Construction and comparison of semi-Latin rectangles

Nseobong Peter Uto

A thesis submitted for the degree of PhD
at the
University of St Andrews

2021

Identifier to use to cite or link to this thesis:

DOI: https://doi.org/10.17630/10023-29182

Candidate's declaration

I, Nseobong Peter Uto, do hereby certify that this thesis, submitted for the degree of PhD, which is approximately 52,000 words in length, has been written by me, and that it is the record of work carried out by me, or principally by myself in collaboration with others as acknowledged, and that it has not been submitted in any previous application for any degree.

I was admitted as a research student at the University of St Andrews in October 2016.

I received funding from an organisation or institution and have acknowledged the funder(s) in the full text of my thesis.

Date 31/08/20 Signature of candidate

Supervisor's declaration

I hereby certify that the candidate has fulfilled the conditions of the Resolution and Regulations appropriate for the degree of PhD in the University of St Andrews and that the candidate is qualified to submit this thesis in application for that degree.

Date 31/8/20 Signature of supervisor

Permission for publication

In submitting this thesis to the University of St Andrews we understand that we are giving permission for it to be made available for use in accordance with the regulations of the University Library for the time being in force, subject to any copyright vested in the work not being affected thereby. We also understand, unless exempt by an award of an embargo as requested below, that the title and the abstract will be published, and that a copy of the work may be made and supplied to any bona fide library or research worker, that this thesis will be electronically accessible for personal or research use and that the library has the right to migrate this thesis into new electronic forms as required to ensure continued access to the thesis.

I, Nseobong Peter Uto, confirm that my thesis does not contain any third-party material that requires copyright clearance.

The following is an agreed request by candidate and supervisor regarding the publication of this thesis:

Printed copy

No embargo on print copy.

Electronic copy

No embargo on electronic copy.

Date 31/08/20 Signature of candidate

Date 31/8/20 Signature of supervisor

Underpinning Research Data or Digital Outputs

Candidate's declaration

I, Nseobong Peter Uto, hereby certify that no requirements to deposit original research data or digital outputs apply to this thesis and that, where appropriate, secondary data used have been referenced in the full text of my thesis.

Date 31/08/20 Signature of candidate

Acknowledgements

I am grateful to God Almighty for preserving me, particularly during the current Covid-19 pandemic, and keeping me continuously healthy and active in research throughout my study period.

I owe a debt of inestimable gratitude to my principal supervisor, Prof. Rosemary A. Bailey, for her supervisory and supportive roles. Her suggestions have always been superb. Moreover, I have been greatly inspired by her academic life and her supervision style challenged me to creative thinking. To my secondary supervisor, Dr Sophie Huczynska, I say, many thanks for always being available for me, particularly, whenever Prof Bailey was away for a couple of weeks.

I appreciate the active role played by the director of postgraduate studies, Prof James Mitchell, particularly during the difficult period. This has left a lasting impression in my mind. I must not fail to commend Profs Len Thomas and David Borchers of CREEM for their humaneness and encouragement.

Many thanks to Rev. Fr. Michael Galbraith of St James' Catholic Parish for his show of love and encouragement.

I appreciate the supportive attitude of my friends, colleagues and staff of the School of Mathematics and Statistics, particularly those in the general/administrative office, both in the Mathematical Institute and at CREEM. They have been wonderful people who have helped to make my stay in St Andrews rewarding, productive and enjoyable.

Special tribute to my family for their show of love, support and understanding despite all the inconveniences.

Finally, I wish to thank TetFund for financial support.

Abstract

This work is concerned with semi-Latin rectangles (SLRs). These designs are row-column designs with nice combinatorial properties; and were introduced in Bailey and Monod (2001). They generalize the Latin squares (LSs) and semi-Latin squares (SLSs) and are useful for many experimental situations in diverse sectors, ranging from agriculture to the industry. We classify these designs as balanced semi-Latin rectangles (BSLRs) and non-balanced semi-Latin rectangles (NBSLRs) and develop some constructions, via algorithms, for good SLRs, that is, SLRs with good statistical properties for each classification using some combinatorial approaches. BSLRs do not always exist, but when they exist, they are optimal among other SLRs in their class over a range of criteria. When a BSLR does not exist, good designs can be sought among RGSLRs, particularly for large number of blocks, if they exist. Hence for the NBSLRs we concentrate on regular-graph semi-Latin rectangles (RGSLRs). For each classification, constructions are given for designs with block size two and for those with block size larger than two; and for block size two, we consider situations when the number of treatments is odd and also when it is even. The construction involving RGSLRs with block size two having an odd number of treatments is generalized to accommodate more columns and a table showing starters in some cyclic groups of small odd orders, 5 to 15 is given to facilitate the construction. Some direct constructions, for different situations, have been developed for RGSLRs whose number of treatments is even and whose block size is two less the number of treatments. These are backed up with some examples, which when compared with designs of the same size obtained via complementation, they are found to be identical under one of the methods but isomorphic under the other method. Finally, for each of BSLRs and RGSLRs, we have given a table containing sets of parameters, which can combine to give a design alongside their construction and also where the design (or its construction, as the case may be) can be found in the thesis.

Contents

Chapter 1

Introduction

1.1 Introductory remarks

A semi-Latin rectangle (SLR) consists of v symbols (treatments) in h rows and p columns with k treatments in each row-column intersection (block) and each treatment appears a definite number of times, n_r, say, in each row and also a definite number of times, n_c, say, in each column, where v, h, p, k, n_r and n_c are positive integers: see Bailey and Monod (2001) and Uto and Bailey (2020). In particular, $h, p, k > 1$, $v|kh$ and $v|kp$. We note that at least one of n_r and n_c need not be unity; and h and p can have the same value or otherwise, though in Bailey and Monod (2001), where these designs were introduced, application was made to plant disease experiments which involve $h < p$, where h denotes the number of leaf-heights considered in the experiment and p, the number of plants. Moreover, $n_c = n_r$ if and only if $h = p$; $n_r > n_c$, if and only if $p > h$ and $n_c > n_r$, if and only if $h > p$: see Uto and Bailey (2020). We represent the structure by $(h \times p)/k$, which is read as h-by-p-by-k. If the rows and columns of a SLR are ignored, then the resulting block design is its quotient block design (QBD). We restrict the QBD to being binary, that is, each treatment appearing at most once in each block.

Semi-Latin rectangles (SLRs) thus constitute a class of row-column designs and they possess some nice combinatorial properties, which include, amongst others, orthogonality of its treatments with respect to the row and column strata and also n_r– and n_c–resolvability of the design: see Bailey and Monod (2001) and Uto and Bailey (2020).

Row-column designs are known to be useful designs for performing experiments in situations where there are two nuisance factors whose levels constitute the rows and columns, respectively, of the design, thereby controlling heterogeneity in the experimental units in two directions by reducing variability in experimental error. Thus the blocking in a row-column design is in two directions: see, for example, Williams and John (1996), Choi and Gupta (2008), Datta et al. (2017) and Godolphin (2019b).

Row-column designs such as the Latin square (LS) and semi-Latin square (SLS) which have been greatly studied in the literature could be derived from a SLR; they are special cases of the SLR and should not be confused with it. That is, the SLR generalizes both the LS and SLS: see Uto and Bailey (2020). In particular, if $h = p$ and $v = kh$, then $n_c = n_r = 1$ and the resulting design is essentially a SLS. Moreover, if $h = p$ and $v = h$, then if we allow $k = 1$, it implies that $n_c = n_r = 1$ and the design reduces to a LS. It follows that a SLS is a one-step generalization of the LS. Concerning Latin squares (LSs), see, for example, Bose and Nair (1941) and Ai et al. (2013); and for semi-Latin squares (SLSs), see Preece and Freeman (1983), Bailey (1988, 1992), Bedford and Whitaker (2001) and Soicher (2013). By superimposing two LSs of the same order, if each ordered pair of sybols (treatments) appears exactly once, then the two LSs are said to be orthogonal. In particular, a set of more than two LSs are said to be mutually orthogonal, called mutually orthogonal Latin squares (MOLSs), if every pair of LSs in the set are orthogonal. Furthermore, a set of MOLSs is said to be complete if its cardinality is 1 less the order of each LS in the set. MOLSs are useful in the construction of other designs such as affine-resolvable designs: see, for example, John and Williams (1995, Chapters 1 & 4), as well as Raghavarao and Padgett (2005, Chapters 4 & 9).

