Pak.j.stat.oper.res. Vol.18 No. 3 2022 pp 643-648 DOI: http://dx.doi.org/10.18187/pjsor.v18i3.3501

Pakistan Journal of Statistics and Operation Research

Construction of Some New Quasi Rees Neighbor Designs Using Cyclic Shifts

Dr. Saira Sharif[1*], Qaiser Mehmood[2], Rashid Ahmed[3] and Muhammad Rizwan Shahid[4]

* Corresponding Author

[1] Department of Economics and Statistics, University of Management and Technology Lahore, Pakistan, saira.sharif@umt.edu.pk
[2] Department of Economics and Statistics, University of Management and Technology Lahore, Government Graduate College Bahawalnagar,Pakistan, qaisarm11@gmail.com
[3] Department of Statistics The Islamia University Bahawalpur, Pakistan, rashid701@hotmail.com
[4] Department of Statistics, Allama Iqbal Open University Islamabad, Pakistan, rizwanuaf255@yahoo.com

Abstract

Many popular neighbor designs are used in serology, agriculture, and forestry which manifest neighbor effects very much. If every treatment appears as a neighbor with other (v-2) treatments once but emerges twice with only one treatment, such designs are called Quasi Rees neighbor designs (QRNDs) in k size of circular blocks. These designs were used for counterbalancing the neighboring effects for the cases for which minimal neighbor designs cannot be constructed. In this article, various generators are constructed to obtain circular binary NDs, using cyclic shifts.

Key Words: Neighbor designs; Minimal neighbor designs; Circular neighbor designs; Quasi Rees neighbor designs and Neighbor effects

Mathematical Subject Classification (2010): 05B05; 62K10; 62K05.

1. Introduction

In serology, initially Rees (1967) used neighbor designs (NDs) and applied this technique for the research on viruses. Then Hwang (1973), Cheng (1983), Azais *et al.* (1993), Ahmed and Akhtar (2008), Akhtar *et al.* (2010), Ahmed and Akhtar (2011) and ZafarYab *et al.* (2010) generated NDs for several cases. Shahid *et al*. (2017) presented some classes of NDs in linear blocks. Preece (1994) defined circular QRNDs in following manners:

(i) Every block has k (>2) distinct treatments from S,

(ii) Formation of blocks should be circular such that every treatment has one left- and one right-neighbor.

(iii) Each treatment from S comes exactly r times.

(iv) No treatment from S has itself as a neighbor.

(v) Every treatment from S appears once as neighbors with (v-2) other elements and appears twice with only one of the other treatments.

In this work to obtain circular binary QRNDs (CBQRNDs), some generators are either searched out from the existing generalized neighbor designs or newly developed. Following generators are searched out from the existing generalized NDs.

- Ahmed *et al*. (2009) expressed a series of generalized NDs for v = 2t +1, with t (integer) > 1.

- ZafarYab *et al*. (2010) expressed some series of generalized NDs in which $v/2$ pairs of treatments appear two times for $k = 8$ and v = 8t, 16t, and 24t.

The layout of this paper is as follows: Method of construction of CBQRNDs is explained in Section 2. Using cyclic shifts, some generators are developed in Section 3 and 4 to obtain CBQRNDs for even and odd v, respectively. In Section 5, a catalogue of CBQRNDs is presented for odd v and k, where $v < 50$ and $3 \leq k \leq 9$. A brief conclusion is provided in Section 6.

2. Construction of CBQRNDs

The method of cyclic shifts introduced by Iqbal (1991), is explained here only to obtain circular QRNDs. The rules of construction are:

Rule I: Let $S_j = [s_{j1}, s_{j2}, \ldots, s_{j(k-1)}]$ be the set of shifts, where $1 \leq s_{ij} \leq v\text{-}1$. **S*** must contain (i) every element from S_j with its complement, and (ii) sum of elements of S_j (mod v) along with its complement for each set. In Rule I, complement of 'a' is 'v-a'.

- If 1, 2 … (v-1) appear exactly once in **S*** except one value which appears twice along with its complement then it will be QRND.
- Design will be binary through Rule I, if aggregate of any two, three… or (k-1) successive elements of S is not 0 (mod v), if so, reorder the values.

Rule II: Let $S_j = [s_{j1}, s_{j2}, \ldots, s_{j(k-1)}]$ and $S_i = [s_{i1}, s_{i2}, \ldots, s_{i(k-2)}]$t. S* must contain (i) every element of S_j and S_i along with its complement, and (ii) sum of elements of S_j (mod (v-1)) along with their complement for each set in S_j. In Rule II, complement of 'a' is 'v-1-a'.

