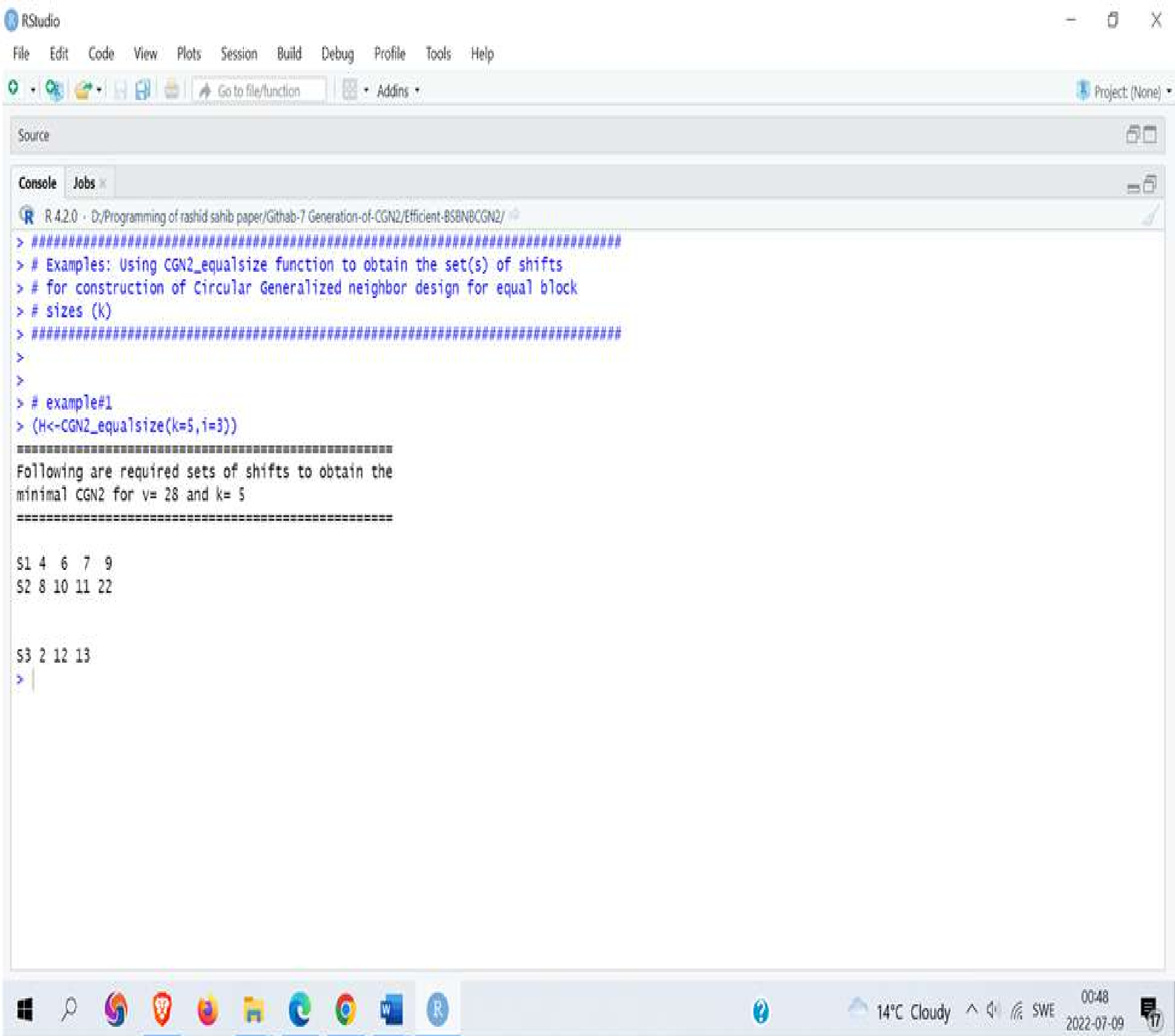
[24] Rees, D. H. [1967], 'Some designs of use in serology', *Biometrics* **23**, 779–791.