Moreover, by superimposing a set of MOLSs, where the treatment sets of the various LSs are pairwise disjoint, the resulting SLS is known as a Trojan square by Darby and Gilbert (1958). See also Edmondson (1998) for some discussion on this. A Trojan square, when it exists, is optimal among all SLSs and incomplete block designs of the size of its QBD over a range of criteria which include the A-, D- and E-criteria: see Cheng and Bailey (1991).

In practice, for certain experimental situations where the number of treatments available is a divisor of the number of plots available under each level of the row and column factors, the row (or column) factor may have more levels than the column (or row) factor, that is, the values of h and p may be different. However, in certain other situations, the values of h and p may be the same but the number of treatments available for the experiment is probably less than the number of plots available under each level of the row factor, hence, of column factor as well such that each treatment would need to appear more than once in each row and in each column. Hence in situations like these, a LS or SLS can not fit in and the SLR becomes a useful design.

Apart from the aforementioned application of SLRs involving plant disease experiments, they can also be used for conducting several other experiments, amongst which are food sensory experiments, consumer testing experiments and in similar experiments where a SLS can be used as reported in Bailey (1992): see Bailey and Monod (2001) and Uto and Bailey (2020).

There can be different kinds (or classes) of SLRs depending on the composition of their QBDs. For purposes of this work, we classify them into two main classes, viz, balanced semi-Latin rectangles (BSLRs) and non-balanced (or unbalanced) semi-Latin rectangles (NBSLRs).

We have named SLRs whose QBDs are balanced incomplete-block designs (BIBDs) as BSLRs otherwise we call them NBSLRs. see Uto and Bailey (2020) for discussions on BSLRs. BIBDs are binary incomplete-block designs that are proper and equireplicate and whose treatments concur the same number of times, for all pairs. Such designs do not always exist, but when they exist they are optimal over all incomplete-block designs of its size. Moreover, the estimation of elementary contrasts for treatment effects is done with the same variance: see Kiefer (1975). It follows that when a BSLR exists, it is optimal over its class: see Uto and Bailey (2020).

Moreover, for the NBSLRs, our focus is on those whose QBDs are regular-graph designs (RGDs) which are named RGSLRs in Bailey and Monod (2001). RGDs are binary incomplete-block designs that are close to balanced in the sense that the difference in the treatment concurrence counts between any two pairs of distinct treatments is at most unity. They give good designs for experiments, particularly, for large number of blocks, if they exist. In particular, when they exist, a D–optimal (or A–optimal or E–optimal) design for sufficiently large number of blocks is among them: see Cheng (1992), which confirms the earlier conjecture by John and Mitchell (1977). BIBDs are considered as a special case of RGDs: see Kreher et al. (1996). Within the class of RGDs, the BIBD-extended designs, called BIBD-extended RGDs, where each comprises a RGD part and a BIBD part, as extension have been found to produce highly A– and D–efficient designs among RGDs, if sufficiently large number of copies of a BIBD are added to a RGD: see Cakiroglu (2018). We call RGSLRs whose QBDs are BIBD-extended, BIBD-extended RGSLRs and in the particular case that a BSLR is adjoined to a RGSLR to give the extension, we call it a BSLR-extended RGSLR.

We give constructions for SLRs which have good statistical properties, that is, their QBDs are BIBDs, which implies BSLRs, when they exist. However, when a BSLR does not exist, we go for RGSLRs (if they exist), which are 'close' in some sense to BIBDs, in particular, we go for those ones whose QBDs are BIBD-extended; or in situations that we can obtain a BSLR that conforms in size to RGSLR and on the same treatment set, we simply extend the RGSLR by the BSLR to obtain a BSLR-extended RGSLR, since these give designs with good statistical properties.

1.2 Method of randomization of SLRs

The randomization procedure for a SLR is as follows: randomize all rows, randomize all columns independently of the rows, and then within each cell, independently randomize the order of the treatments.

1.3 BSLRs and RGSLRs

A BSLR is a SLR whose QBD is a BIBD, that is, its QBD is a binary, equireplicate and proper incomplete-block design which has a constant concurrence counts, λ (λ being a positive integer), between every pair of distinct treatments, that is, every pair of distinct treatments appears the same number of times in a block. A block design is incomplete if each block does not contain all the treatments, that is, the size of each block is less than the number of treatments in the design. As mentioned earlier, it is binary if every treatment appears at most once in each block. Moreover, a design is said to be equireplicate if each treatment appears the same number of times in the design, and in particular, for binary designs, equireplication implies each treatment appears in the same number of blocks. A design being proper means that its blocks are equal in size, that is, each block contains the same number of treatments. For a BIBD, since it is binary, equireplicate and proper, then each treatment appearing the same number of times implies that it appears in the same number of blocks.

Similarly, a RGSLR is a NBSLR whose QBD is a RGD, which is 'close' in some sense to being balanced as no two treatment concurrence counts differ by more than unity.

A BSLR does not always exist, just like a BIBD. For a fixed set of parameters $v, h, p, n_r, n_c, k, \lambda$, it exists only if the conditions in (1.1) and (1.2) are satisfied: see Uto and Bailey (2020). That is, the necessary conditions for the existence of a BSLR are as given in (1.1) and (1.2).

$$vhn_r = vpn_c = khp \tag{1.1}$$

$$\lambda(v-1) = hn_r(k-1) = pn_c(k-1) \tag{1.2}$$

We note that RGSLRs also need to satisfy the condition in (1.1). Clearly, what distinguishes a BSLR from a RGSLR is the balance property expressed in (1.2). Moreover, for certain sets of parameters a BSLR fails to exist, a RGSLR may exist.

Let $V = \{1, \ldots, v\}$ denote the treatment set of a RGSLR. For all $\tau \in V$, let τ appear with x treatments in λ^* blocks of its QBD. Similarly, let τ appear with y treatments in λ' blocks. Then a set of parameters which satisfy (1.3) and (1.4) (making a sensible choice of x and y such that $\lambda^* = \lambda' \pm 1$) can give the set of parameters that can make a RGSLR. However, in Table 7.2, we have assumed λ^* to be greater than λ'.

$$x + y = v - 1 \tag{1.3}$$

$$\lambda^* x + \lambda' y = hn_r(k-1) = pn_c(k-1) \tag{1.4}$$

We note that adjoining multiple copies of a BSLR gives another BSLR: see Theorem 4.7.1. However, this does not hold for a RGSLR (except with some suitable permutations applied to the treatments in each copy of the 'parent' design made) as the resulting design without some suitable permutation of the treatments of the 'parent' design will violate the treatment concurrence counts requirements; but if a BSLR is adjoined to a RGSLR, it gives another RGSLR: see Theorem 5.1.1. We call the resulting design a BSLR-extended

RGSLR since it is directly extended by a BSLR. Moreover, block (cell) complementation of a BSLR (or RGSLR) gives another BSLR (or RGSLR): see Theorems 4.6.1 and 6.5.1 for the case of BSLR and RGSLR, respectively.

Similarly, if the 'parent' and complementary designs, for the case involving a BSLR are adjoined, then the resulting design is also a BSLR: see Theorem 4.7.2. However, for this to work, both the 'parent' design and the complementary design need to have same block size, that is, $k = v/2$ such that $k + k' = v$, k' being the block size of the complementary design.

Moreover, given an $(h \times p)/k$ BSLR (or RGSLR), a transposition of it gives a $(p \times h)/k$ BSLR (or RGSLR).

1.4 Good designs for experiments

Good designs for experiments are usually sought among connected designs. In a connected design, all elementary contrasts of treatment effects are estimable: see, for example, Rao (1958) and Godolphin (2019a). If these contrasts can be estimated with the same variance, then the design is said to be balanced. In particular, such design is said to be variance balanced. The notion of connectedness of a design can also be viewed as having a chain of treatments in which every pair of adjacent treatments in the chain concurs in at least one block of the design: see, for example, Tianyao and Yu (2010). For a design with v treatments, this is akin to having a v–gon (a polygon with v vertices), where the vertices correspond to the treatments and each pair of adjacent vertices which forms an edge on the v–gon appears as a block in the design. Hence, from each vertex of the v–gon, every other vertex can be reached via an edge or a sequence of edges. Typical of a connected design is the balanced incomplete-block design (BIBD), which when it exists, is known to be optimal over a range of criteria among all incomplete block designs of its size. The canonical efficiency factors of a BIBD are all equal: see, for example John and Williams (1982). This equality of all its canonical efficiency factors implies it is efficiency balanced. Hence a BIBD is both variance balance and efficiency balanced. Moreover, an efficiency balanced design that is equireplicate (having equal replication of its treatments) is variance balanced: see Tianyao and Yu (2010).