- If 1, 2 … (v-2) appear exactly once in **S*** except one value which comes twice along with its complement then it will be QRND.
- Design will be binary through Rule II, if aggregate of any two, three …, or (k-2) successive values of S is not 0 mod (v-1), if so, reorder the values.

3. Generators to obtain CBQRNDs for *v* even

Generator 3.1

Following i sets provide the CBQRNDs for $v = 8i\text{-}2$ and k = 4. Here (v-1) treatment appears two times as neighbors with each of other treatments while remaining other pairs appear only once as neighbors.

$S_{j+1,} = [v\text{-}(2+4j), 2+4j, 3+4j]$; $j \forall$ 0, 1 …i-2.

$S_{i,} = [v\text{-}(2+4j), 2+4j]$t; $j = i\text{-}1$.

Example 3.1: Sets of shifts [20,2,3]+[16,6,7]+[12,10]t provide CBQRND for v = 22 in blocks of size four.

Blocks

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

16	17	18	19	20	0	1	2	3	4	5	6	7	8	9
19	20	0	1	2	3	8	9	10	11	12	13	14	15	16
31	**32**	**33**	**34**	**35**	**36**	**37**	**38**	**39**	**40**	**41**	**42**	**43**	**44**	**45**
9	10	11	12	13	14	15	16	17	18	19	20	0	1	2
4	5	6	7	8	9	10	11	12	13	14	15	12	13	14
10	11	12	13	14	15	16	17	18	19	20	0	1	2	3
17	18	19	20	0	1	2	3	4	5	6	7	21	21	21
46	**47**	**48**	**49**	**50**	**51**	**52**	**53**	**54**	**55**	**56**	**57**	**58**	**59**	**60**
3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
15	16	17	18	19	20	0	1	2	3	4	5	6	7	8
4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
21	21	21	21	21	21	21	21	21	21	21	21	21	21	21
61	**62**	**63**	The following pairs appear twice.											
8	19	20	(0,21), (1,21), (2,21), (3,21), (4,21), (5,21), (6,21), (7,21),(8,21), (9,21), (10,21), (11,21), (12,21), (13,21), (14,21), (15,21), (16,21), (17,21), (18,21),(19,21), (20,21).											
9	10	11												
19	20	0												
21	21	21												

Generator 3.2

Following i sets provide CBQRNDs for $v = 12i$-2 and k = 6.

$S_{j+1,} = [v\text{-}(2+6j), 2+6j, v\text{-}(4+6j), 4+6j, 5+6j]$; $j \forall\ 0, 2\ldots, i\text{-}2$.

$S_{j+1,} = [v\text{-}(1+6j), 2+6j, 3+6j, v\text{-}(5+6j), 5+6j]$; $j \forall\ 1, 3\ldots, i\text{-}2$.

$S_{i,} = [v\text{-}(6j+2+w), 2+6j, 3+6j, (4+6j)]\text{t}$; $j = i\text{-}1$.

Where w = 0 for i odd and w = 1 for i even.

Generator 3.3

Following i sets provide CBQRNDs for $v = 16i–2$ and k = 8.

$S_{j+1,} = [v\text{-}(2+8j), 2+8j, v\text{-}(4+8j), 4+8j, 7+8j, v\text{-}(7+8j), 5+8j]$; $j \forall\ 0, 1, \ldots, i\text{-}2$.

$S_{i,} = [v\text{-}(2+8j), v\text{-}(4+8j), 2+8j, 4+8j, 5+8j, v\text{-}(7+8j)]\text{t}$; $j = i\text{-}1$

Generator 3.4

Following i sets provide CBQRNDs for $v = 20i$-2 and k = 10.

$S_{j+1,} = [v\text{-}(2+10j), 2+10j, v\text{-}(4+10j), 4+10j, v\text{-}(6+10j), 6+10j, 9+10j, v\text{-}(9+10j), 7+10j]$;

$j \forall\ 0, 2, \ldots, i\text{-}2$

$S_{j+1,} = [v\text{-}(1+10j), 2+10j, v\text{-}(4+10j), 4+10j, v\text{-}(7+10j), 7+10j, 5+10j, v\text{-}(9+10j), 9+10j]$;

$j \forall 1, 3\ldots, i\text{-}2$.

$S_{i,} = [v\text{-}(10j+w+1), 2+10j, 8+10j, 3+10j, v - (5+10j), 5+10j, 7+10j, v\text{-}(7+10j)]\text{t}$;
$j = i\text{-}1$.

Where w = 1 for i odd and w = 0 for i even.