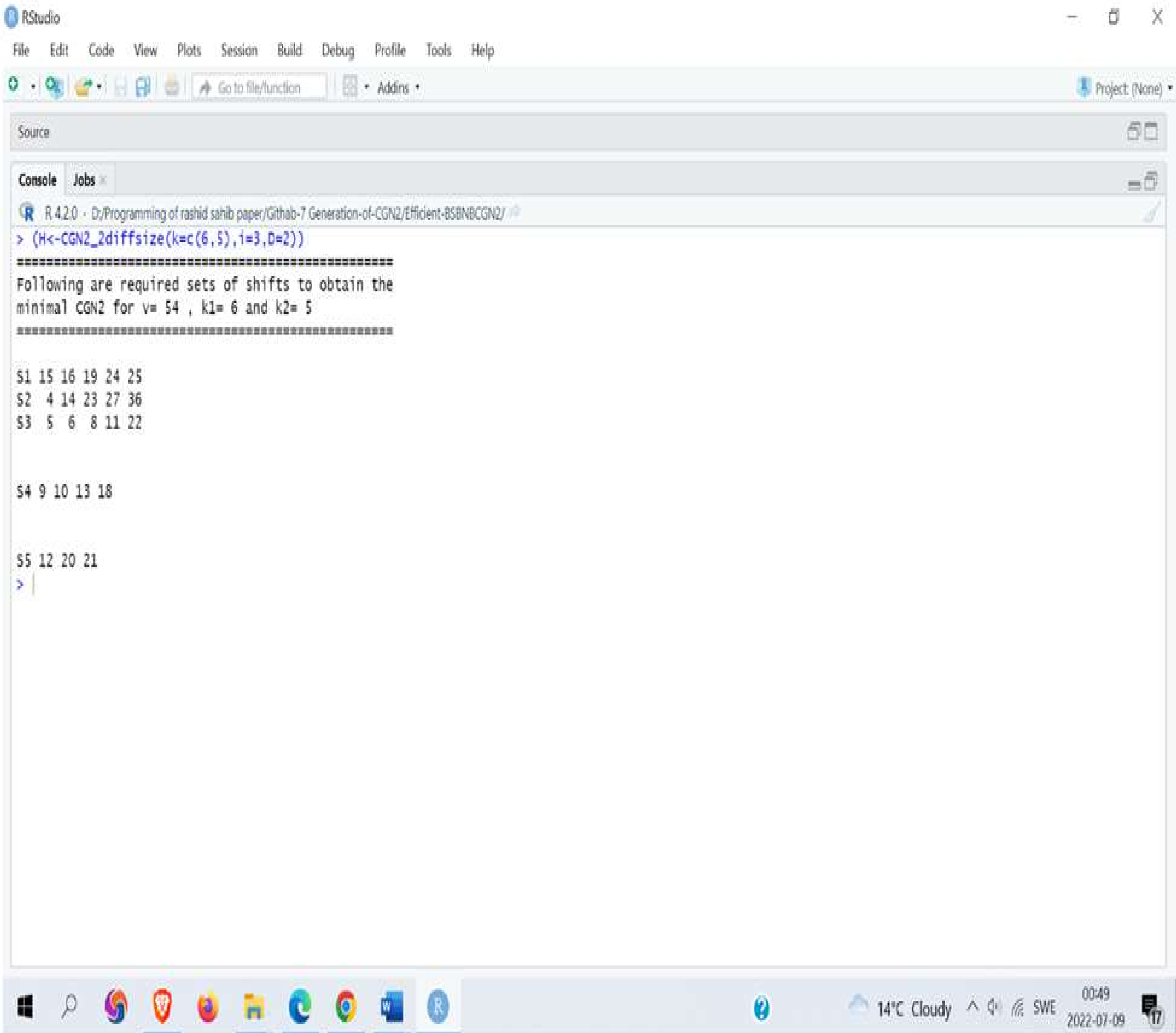
[25] Shehzad, F., Zafaryab, M. and Ahmed, R. [2011a], 'Minimal neighbor designs in circular blocks of unequal sizes', *Journal of Statistical Planning and Inference* **141**, 3681–3685.

[26] Shehzad, F., Zafaryab, M. and Ahmed, R. [2011b], 'Some series of proper generalized neighbor designs', *Journal of Statistical Planning and Inference* **141**, 3808–3818.

[27] Zafaryab, M., Shehzad, F. and Ahmed, R. [2010], 'Proper generalized neighbor designs in circular blocks', *Journal of Statistical Planning and Inference* **140**, 3498–3504.

Appendix


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
Project: (None)
Source
Console
Jobs
R 4.2.0 · D:/Programming of rashid sahib paper/Github-7 Generation-of-CGN2/Efficient-BSBNBCGN2/
> ##
> # Examples: Using CGN2_equalsize function to obtain the set(s) of shifts
> # for construction of Circular Generalized neighbor design for equal block
> # sizes (k)
> ##
>
>
> # example#1
> (H<-CGN2_equalsize(k=5,i=3))
==
Following are required sets of shifts to obtain the
minimal CGN2 for v= 28 and k= 5
==
S1 4 6 7 9
S2 8 10 11 22
S3 2 12 13
>
14°C Cloudy
SWE
00:48
2022-07-09


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
Project: (None)
Source
Console Jobs
R 4.2.0 · D:/Programming of rashid sahib paper/Githab-7 Generation-of-CGN2/Efficient-BSBNBCGN2/
> (H<-CGN2_2diffsize(k=c(6,5),i=3,D=2))
==
Following are required sets of shifts to obtain the
minimal CGN2 for v= 54 , k1= 6 and k2= 5
==
S1 15 16 19 24 25
S2 4 14 23 27 36
S3 5 6 8 11 22
S4 9 10 13 18
S5 12 20 21
>
14°C Cloudy
SWE
00:49
2022-07-09