Apart from connectedness, some other nice properties of a design include, amongst others, binarity, equireplication (mentioned above) and equal-sized blocks. A BIBD possesses all these properties. Designs which possess the property of binarity are called binary designs. Moreover, designs with equal-sized blocks are also known as proper designs.

For some parameter sets that a BIBD does not exist, search is usually made among regular-graph designs (RGDs) which are close to balanced designs in the sense that no two pairs of distinct treatments differ in their concurrence counts by more than unity. It has been conjectured in John and Mitchell (1977) that a D–optimal (or A– optimal or E–optimal) incomplete-block design is among RGDs, provided a RGD exists. However, Jones and Eccleston (1980) found some counterexamples regarding A-optimality; and Constantine (1986) refuted it with respect to the E-optimality. In the light of these, Cheng (1992) confirms the conjecture for sufficiently large number of blocks

Moreover, by the conjecture in John and Mitchell (1977), conjecture 2 of John and Williams (1982) implies that, if a RGD exists, then an A–optimal RGD is also D-optimal and vice versa. However, Cakiroglu (2018) gives a counterexample to this (though under a given situation), where the A-best RGD is found to be different from the D-best design for the case where the RGD is not extended by adding copies of a BIBD to it; but when the RGD is extended with sufficient copies of a BIBD, the D-best RGD is found to remain D-best BIBD-extended RGD and also becomes the A-best BIBD-extended RGD, hence no counterexample for the case where the RGD is BIBD-extended.

Concerning designs that are obtained via complementation; it is conjectured that an A–optimal (or D–optimal) design has as its complement an A–optimal (or D–optimal) design over an appropriate class of complementary designs. However, the conjecture holds if the 'parent' design is a BIBD: see John and Williams (1982).

Thus, by the discussions above, when a BSLR exists, it is optimal over all SLRs of its size: see Uto and Bailey (2020), and its complementary SLR, which is also a BSLR is optimal over an appropriate class of SLRs. Moreover, when a BSLR does not exist, good SLRs can be found among BIBD-extended RGSLRs and their complements also produce good SLRs.

1.5 Organization of the chapters

This thesis consists of seven chapters; each chapter contains an introduction. Chapter 2 gives a background of the work containing some definitions and discussions of some basic concepts and also some related works on the class of designs considered in this thesis are brought to view together with some information on how we approached the work.

The main work that has been done can be divided into two major parts, viz, BSLRs and NBSLRs, where we consider RGSLRs. These two parts are captured in four chapters, viz, chapters 3, 4, 5 and 6. Chapters 3 and 4 which constitute one part of the work are dedicated to BSLRs. In particular, Chapter 3 concerns BSLRs with $k = 2$ and the relevant algorithms are given for each case regarding the nature of the value of v, that is, when v is even and when it is odd. Designs with $k > 2$ are discussed in Chapter 4.

Moreover, the second part of the work spans Chapters 5 and 6 where NBSLRs with $k = 2$ are considered in Chapter 5 while Chapter 6 concerns NBSLRs with $k > 2$. The relevant algorithms are also given.

The last chapter, gives a summary of the main results and some general comments with some conclusions and suggestions for further work. Moreover, separate tables of parameters that can give designs that are BSLRs and RGSLRs are also presented.

Chapter 2

Historical Background

2.1 Introduction

In this chapter, we give a historical background of this work. Some terms and concepts associated with the work are discussed. A description of the design under investigation, the semi-Latin rectangle alongside some of its applications in the design of various experiments ranging from Agriculture to the industry are given. Again, we show how the semi-Latin rectangle is related to some other designs–the Latin squares and semi-Latin squares, which could be derived from it; and also discuss some concepts related to the efficiency and optimality of designs. Finally, a related work by Bailey and Monod (2001) which stands out as the pioneer work on semi-Latin rectangles is quoted, and also a recent work by Uto and Bailey (2020) which introduces balanced semi-Latin rectangles is also quoted, alongside our direction of research.

2.2 Definitions and Notations

Definition 2.2.1. Let Ω denote the set of plots of an experiment, and $\mathcal{V}$ the set of associated treatments. Denote by $|\Omega|$, the number of plots and $|\mathcal{V}| = v$, the number of treatments. Again, let τ denote a function such that

$$\tau : \Omega \to \mathcal{V}$$

$\forall\ \omega \in \Omega$, there exists $t \in \mathcal{V}$ such that $\tau(\omega) = t$. Then τ is said to be a design.

We note that treatment t is allocated to plot ω, or equivalently, plot ω receives treatment t. Thus, a design, indeed, specifies the allocation of treatments to plots in an experiment.

Definition 2.2.2. Let $\mathcal{V}$ denote the treatment set of a design, d with cardinality, $|\mathcal{V}| = v$. Suppose the v treatments are allocated to the plots, arranged in b blocks, each being of size k_j, $j = 1, 2, ..., b$; and each treatment is replicated r_i times, $i = 1, 2, ..., v$. Then d is said to be a block design.

Remark. In particular, d is said to be a complete block design if $k_j = v\ \forall j$, and all the treatments appear in each block. But if $k_j < v$, then it is described as an incomplete block design.

Definition 2.2.3. Let d be a block design. Denote by n_{ij}, the number of times that the ith treatment appears in the jth block, or equivalently, the number of plots in block j that receive the ith treatment. Then d is called a binary design if $n_{ij} = 0$ or 1.

We note that, if d is binary, its incidence matrix, $N = (n_{ij})$ has all its entries as 0s and 1s: see, for example, John and Williams (1995, chapter 1). Furthermore, binary designs have maximal trace of their information matrix, hence an (M, S)-optimal design can be found among them. The (i, j)th element, $i \neq j$ of the concurrence matrix of such class of designs gives the number of blocks in which the ith treatment concurs with the jth treatment: see, for example, John and Williams (1982).

Definition 2.2.4. Let d denote a block design with b blocks, and k_j, $j = 1, 2, ..., b$ the number of plots in the jth block, called the size of the jth block. Suppose $k_j = k\ \forall j$. Then d is said to be a proper block design with block size k.

Definition 2.2.5. Given a design, d with treatment set $\mathcal{V}$. Denote the cardinality of $\mathcal{V}$, $|\mathcal{V}| = v$. Let r_i, $i = 1, 2, ..., v$ denote the number of times the ith treatment appears in the design, called the number of replications (or the replication number) of the ith treatment. Suppose $r_i = r\ \forall i$. Then d is called an equireplicate design with replication number r.

Definition 2.2.6. A binary, proper and equireplicate design $d(v, k, \lambda)$ is said to be a balanced incomplete block design (BIBD) if every pair of treatments appear together (or concur) in a constant number, λ of blocks. The design (or combinatorial) parameters v, k, λ denote the respective number of treatments, block size and between-treatment concurrences, called the concurrence number: see, for example, Caliński and Kageyama (2003, Chapter 6), Morgan (2007), Raghavarao and Padgett (2005, Chapter 4), Stinson (2004, Chapter 1), Hedayat et al. (1995), Abel (1994), Hanani (1961), and Bose (1939) for various descriptions/discussions of this design. Its parameters satisfy the combinatorial properties:

$$vr = bk \tag{2.1}$$

$$\lambda(v-1) = r(k-1) \tag{2.2}$$

where $v \leq b$.

The parameters b and r denote the numbers of blocks and replications of each treatment (or the replication number of each treatment) in the design, respectively; and could be obtained from (2.1) and (2.2), simultaneously with known values of v, k and λ. The restriction, $v \leq b$ is called Fisher's inequality.

Definition 2.2.7. Given a BIBD, d with parameters v, b, r, k, λ. Suppose $v = b$, and consequently $r = k$, such that $\lambda(v-1) = k(k-1)$. Then d is said to be a symmetric BIBD.

Remark. If d is a symmetric (v, b, r, k, λ)-BIBD, then any pair of blocks in d contain λ number of treatments in common; which is a useful result for constructing new BIBDs from old ones: see, for example, Stinson (2004, chapter 2).

Definition 2.2.8. Let $d(v, b, r, k, \lambda)$ denote a balanced incomplete block design (BIBD). Then d is said to be α-resolvable if its b blocks can be subdivided into s groups called superblocks (or α-resolution sets), each containing b^* blocks such that in each superblock (or α-resolution set) every treatment is replicated exactly α (≥ 1) times. This leads to the following restrictions on the parameters, *viz*: $b = sb^*$, $r = s\alpha$, $v\alpha = kb^*$, $b\alpha = rb^*$.