Generator 3.5

Following set of shifts provide CBQRNDs for v = 2k-2, here k is size of the block.

$$S= [1, v\text{-}3, 3, v\text{-}5, 5, \ldots, (k\text{-}2) \text{ or } v\text{-}(k\text{-}1)]t$$

4. Generators to obtain CBQRNDs for *v* odd

Generator 4.1

Following i sets provide CBQRNDs for $v = 8i-1$ and k = 4. Here(0, (v-1)/2) (1, (v+1)/2) …, ((v-1)/2, v-1), (0, (v+1)/2), (1, (v+3)/2) …, ((v-3)/2, v-1) pairs appear two times as neighbors while all others appear once only.

$S_{j+1,}$= [v-(1+4j), 2+4j, 3+4j]; $j\forall$ 0, 1, 2…,i-1.

Generator 4.2

Following i sets provide CBQRNDs for $v = 12i-1$ and k = 6. Here (0, (v-1-u)/2), (1, (v+1+u)/2), …, ((v-1+u)/2, v-1), (0, (v+1+u)/2), (1, (v+3+u)/2), …, ((v-3-u)/2, v-1) pairs appear two times while all others appear only once.

$S_{j+1,}$= [v-(1+6j), 2+6j, v-(3+6j), 4+6j, 5+6j]; $j\forall$ 0, 2…,i-1.

$S_{j+1,}$= [v-6j, 6j+2, 6j+3, v-(6j+4), 6j+5]; $j\forall$ 1, 3…,i-1.

Where u = 2 for i odd and u = 0 for i even.

Generator 4.3

Following i sets provide CBQRNDs for $v = 16i-1$ and k = 8. Here(0, (v-1)/2), (1, (v+1)/2)… ((v-1)/2, v-1), (0, (v+1)/2), (1, (v+3)/2) …, ((v-3)/2, v-1) pairs appear two times as neighbors while all others appear only once.

$S_{j+1,}$=[v-(1+8j), 2+8j, v-(3+8j), 4+8j, 7+8j, v-(6+8j), 5+8j]; $j\forall$ 0, 1 …, i-1.

Generator 4.4

Following i sets provide CBQRNDs for $v = 20i-1$ and k = 10. Here(0, (v-1-u)/2), (1, (v+1-u)/2) … ((v-1+u)/2, v-1), (0, (v+1+u)/2), (1, (v+3+u)/2) … ((v-3-u)/2, v-1) pairs appear two times as neighbors while all others appear once.

$S_{j+1,}$= [v-(1+10j), 2+10j, v-(3+10j), 4+10j, v-(5+10j), 6+10j, 9+10j, v-(8+10j), 7+10j];
$j\forall$ 0, 2 …, i-1.

$S_{j+1,}$= [v-10j, 2+10j, v-(3+10j), 4+10j, v-(6+10j), 5+10j, 7+10j, v-(8+10j), 9+10j];
$j\forall$1, 3 …, i-1.

Where u = 2 for i odd and u = 0 for i even.

Generator 4.5

Following i partial sets of shifts provide CBQRNDs for v (prime) = k = 2m+1, where k is size of the block. Here, (0, 1), (1, 2), (v-2, v-1) and (0, v-1) pairs appear two times as neighbors while all other pairs appear only once as neighbors.

$$[1, 1 \ldots, 1](2/v) + [2, 2 \ldots, 2](1/v) + \ldots + [m, m \ldots, m](1/v)$$

5. CBQRNDs for odd *v* and k

CBQRNDs are presented below for odd *v* and k, where $v < 50$ and $3 \le k \le 9$.