In particular, d is affine α-resolvable if, in addition to being α-resolvable, it satisfies further condition that every pair of distinct blocks from the same superblock (or α-resolution set) contain the same number, say, w_s, of treatments in common, and every pair of blocks from distinct superblocks contain the same number, say, b_s, of treatments in common, where $w_s = k(\alpha - 1)/(b^* - 1) = k - r + \lambda$, and $b_s = k\alpha/b^* = k^2/v$.

We note that if $\alpha = 1$, d reduces to a 1-resolvable and affine 1-resolvable design, respectively, which for simplicity are called resolvable and affine resolvable designs: see, for example, Caliński and Kageyama (2003, chapter 9), Kadowaki and Kageyama (2009), and Raghavarao and Padgett (2005, chapter 4) for discucussions on the general case of the concept of α (≥ 1)-resolvability, as well as that of affine α-resolvability; and Bailey et al. (1995), Bose (1942) for the particular case of these concepts, when $\alpha = 1$.

Definition 2.2.9. A binary, equireplicate block design is said to be a regular graph design (RGD) if no two pairs of distinct treatments differ in their concurrences by more than unity, in absolute terms. Moreover, there exists two distinct treatment concurrences, λ and $\lambda + 1$ in the design: see, for example, Cakiroglu (2018), Cheng (1978), John and Mitchell (1977).

Let treatments i and i', $i \neq i'$ concur in $\lambda_{ii'}$ blocks. Denote $\lambda_{ii'}$ by λ. Suppose (l, l') is another distinct pair of treatments with concurrences $\lambda_{ll'}$. Then $\lambda_{ll'}$ needs to be either λ or $\lambda + 1$, assuming $\lambda_{ii'} \leq \lambda_{ll'}$ such that $|\lambda_{ii'} - \lambda_{ll'}| = 0$ or 1.

Remark. On the contrary, suppose $\lambda_{ii'} \geq \lambda_{ll'}$, then $\lambda_{ll'}$ would be either λ or $\lambda - 1$, and the absolute difference in the treatment-concurrences between distinct pairs of treatments is invariant.

However, we observe that, John and Mitchell (1977) and John and Williams (1982) consider a BIBD to belong to the class of RGDs

Definition 2.2.10. Let $\mathcal{V}$ denote the set of treatments associated with a design, d. Denote by $|\mathcal{V}| = v$, the number of treatments in d. Suppose $p, q \in \mathbb{Z}$ $(p, q > 1)$ denote the respective numbers of rows and columns of a p-by-q array. Let the v treatments be allocated to plots, grouped in two directions, *viz*, the rows and columns, denoting two blocking factors such that they form this array. Then d is said to be a row-column design with p rows, q columns and pq row-column intersections called cells/ blocks if k (≥ 1) treatments corresponding to the plots are embedded in each row-column intersection.

Definition 2.2.11. An $n \times n$ Latin square is an arrangement of $v = n$ treatments in n rows and n columns, thereby forming an n-by-n array with n^2 cells, such that each cell contains $k = 1$ plot, accommodating one treatment, and each treatment appears exactly once in each row and exactly once in each column.

Remark. A Latin square is said to be in standard form if its first row and column contain treatments that appear in a natural order. Each cell of a Latin square denotes a row-column intersection. Again, an $n \times n$ Latin square is also called a Latin square of order n.

Definition 2.2.12. Let Ω denote a set of plots, with cardinality, $|\Omega| = kn^2$, and $\mathcal{V}$ the set of treatments with cardinality, $|\mathcal{V}| = kn$. Suppose the kn treatments are arranged in an n-by-n array consisting of the kn^2 plots displayed in n rows and n columns such that each row-column intersection (block) contains $k(> 1)$ plots, hence k distinct treatments. Then this arrangement constitutes an $(n \times n)/k$ semi-Latin square if each treatment appears exactly once in each row, and exactly once in each column.

Definition 2.2.13. Let h, p, k, n_r and $n_c \in \mathbb{Z}$. Suppose $k \geq 1$; $h, p > 1$; and $v|kh, kp$. Then, we define an $(h \times p)/k$ semi-Latin rectangle to be a row-column design in which v treatments are arranged into h rows and p columns, where each row-column intersection (block) contains k treatments, and each treatment appears the same number, n_r of times in each row, and also the same number, n_c of times in each column.

We note that $n_c \leq n_r \iff h \leq p$. Similarly, $n_c > n_r \iff h > p$

Remark. The definition of semi-Latin rectangle given in Bailey and Monod (2001) does not accommodate $h = p$, as $h < p$ is assumed; and by virtue of this, $n_r > n_c$.

In a semi-Latin rectangle, each row-column intersection constitutes a block. By ignoring the rows and columns classification, its quotient block design results. Semi-Latin rectangles exhibit the property of orthogonality with respect to the row and column strata: see Bailey and Monod (2001). Its treatments are orthogonal to the row strata, in the sense that each treatment appears the same number, n_r of times in each row. Similarly, the treatments are also orthogonal to the column strata since each treatment occurs the same number, n_c of times in each column. This design is also, in general, n_r- and n_c-resolvable with regards to the rows and columns, respectively.

Within the sphere of semi-Latin rectangles, there exist some designs whose quotient block design gives a balanced incomplete block design (BIBD), which we shall designate balanced semi-Latin rectangle (BSLR). Again, a pair of semi-Latin rectangles could possess the property of isomorphism; such semi-Latin rectangles are known as isomorphic semi-Latin rectangles.

Definition 2.2.14. A balanced semi-Latin rectangle (BSLR) is a semi-Latin rectangle with the property that its quotient block design forms a balanced incomplete-block design (BIBD). It is, indeed, a semi-Latin rectangle with an additional property of balance.

Definition 2.2.15. Suppose Γ_1 and Γ_2 denote two semi-Latin rectangles. Then Γ_1 and Γ_2 are isomorphic if there exists a sequence of permutations involving the rows, columns and treatments such that when these are applied to one of them it leads to the other.

Definition 2.2.16. Let $(G, +)$ denote a finite Abelian group of order v with 0 as the identity element. Suppose $G = (\mathbb{Z}_v, +)$, where $\mathbb{Z}_v$, is the set of integers, reduced mod v. Suppose further that k and λ satisfy $2 \leq k < v$, $\lambda > 0$. Let $S = \{a_1, a_2, ..., a_k\} \subseteq \mathbb{Z}_v$, with cardinality, $|S| = k$. If the multiset $\{(a_i - a_{i'}) \bmod v : a_i, a_{i'} \in S, i \neq i'\}$, the set of the differences between all possible pairs of elements of S contains each non-zero element of $\mathbb{Z}_v$ exactly λ times, then S is said to be a difference set (or a perfect difference set) of cardinality k and index λ, or simply, a (v, k, λ)-difference set for $\mathbb{Z}_v$. In particular, $\lambda(v-1) = k(k-1)$ if the (v, k, λ)-difference set, S exists.: see, for example, Stinson (2004, chapter 3).

We note that if S is a (v, k, λ)-difference set for $\mathbb{Z}_v$, it generates a (v, k, λ)-symmetric BIBD by a cyclic development of it, *via* successive addition of each element of $\mathbb{Z}_v$ to the elements of S, reduced mod v. Again, the set, S forms the initial block of the BIBD.

Remark. A generalization of the concept of difference set gives the difference family.

Suppose $\mathbb{Z}_v$ is as defined before, and $0, v, k, \lambda$ satisfy the conditions in definition 2.2.16. Let $[S_1, S_2, ..., S_l]$ be such that $S_i \subseteq \mathbb{Z}_v$, for $i = 1, 2, ..., l$ and $|S_i| = k$, $\forall i \in \{1, 2, ..., l\}$, and the multiset union, $\bigcup_{i=1}^{l}\{(a_u - a_{u'}) \bmod v : a_u, a_{u'} \in S_i, u \neq u'\}$ contains each non-zero element of $\mathbb{Z}_v$ exactly λ times. Then, the sets $S_1, S_2, ..., S_l$ together constitute a (v, k, λ)-difference family for $\mathbb{Z}_v$. In particular, $l = \frac{\lambda(v-1)}{k(k-1)} \in \mathbb{Z}$, or equivalently, $\lambda(v-1) \equiv 0 (\bmod\ k(k-1))$ if the (v, k, λ)-difference family, $[S_1, S_2, ..., S_l]$ exists.