v	k	Set(s) of Shifts	Pairs appear Twice
5	3	[2,4]	(0,1),(0,4),(1,2),(2,3),(3,4).
11	3	[1,2]+[5,7]	(0,1),(0,10),(1,2),(2,3),(3,4),(4,5),(5,6),(6,7),(7,8),(8,9),(9,10).
17	3	[1,2]+[4,5]+[7,11]	(0,1),(0,16),(1,2),(2,3),(3,4),(4,5),(5,6),(6,7),(7,8),(8,9),(9,10),(10,11),(11,12),(12,13),(13,14),(14,15),(15,16).
23	3	[1,5]+[3,4]+[2,10]+ [9,15]	(0,1),(0,22),(1,2),(2,3),(3,4),(4,5),(5,6),(6,7),(7,8),(8,9),(9,10),(10,11),(11,12),(12,13),(13,14),(14,15),(15,16),(16,17),(17,18),(18,19),(19,20),(20,21),(21,22).
29	3	[1,8]+[2,10]+[3,4]+ [5,6]+[14,16]	(0,1),(0,28),(1,2),(2,3),(3,4),(4,5),(5,6),(6,7),(7,8),(8,9),(9,10),(10,11),(11,12),(12,13),(13,14),(14,15),(15,16),(16,17),(17,18),(18,19),(19,20),(20,21),(21,22),(22,23),(23,24),(24,25),(25,26),(26,27),(27,28).
35	3	[1.9]+[3,4]+[5,12]+ [6,8]+[13,33]+ [16,20]	(0,1),(0,34),(1,2),(2,3),(3,4),(4,5),(5,6),(6,7),(7,8),(8,9),(9,10),(10,11),(11,12),(12,13),(13,14),(14,15),(15,16),(16,17),(17,18),(18,19),(19,20),(20,21),(21,22),(22,23),(23,24),(24,25),(25,26),(26,27),(27,28),(28,29),(29,30),(30,31),(31,32),(32,33),(33,34).
41	3	[1,9]+[1,16]+ [3,18]+[4,15]+ [6,8]+[12,36]+ [13,39]	(0,1),(0,40),(1,2),(2,3),(3,4),(4,5),(5,6),(6,7),(7,8),(8,9),(9,10),(10,11),(11,12),(12,13),(13,14),(14,15),(15,16),(16,17),(17,18),(18,19),(19,20),(20,21),(21,22),(22,23),(23,24),(24,25),(25,26),(26,27),(27,28),(28,29),(29,30),(30,31),(31,32),(32,33),(33,34),(34,35),(35,36),(36,37),(37,38),(38,39),(39,40).
47	3	[1,14]+[2,18]+ [3,19]+[5,43]+ [6,7]+[8,9]+ [10,16]+[11,12]	(0,1),(0,46),(1,2),(2,3),(3,4),(4,5),(5,6),(6,7),(7,8),(8,9),(9,10),(10,11),(11,12),(12,13),(13,14),(14,15),(15,16),(16,17),(17,18),(18,19),(19,20),(20,21),(21,22),(22,23),(23,24),(24,25),(25,26),(26,27),(27,28),(28,29),(29,30),(30,31),(31,32),(32,33),(33,34),(34,35),(35,36),(36,37),(37,38),(38,39),(39,40),(40,41),(41,42),(42,43),(43,44),(44,45),(45,46).
9	5	[2,8,3,4]	(0,1),(0,8),(1,2),(2,3),(3,4),(4,5),(5,6),(6,7),(7,8).
19	5	[1,17,3,4]+ [14,7,8,10]	(0,1),(0,18),(1,2),(2,3),(3,4),(4,5),(5,6),(6,7),(7,8),(8,9),(9,10),(10,11),(11,12),(12,13),(13,14),(14,15), (15,16),(16,17),(17,18).
29	5	[1,27,3,4]+ [5,7,8,20]+ [19,12,13,15]	(0,1),(0,28),(1,2),(2,3),(3,4),(4,5),(5,6),(6,7),(7,8),(8,9),(9,10),(10,11),(11,12),(12,13),(13,14),(14,15),(15,16),(16,17),(17,18),(18,19),(19,20),(20,21),(21,22),(22,23),(23,24),(24,25),(25,26),(26,27),(27,28).
39	5	[1,37,3,4]+ [5,7,8,30]+ [10,12,26,14]+ [24,17,18,20]	(0,1),(0,38),(1,2),(2,3),(3,4),(4,5),(5,6),(6,7),(7,8),(8,9),(9,10),(10,11),(11,12),(12,13),(13,14),(14,15),(15,16),(16,17),(17,18),(18,19),(19,20),(20,21),(21,22),(22,23),(23,24),(24,25),(25,26),(26,27),(27,28),(28,29),(29,30),(30,31),(31,32),(32,33),(33,34),(34,35),(35,36),(36,37),(37,38).
49	5	[1,47,3,4]+ [5,7,8,40]+ [39,12,13,14]+ [15,16,17,18]+ [21,25,27,26]	(0,1),(0,48),(1,2),(2,3),(3,4),(4,5),(5,6),(6,7),(7,8),(8,9),(9,10),(10,11),(11,12),(12,13),(13,14),(14,15),(15,16),(16,17),(17,18),(18,19),(19,20),(20,21),(21,22),(22,23),(23,24),(24,25),(25,26),(26,27),(27,28),(28,29),(29,30),(30,31),(31,32),(32,33),(33,34),(34,35),(35,36),(36,37),(37,38),(38,39),(39,40),(40,41),(41,42),(42,43),(43,44),(44,45),(45,46),(46,47),(47,48).
13	7	[12,10,2,4,5,7]	(0,1),(0,12),(1,2),(2,3),(3,4),(4,5),(5,6),(6,7),(7,8),(8,9),(9,10),(10,11),(11,12).
27	7	[26,2,3,4,5,21]+ [19,17,9,11,12,14]	(0,1),(0,26),(1,2),(2,3),(3,4),(4,5),(5,6),(6,7),(7,8),(8,9),(9,10),(10,11),(11,12),(12,13),(13,14),(14,15),(15,16),(16,17),(17,18),(18,19),(19,20),(20,21),(21,22),(22,23),(23,24),(24,25),(25,26).