Definition 2.2.17. Let $\mathbf{c}$ denote a non-zero n-component vector of coefficients; and $\boldsymbol{\tau}$, an n-component vector of parameters. Suppose $n \geq 2$. Define $\mathbf{c}' = (c_1, c_2, ..., c_n)$, and $\boldsymbol{\tau} = (\tau_1, \tau_2, ...\tau_n)'$. Again, let

$$\mathbf{c}'\boldsymbol{\tau} = \sum_{i=1}^{n} c_i\tau_i \tag{2.3}$$

denote a linear combination in $\boldsymbol{\tau}$, where τ_i is the effect of treatment i. Then $\mathbf{c}'\boldsymbol{\tau}$ is a contrast if

$$\mathbf{c}'\mathbf{1} = \sum_{i=1}^{n} c_i = 0$$

where $\mathbf{1}$ is an n-component vector of 1s.

Remark. $\mathbf{c}'\tau$ is said to be a simple (or elementary) contrast if there are only two non-zero entries, 1 and -1 in the coefficient vector, $\mathbf{c}'$. Again, $\mathbf{c}'\tau$ is said to be normalized if

$$\mathbf{c}'\mathbf{c} = \sum_{i=1}^{n} c_i^2 = 1$$

.

Given $\mathbf{c}'\boldsymbol{\tau}$ as defined in (2.3). Suppose $\mathbf{l}'\boldsymbol{\tau}$ is another contrast in $\boldsymbol{\tau}$, where $\mathbf{l}' = (l_1, l_2, ..., l_n)$, such that $\mathbf{l}'\boldsymbol{\tau} = \sum_{i=1}^{n} l_i\tau_i$, and $\mathbf{l}'\mathbf{1} = \sum_{i=1}^{n} l_i = 0$. If $\mathbf{c}'\mathbf{l} = 0$, or equivalently, in scalar notation $\sum_{i=1}^{n} c_i l_i = 0$, then $\mathbf{c}'\boldsymbol{\tau}$ and $\mathbf{l}'\boldsymbol{\tau}$ are said to be orthogonal.

Definition 2.2.18. Let d be a (v, b, r, k)-design. Then d is said to be connected if all the elementary contrasts of its treatment parameters are estimable.

Remark. A design that is not connected is said to be disconnected.

2.3 Preliminaries

Semi-Latin rectangles (SLRs) form an important class of row-column designs with interesting/attractive combinatorial properties. From the general perspective, row-column designs admit two systems of blocks: the rows and columns, hence control heterogeneity in the experimental units which could have some effect on the response, in two directions, as well, corresponding to the rows and columns.

Most of the classical row-column designs, which include, amongst others, the Latin squares, Youden Squares, and generalized Youden designs have only one plot in each row-column intersection (cell), hence just one treatment can be applied to each cell, which could lead to the experimenter spending more resources–materials, time and cost in performing his experiment using such designs, which would involve more replications of each treatment, compared to when he uses a similar design with more plots in each row-column intersection (though this may involve fewer number of replications of each treatment): see, for example, Datta et al. (2017), Datta et al. (2014, 2015, 2016), Dash et al. (2014), Donev (1998), Shah and Sinha (1996), John and Williams (1995, Chapter 5), Shah and Sinha (1989, Chapter 4), Agrawal (1966), for a general description of row-column designs; Keedwell and Dénes (2015), Ai et al. (2013), Raghavarao and Padgett (2005, Chapters 4 & 9), Raghavarao (1971, Chapter 1), Bose et al. (1960), Bose and Nair (1941), Bose (1938), for several discussions on Latin squares; Preece (1996), John and Williams (1995, Chapter 5), Shrikhande (1951), Raghavarao (1971, Chapter 6) for discussions on Youden squares; and Colbourn (1996), Ash (1981), Kiefer (1975), Ruiz and Seiden (1974) for generalized Youden designs.

The semi-Latin rectangle is one of such designs with multiple, in general, $k(> 1)$ plots in each row-column intersection, called a block, thereby allowing for more treatment allocation to each block in the design, and subsequently saving materials, time and cost, geared towards enhancing efficient use of resources in a comparative experiment. For instance, an experimenter may wish to compare eight treatments. To do this, he will need a total of sixty-four plots, if he uses a Latin square, each treatment being replicated eight times. But he could do the same experiment with just thirty-two plots using a semi-Latin rectangle that has fewer number of replications, four, say, for each treatment: see Figure 2.3, which offers a 50% reduction in the experimental material requirement.

Semi-Latin rectangles have been found useful in agricultural experiments, such as plant disease experiments; food sensory experiments; as well as consumer testing experiments. For instance, a $(4 \times 8)/2$ semi-Latin rectangle involving eight plants and pairs of half-leaves at four heights has been used for experiment on tobacco plants at Rothamsted Experimental Station to check whether a mechanism to inhibit tobacco mosaic virus had been transferred to following generations given that it was present in certain transgenic plants. Eight treatments were used, where each was a solution made from an extract of one of the offspring plants. In particular, experiments on plant diseases often use half-leaves as experimental units, and the numbers of leaf-heights (h), and plants (p) used, in general, correspond to the numbers of rows and columns of the design, respectively ($h < p$): see Bailey and Monod (2001).

In food sensory experiments, there are p panellists, and h food-tasting sessions, where the treatments are the various food items available for tasting. Each panellist is made to taste k items of food in each of h sessions, where $h \ll p$, and $k = 2$ or 3. For the consumer testing experiments, p consumers are available for the experiment, which is to be performed in h weeks. Various brands of a given consumer good to be tested constitute the treatments. Each consumer tests k brands of the product each week: see Bailey and Monod (2001). Moreover, Bailey (1992) describes some experimental situations where a semi-Latin square can be used. In similar experimental situations where the number of rows is not identical to the number of columns, a semi-Latin rectangle becomes a useful design.

We give a few illustrative examples of semi-Latin rectangles in Figures 2.1, 2.2 and 2.3. Figures 2.1 and 2.2 can be found in Bailey and Monod (2001). In Figure 2.1, there are four treatments arranged in eight blocks of two plots, each which are embedded in a 2-by-4 array. This design, has, indeed, a simple orthogonal block structure, viz $(2 \times 4) \rightarrow 2$: see Nelder (1965), but we shall denote this by $(2 \times 4)/2$ in conformity with modern literature. That is, there are two rows and four columns, with two plots in each row-column intersection of the design. Each treatment appears twice in each row, and exactly once in each column. The design in Figure 2.2 is a $(4 \times 8)/2$ semi-Latin rectangle. It has four rows, eight columns and two plots in each block. There are eight treatments; each appears twice per row and once per column, just like the design in Figure 2.1. The randomized form of the design in Figure 2.2 was used for the experiment on tobacco plants mosaic virus at the Rothamsted Experimental Station: see Bailey and Monod (2001). The design in Figure 2.3 is a $(2 \times 4)/4$ semi-Latin rectangle. It has eight treatments just like the one in Figure 2.2, but there are two rows, four columns, and four plots in each block. Each treatment makes an appearance two times in each row, and once in each column, just like the designs in Figures 2.1 and 2.2.

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

Figure 2.1: Semi-Latin rectangle for four treatments in blocks of size two

4	5	5	6	6	7	7	0	0	1	1	2	2	3	3	4
1	7	2	0	3	1	4	2	5	3	6	4	7	5	0	6
0	3	1	4	2	5	3	6	4	7	5	0	6	1	7	2
2	6	3	7	4	0	5	1	6	2	7	3	0	4	1	5

Figure 2.2: Semi-Latin rectangle for eight treatments in blocks of size two

2.4 Relationship with Latin squares and semi-Latin squares

An immediate generalization of the Latin square is the semi-Latin square, which is further generalized by the semi-Latin rectangle.

The Latin square has the same number of rows as columns, and each treatment appears equally often (just once) in each row, and also once in each column. Again, it has only one experimental unit (plot) in each row-column intersection (cell), hence only one treatment can be applied there; while the semi-Latin square, like the semi-Latin rectangle has multiple, in general, k (> 1) units in each row-column intersection (block), though with the restriction/limitation of equal number of rows as columns, as well as the appearance of each treatment, exactly once in each row and also in each column: see, for example, Soicher (2013), Parsad (2006), Bedford and Whitaker (2001), Bailey and Royle (1997), Bailey and Chigbu (1997), Bailey (1992, 1988), Preece and Freeman (1983), as well as Rojas and White (1957) for several discussions on semi-Latin squares.

Let Γ denote an $(h \times p)/k$ semi-Latin rectangle. Suppose $h = p = n$, say, and $n_r = n_c = 1$; where n_r and n_c denote the respective number of times each treatment appears in each row and column of the semi-Latin rectangle. Then Γ reduces to an $(n \times n)/k$ semi-Latin square. Denote this design Δ. Furthermore, if k, the number of plots in each row-column intersection of Δ is equal to 1, then it reduces to an $n \times n$ Latin square.

Hence, the semi-Latin rectangle is a generalization of the Latin square/semi-Latin square, or equivalently, the Latin squares/semi-Latin squares are special cases of the semi-Latin rectangle.

1	2	3	4	5	6	7	8	1	2	5	6	3	4	7	8
5	6	7	8	1	2	3	4	3	4	7	8	1	2	5	6

Figure 2.3: Semi-Latin rectangle for eight treatments in blocks of size four

2.5 The Quotient block design of a semi-Latin rectangle and some related matrices

Definition 2.5.1. Given an $(h \times p)/k$ semi-Latin rectangle, Γ. Denote by Γ^*, the incomplete block design obtained by ignoring the rows and columns of Γ. Then Γ^* is said to be the quotient block design (QBD) of Γ.

We note that Γ^* is indeed a (v, b, r, k)-design, where $v = kh/n_c = kp/n_r$, $b = hp$, $r = hn_r = pn_c$, and $k < v$.

Definition 2.5.2. Let d denote a (v, b, r, k) design. Suppose $N = (n_{ij})$ is a matrix, where $i = 1, 2, ..., v;\ j = 1, 2, ..., b$. If n_{ij} is the number of times that the ith treatment appears in the jth block, such that $n_{ij} \in \mathbb{Z},\ n_{ij} \geq 0\ \forall\, i, j$, then $N = (n_{ij})$ is said to be the incidence matrix of Γ^*.

Remark. The incidence matrix, N of d is a v-by-b treatments-by-blocks matrix with non-negative entries. The structure of d is completely determined by its incidence matrix, N: see, for example, Jacroux (1980). In particular, if d is binary, $n_{ij} = 0$ or 1.

In general, the sum of the entries on each row of N gives the number of replications of the treatment corresponding to that row. Again, if this sum is a constant for all the rows, it follows that d is an equireplicate block design.

Similarly, the sum of the entries on each column of N gives the size of the block corresponding to that column. Also, if this sum gives a constant value for all the columns, then d is a proper block design.

Definition 2.5.3. Given a v-by-b incidence matrix, N of a (v, b, r, k) design, d. Let NN' denote a matrix of the product of N and N', where N' is the transposed matrix of N. Then NN' is said to be the concurrence matrix of d.

Remark. The concurrence matrix, NN' of d is a square matrix of order v. If d is binary, then the ith leading diagonal entry of NN' is, simply, the number of times that the ith treatment appears in the design, called the number of replications, r_i of the ith treatment, $i = 1, 2, ..., v$; while each off-leading diagonal entry, $(i, i')\ , i \neq i'$ corresponds to the number of blocks that treatments i and i' appear together, called the number of concurrences between treatments i and i', and denoted $\lambda_{ii'}$. Furthermore, if d is, in addition, equireplicate, each leading diagonal entry has the same value, r, say.

Again, if each leading diagonal entry has a constant value, r, and in addition each pair of its off-leading diagonal entries differ by not more than 1, in absolute terms, then d is, thus, a regular graph design (RGD). Suppose d is an RGD and each off-leading diagonal entry is equal to the other, then it is a balanced block design (BBD): see, for example, Jacroux (1980).

Definition 2.5.4. Let d be a binary, proper and equireplicate (v, b, r, k)-design. Suppose L is the symmetric, non-negative definite, and zero-row-sums as well as zero-column-sums matrix of order v defined by

$$L = rI - k^{-1}NN' \tag{2.4}$$

where I is an identity matrix of order v. Then L is said to be the information matrix of d.

The scaled version of the information matrix given in (2.4) is thus

$$L^* = r^{-1}L = I - (rk)^{-1}NN' \tag{2.5}$$

which is called the scaled information matrix: see, for example, Soicher (2013).

Remark. The information matrix of a design is not of full rank. It has rank of at most $v-1$, that is $\text{rank}(L) = \text{rank}(L^*) \leq v-1$. In particular, the equality holds if the design is connected, such that the upper bound of its rank is attained. Hence $\text{rank}(L) = \text{rank}(L^*) = v-1$, if d is connected.

Conversely, the strict inequality holds if d is disconnected; such that $\text{rank}(L) = \text{rank}(L^*) < v-1$.

Definition 2.5.5. Let A denote an $m \times n$ matrix; and A^- a matrix of order $n \times m$ satisfying

$$AA^-A = A \tag{2.6}$$

Then, A^- is said to be a generalized inverse of A: see, for example, John and Williams (1995, Appendix A) and Searle (1982, chapter 8).

Remark. A^- is also known as g-inverse, pseudo inverse, or conditional inverse: see, for example, John and Williams (1995, Appendix A).

For a given A, there could be many generalized inverses, A^- since there exist many A^- that could satisfy (2.6): see, for example, Searle (1982, chapter 8). Hence, a generalized inverse of a singular matrix is not, generally, a unique matrix.

However, there exists a unique version of the generalized inverse, A^+ of A, called the Moore-Penrose generalized inverse.

Definition 2.5.6. Let A be an arbitrary $m \times n$ real matrix. Suupose there exists a unique $n \times m$ real matrix, A^+ satisfying the following conditions:

$$AA^+A = A \tag{2.7}$$

$$A^+AA^+ = A^+ \tag{2.8}$$

$$(AA^+)' = AA^+ \tag{2.9}$$

$$(A^+A)' = A^+A \tag{2.10}$$

Then A^+ is said to be the Moore-Penrose generalized inverse of A: see, for example, Courrieu (2005), John and Williams (1995, Appendix A), Searle (1982, chapter 8), Plemmons and Cline (1972), as well as Penrose (1955).

Remark. The four conditions given in (2.7), (2.8), (2.9), and (2.10) are usually called the Penrose conditions: see, for example, Searle (1982, chapter 8). For a given matrix, A^+ to be called the Moore-Penrose generalized inverse of another, A, it must satisfy the four Penrose conditions; and it is unique for a given A. Since (2.7) is identical to (2.6) if the different notations, A^+ and A^- for the generalized inverses are ignored, it follows that A^+ is a generalized inverse which satisfies three extra conditions, *viz*, (2.8), (2.9), and (2.10).

If a given generalized inverse satisfies (2.8), in addition, then it is called a reflexive generalized inverse: see, for example, Searle (1982, chapter 8).

Suppose A is non-singular, then its inverse is unique, and this is identical to the regular inverse. Thus, $A^- = A^{-1} = A^+$, where A^{-1} is the regular inverse: see, for example, Searle (1982, chapter 8).

Again, suppose B denote an information matrix of order v for a (v, b, r, k)-design. Provided d is connected, then $(B + aJ)$ is non-singular, where $a \neq 0$ is any real scalar, and J, a matrix of 1s, also of the same order as B. Consequently, $(B + aJ)^{-1}$ is another version of generalized inverse of B: see, for example, John (1980, chapter 2) and Shah (1959).

Definition 2.5.7. Let $\mathbf{X} \in \mathbb{R}^n$ be a random vector. Suppose $F = (f_{ij})$, $i, j = 1, 2, ..., n$ is a positive semi-definite matrix such that

$$f_{ij} = \begin{cases} \text{Cov}(X_i, X_j), & i \neq j \\ \text{Var}(X_i), & i = j \end{cases}$$

where $\text{Cov}(X_i, X_j)$ is the covariance between X_i and X_j, $i \neq j$; and $\text{Var}(X_i)$ is the variance of X_i. Then $F = (f_{ij})$ is said to be the variance-covariance matrix, or simply, the covariance matrix of $\mathbf{X}$.

Remark. Suppose d is a (v, b, r, k)-design. Suppose further that B denote the information matrix of d. Let E denote any generalized inverse of B. Then E is a variance-covariance matrix for estimating the variance of treatment contrasts $\mathbf{a}'\boldsymbol{\tau}$ in d, where $\text{Var}(\widehat{\mathbf{a}'\boldsymbol{\tau}}) = \sigma^2\mathbf{a}'E\mathbf{a}$: see Morgan (2007).

2.6 Design Efficiency

The efficiency of a design of a given size is concerned with the measure of the gain (or loss) resulting from the use of the design. Blocking, which reduces the error variance; as well as small block sizes often lead to a gain in efficiency compared to when large block sizes are used: see, for example, John and Williams (1995, chapter 2). A design in which the gain is high (or the loss is low) is, thus, an efficient design; and this is achievable by binary designs.