41	7	[40,2,3,4,5,35]+ [8,9,10,30,12,28]+[27,16,24,18 ,19,20]	(0,1),(0,40),(1,2),(2,3),(3,4),(4,5),(5,6),(6,7),(7,8),(8,9),(9,10),(10 ,11),(11,12),(12,13),(13,14),(14,15),(15,16),(16,17),(17,18),(18,1 9),(19,20),(20,21),(21,22),(22,23),(23,24),(24,25),(25,26),(26,27) ,(27,28),(28,29),(29,30),(30,31),(31,32),(32,33),(33,34),(34,35),(35,36),(36,37),(37,38),(38,39),(39,40).
17	9	[2,16,14,8,5,11,4,7]	(0,1),(0,16),(1,2),(2,3),(3,4),(4,5),(5,6),(6,7),(7,8),(8,9),(9,10),(10 ,11),(11,12),(12,13),(13,14),(14,15),(15,16).
35	9	[1,2,30,6,7,4,27]+ [26,22,20,11,12,19,14,17]	(0,1),(0,34),(1,2),(2,3),(3,4),(4,5),(5,6),(6,7),(7,8),(8,9),(9,10),(10 ,11),(11,12),(12,13),(13,14),(14,15),(15,16),(16,17),(17,18),(18,1 9),(19,20),(20,21),(21,22),(22,23),(23,24),(24,25),(25,26),(26,27) ,(27,28),(28,29), (29,30),(30,31),(31,32),(32,33),(33,34).

6. Conclusions

Various generators are constructed to obtain circular binary NDs, using method of cyclic shifts. These generators are derived to obtain CBQRNDs for even and odd v with even and odd k size of the block, respectively. Further, all pairs which appear twice as neighbors and which appear only once as neighbors are segregated.

References

1. Ahmed, R., Akhtar, M. (2008). Construction of neighbor balanced block designs. *Journal of Statistical Theory and Practice*, 2 (4), 551- 558.
2. 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.
3. Ahmed, R. and Akhtar, M. (2011). Designs balanced for neighbor effects in circular blocks of size six. *Journal of Statistical Planning and Inference*, 141, 687-691.
4. Ahmed, R., Akhtar, M. and Yasmin, F. (2011). Brief review of one dimensional neighbor balanced designs since 1967. *Pakistan Journal of Commerce and Social Sciences*, 5(1), 100-116.
5. 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*, 26(2), 397- 405.
6. Azais, J. M., Bailey, R. A. and Monod, H. (1993). A catalogue of efficient neighbour- designs with border plots. *Biometrics*, 49 (4), 1252- 61.
7. Cheng, C.S. (1983). Construction of optimal balanced incomplete block designs for correlated observations. *Annals of Statistics*, 11, 240-246.
8. Hwang, F. K. (1973). Constructions for some classes of neighbor designs. *Annals of Statistics*, 1(4) 786- 790.
9. Iqbal, I. (1991). *Construction of experimental design using cyclic shifts.* Unpublished Ph.d Thesis. U.K: University of Kent at Canterbury.
10. 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.
11. Lawless, J. F. (1971). A note on certain types of BIBDs balanced for residual effects. *Annals of Mathematics and Statistics*, 42, 1439-1441.
12. Preece, D. A. (1994). Balanced Ouchterlony neighbor designs. *Journal of Combinatorial Mathematics and Combinatorial Computing*, 15, 197–219.
13. Rees, D. H. (1967). Some designs of use in serology. *Biometrics*, 23, 779- 791.
14. Shahid, M. R., Zakria, M., Shehzad, F. and Ahmed, R. (2017). Some important classes of generalized neighbor designs in linear blocks. *Communications in Statistics-Simulation and Computation*. 46(3), 1991-1997.
15. ZafarYab, M., Shehzad, F. and Ahmed, R. (2010). Proper generalized neighbor designs in circular blocks. *Journal of Statistical Planning and Inference*, 140(11), 3498-3504.