Equivalently, design efficiency could be viewed as the performance ability (or inability) of a design in estimating elementary contrasts of its treatment parameters with minimal variances, which leads to a high (or low) precision in the estimation.

Definition 2.6.1. Given a (v, b, r, k)-design, d. Let τ_i be a parameter denoting the effect of the ith treatment, $i = 1, 2, ..., v$. Again, let $\tau_{i'}$, $i \neq i'$ denote the effect of another treatment, i'. Then, the difference, $\tau_i - \tau_{i'}$, between the effects of treatments i and i', $i \neq i'$ is an elementary contrast.

Let $\widehat{\tau_i - \tau_{i'}}$ denote the estimator of this contrast. Denote $\widehat{\tau_i - \tau_{i'}}$ by $\hat{\tau}_i - \hat{\tau}_{i'}$, and its variance by $\mathrm{Var}(\hat{\tau}_i - \hat{\tau}_{i'})$. If L^{*-} is a generalized inverse of the scaled information matrix, $L^* = I - (rk)^{-1}NN'$ of d, then

$$\mathrm{Var}(\hat{\tau}_i - \hat{\tau}_{i'}) = (L^{*-}_{ii} + L^{*-}_{i'i'} - 2L^{*-}_{ii'})\sigma^2/r \tag{2.11}$$

where L^{*-}_{uu}, $u = i, i'$ is the uth leading diagonal entry of L^{*-}, $L^{*-}_{ii'}$ is the (i, i')th entry of the same matrix, L^{*-} and σ^2 is the error variance: see, for example, Bailey (2009), Bailey (2004, chapter 4), as well as John and Williams (1995, chapter 2).

Remark. In general, given a (v, b, r, k)-design, d with the scaled information matrix, L^*, let $\mathbf{c}'\boldsymbol{\tau} = \sum_{i=1}^{v} c_i\tau_i$ be a contrast in $\boldsymbol{\tau}$, and $\mathbf{c}'\hat{\boldsymbol{\tau}} = \sum_{i=1}^{v} c_i\hat{\tau}_i$, its estimator. Then

$$\begin{aligned}\mathrm{Var}(\mathbf{c}'\hat{\boldsymbol{\tau}}) = \mathrm{Var}(\sum_{i=1}^{v} c_i\hat{\tau}_i) &= \sigma^2 \frac{\sum_{i=1}^{v} c_i^2}{r} \frac{\mathbf{c}'L^{*-}\mathbf{c}}{\sum_{i=1}^{v} c_i^2} \\ &= (\mathbf{c}'L^{*-}\mathbf{c})\frac{\sigma^2}{r} \end{aligned} \tag{2.12}$$

where $\dfrac{\sum_{i=1}^{v} c_i^2}{r}$ is a scale factor, and $\frac{\mathbf{c}'L^{*-}\mathbf{c}}{\sum_{i=1}^{v} c_i^2}$ a scalar which depends on the coefficient vector $\mathbf{c}' = (c_1, c_2, ..., c_v)$ and the design, d but not on the experimental material: see, for example, Bailey and Royle (1997). It is obvious from (2.12) that $\mathrm{Var}(\mathbf{c}'\hat{\boldsymbol{\tau}})$ is a product of these two quantities and the error variance, σ^2.

In particular, if the vector $\mathbf{c}$ (or $\mathbf{c}'$) has only two non-zero entries, 1 and -1, then $\mathbf{c}'\boldsymbol{\tau} = \sum_{i=1}^{v} c_i\tau_i$ reduces to $\tau_i - \tau_{i'}$, $i \neq i'$ which is an elementary contrast. Consequently, $\mathrm{Var}(\mathbf{c}'\boldsymbol{\tau}) = \mathrm{Var}(\hat{\tau}_i - \hat{\tau}_{i'}) = (L^{*-}_{ii} + L^{*-}_{i'i'} - 2L^{*-}_{ii'})\sigma^2/r$, obtained by using (2.12), which result is identical to (2.11).

Again, if d is connected, all the elementary contrasts of its treatment parameters are estimable. Hence there are $\binom{v}{2} = \frac{v(v-1)}{2}$ estimable distinct elementary contrasts.

Definition 2.6.2. Suppose d is a connected (v, b, r, k)-design. Let $L^{*-}_{ii} + L^{*-}_{i'i'} - 2L^{*-}_{ii'}$ in (2.11) be denoted $v_{ii'}$, $i \neq i'$ such that (2.11) becomes

$$\mathrm{Var}(\hat{\tau}_i - \hat{\tau}_{i'}) = v_{ii'}\sigma^2/r$$

Let

$$\overline{\mathrm{Var}(\hat{\tau}_i - \hat{\tau}_{i'})} = (\sigma^2/r)\bar{v}_g = (\sigma^2/r)(2\sum_{i}\sum_{i'>i} v_{ii'})/v(v-1)$$

where $\bar{v}_g$ is the average over the values of $v_{ii'}$ for all distinct pairs, i and i' of treatments. Denote $\overline{\mathrm{Var}(\hat{\tau}_i - \hat{\tau}_{i'})}$ by $\bar{v}$. Then

$$\bar{v} = (\sigma^2/r)(2\sum_{i}\sum_{i'>i} v_{ii'})/v(v-1) \tag{2.13}$$

is the average variance of all the $\binom{v}{2}$ estimated elementary contrasts (or pairwise comparisons) of the treatment parameters in d.

Remark. $\bar{v}$ provides a good measure of the efficiency of d. To enhance the efficiency of d, the value of $\bar{v}$ needs to be small.

Definition 2.6.3. Suppose d is a connected (v, b, r, k)-design with the associated scaled information matrix, $L^* = I - (rk)^{-1}NN'$. Denote by $e_1, e_2, ..., e_{v-1}$, the non-zero eigenvalues of L^*. Then $e_1, e_2, ..., e_{v-1}$ are called the canonical efficiency factors (c.e.f.s) of d.

Remark. Since L^* is not of full rank, there exists at least one zero eigenvalue. Again, because d is connected, $\text{rank}(L^*) = v - 1$: see the remark in definition 2.5.4. Hence, there are $v - 1$ non-zero eigenvalues, with only one of it being zero.

Let $|\Omega| = vr = bk$ denote the number of experimental units in d. Then $|\Omega| \geq v + b - 1$: see, for example, Jacroux (1978). Consequently, $v(r-1) \geq b-1$, or equivalently, $b(k-1) \geq v - 1$.

Suppose d is disconnected, then L^* would have fewer number of eigenvalues that are non-zeros, with more than one being zero, since in that case, $\text{rank}(L^*) < v - 1$.

The canonical efficiency factors are often utilized through some function of it to measure the efficiency of a design. This needs to be large enough to enhance a design efficiency.

Definition 2.6.4. Let d be a (v, b, r, k)-design. Suppose each elementary contrast, $\tau_i - \tau_{i'}$, where $i \neq i'$ is estimable in d. We remind that $\text{Var}(\hat{\tau}_i - \hat{\tau}_{i'}) = v_{ii'}\sigma^2/r$, where $v_{ii'} = L^{*-}_{ii} + L^{*-}_{i'i'} - 2L^{*-}_{ii'}$, $i \neq i'$: see definition 2.6.2. Define $e_{ii'}$ by

$$e_{ii'} = \frac{2\sigma^2/r}{v_{ii'}\sigma^2/r} = \frac{2}{v_{ii'}} \tag{2.14}$$

where $2\sigma^2/r$ is the variance associated with the estimator of the elementary contrast (or pairwise comparison), $\tau_i - \tau_{i'}$, $i \neq i'$ for a complete block design with the same parameters as d; which is believed to have the minimum value for this variance: see, for example, John and Mitchell (1977). Then $e_{ii'}$ is the efficiency factor for the elementary contrast (or pairwise comparison), $\tau_i - \tau_{i'}$, which compares treatment i with treatment i', $i \neq i'$; and has values between 0 and 1: see, for example, Bailey and Royle (1997). It is also called the pairwise efficiency factor.

Remark. The pairwise efficiency factor, $e_{ii'}$ $\forall$ pair (i, i') of treatments, $i \neq i'$ is not less than the minimum of the canonical efficiency factors: see, for example, John and Williams (1982).

By virtue of (2.14), (2.11) is identical to

$$\text{Var}(\hat{\tau}_i - \hat{\tau}_{i'}) = \frac{2\sigma^2}{r}\frac{1}{e_{ii'}} \tag{2.15}$$

Hence, from (2.15), it is obvious that the variance, $\mathrm{Var}(\hat{\tau}_i - \hat{\tau}_{i'})$ of the estimated elementary contrast, $\hat{\tau}_i - \hat{\tau}_{i'}$ of the treatment parameters τ_i and $\tau_{i'}$, $i \neq i'$ is proportional to the reciprocal of the efficiency factor for the contrast $\tau_i - \tau_{i'}$, for a fixed r.

Again, the harmonic mean (HM) of the efficiency factor, $e_{ii'}$ for all pairwise comparisons $\tau_i - \tau_{i'}$, $i \neq i'$ is proportional to the reciprocal of the average variance, $\bar{v}$ of all the estimated elementary contrasts; and this is seen as follows:

Since each contrast is estimable in d, thus making d connected, there exists $\binom{v}{2} = \frac{v(v-1)}{2}$ estimable distinct pairwise contrasts, which correspond to the number of values of $e_{ii'}$ obtainable. Thus

$$\begin{aligned}
\text{Harmonic mean of } e_{ii'} &= \frac{1}{2\frac{\sum_i \sum_{i'>i}(1/e_{ii'})}{v(v-1)}} \\
&= \frac{v(v-1)}{2\sum_i \sum_{i'>i}(1/e_{ii'})} \qquad (2.16)\\
&= \frac{v(v-1)}{\sum_i \sum_{i'>i} v_{ii'}}
\end{aligned}$$

obtained by using (2.14) in (2.16)

$$\begin{aligned}
&= \frac{1}{(2\sigma^2/r)\frac{\sum_i \sum_{i'>i} v_{ii'}}{v(v-1)}}(2\sigma^2/r) \\
&= \frac{1}{\bar{v}}(2\sigma^2/r) \\
&\propto \frac{1}{\bar{v}}
\end{aligned}$$

Hence, the desired result.

There is a quantity which measures the average variance, $\bar{v}$ over all estimated elementary contrasts of treatment parameters in a given design relative to the average of the same variance for a complete block design with the same parameters. This is the overall average efficiency factor of the design.

Definition 2.6.5. Let d be a (v, b, r, k)-design. Suppose d is connected. Let a quantity, E_o be defined by

$$E_o = \frac{2\sigma^2/r}{\bar{v}} = \frac{v(v-1)}{\sum_i \sum_{i'>i} v_{ii'}} \qquad (2.17)$$

Then E_o is the overall average efficiency factor of d.

Remark. The quantity, E_o associated with d gives a measure of how good d is relative to a complete block design of the same size/parameters. A design with a high efficiency factor would tend to have low variances of within-block estimators (contrasts): see, for example, Bailey and Royle (1997).

Bailey and Royle (1997) give four measures of the efficiency of a design, *viz*, the harmonic mean, A; the geometric mean, D; and the minimum, E of the canonical efficiency factors; as well as the minimum, E' of the efficiency factors, $e_{ii'}$ for elementary contrasts.

2.7 Design optimality

Designs within a given class are usually compared to determine the one(s) that outperform (perform best) and could, as such, be preferred over the other design(s) in that class, for purposes of experimentation.

Design optimality is, thus, concerned with finding (or choosing) the best possible realizable design(s) from amongst a given class of designs. This usually involves imposing certain well- and predefined conditions (or criteria), called optimality criteria, which follows somewhat from their corresponding efficiency measure. The design(s) which is/are found to be the best under the imposed criterion(a) is/are said to be the optimal design(s) in that class with respect to that criterion(a).

Optimality criteria are said to be functionals of the information matrix of the design: see, for example, Das (2002), and Cheng (1978).

Definition 2.7.1. Let $\mathcal{D}(v, b, r, k)$ denote a class of designs with the same values of the parameters v, b, r, k. Denote by ψ, an optimality criterion. Let d be any design in $\mathcal{D}$. Then $d^+ \in \mathcal{D}$ is said to be optimal in $\mathcal{D}$ with respect to ψ, or simply, ψ-optimal, if it satisfies the condition in ψ over other competing designs in $\mathcal{D}$.

We note that, this condition, usually involves an optimization (maximization or minimization) of some function of the canonical efficiency factors of a design. Thus, canonical efficiency factors play a vital role in determining optimal designs.

Some of the commonly used optimality criteria include the A-, D-, and E-criteria. Under these three criteria, a balanced incomplete block design (BIBD), when it exists, is known to be optimal; and when a BIBD does not exist, it is conjectured that the A-, D-, and E-optimal designs are to be found among regular-graph designs (RGDs), if such exist: see, for example, John and Mitchell (1977).

Remark. If more than one design in $\mathcal{D}$ satisfies the condition in ψ, then each of those designs satisfying this condition is said to be ψ-optimal in $\mathcal{D}$.

Again, suppose ψ and ϕ are two distinct optimality criteria. If d^* is ψ-optimal amongst all the competing designs in $\mathcal{D}$, it may not necessarily be ϕ-optimal in the same class: see, for example, John and Williams (1995, chapter 2).

Definition 2.7.2. Let d be a (v, b, r, k)-design with the treatments and blocks sets $\mathcal{V}$ and $\mathcal{B}$, respectively, where the cardinalities of $\mathcal{V}$ and $\mathcal{B}$, denoted $|\mathcal{V}|$ and $|\mathcal{B}|$ are, v and b, respectively. Suppose d^* is a (v^*, b^*, r^*, k^*)-design obtained from d by interchanging the roles of $\mathcal{V}$ and $\mathcal{B}$ such that the block labels for those blocks in which the ith treatment in d^* appears, $i \in \{1, 2, ..., v^*\}$ correspond to the treatment labels of those treatments in the jth block of d, $i = j \in \{1, 2, ..., b\}$; or equivalently, the treatment labels for those treatments contained in the jth block of d^*, $j \in \{1, 2, ..., b^*\}$ correspond to the block labels of those blocks that contain the ith treatment in d, $j = i \in \{1, 2, ..., v\}$. Then d^* is said to be the dual design of d, and *vice versa*.

As an illustration, suppose d is a $(4, 6, 3, 2)$-design. Denote by b_i, $i = 1, 2, ..., 6$ its blocks formed by the elements in the braces.

Suppose d: $b_1 = \{\tau_1, \tau_2\}, b_2 = \{\tau_1, \tau_3\}, b_3 = \{\tau_1, \tau_4\}, b_4 = \{\tau_2, \tau_3\}, b_5 = \{\tau_2, \tau_4\},$ and $b_6 = \{\tau_3, \tau_4\}$. Let b_i^*, $i = 1, 2, ..., 4$ denote the blocks of its dual design, d^*. Then d^* is the design:

$b_1^* = \{\tau_1, \tau_2, \tau_3\}, b_2^* = \{\tau_1, \tau_4, \tau_5\}, b_3^* = \{\tau_2, \tau_4, \tau_6\},$ and $b_4^* = \{\tau_3, \tau_5, \tau_6\}$.

Notice that d^* is a $(6, 4, 2, 3)$-design. Again, d is a BIBD with $\lambda = 1$, whereas d^* is not.

We note that $v^* = b, b^* = v, r^* = k,$ and $k^* = r$. Again, If N is the incidence matrix of d, then N' is the incidence matrix of d^*, and *vice versa*. Their information matrices are of order v and b, respectively. Suppose d is connected, and consequently, d^* is also connected. Then, their canonical efficiency factors (c.e.f.s), including multiplicities are identical apart from the excess $|b - v|$, each with value equal to 1. In particular, if $b \geq v$, then d^* has at least the same number of c.e.f.s as d, and the excess $b - v$ is attributed to d^*. On the contrary, if $b < v$, then d^* has fewer c.e.f.s than d such that each of the excess $v - b$ c.e.f.s is attributed to d: see, for example, Bailey (2004, chapter 4) and John and Williams (1995, chapter 2).

Remark. The dual design becomes most useful if v is very large compared to b, as the c.e.f.s can be easily obtained from its dual: see, for example, John and Williams (1995, chapter 2).

If d is a BIBD, then d^* is also a BIBD $\Longleftrightarrow$ d is symmetric.

Again, d^* is A-, D-, or E- optimal $\Longleftrightarrow$ d is A-, D-, or E- optimal, respectively: see, for example, John and Mitchell (1977). These authors observe that this is true for situations where d or d^* is (or not) a regular graph design.

We now describe some basic optimality criteria:

2.7.1 A-optimality criterion

The A-optimality criterion is concerned with maximizing the harmonic mean, A, of the canonical efficiency factors of a design. Equivalently, it minimizes the average variance of the estimators of elementary treatment contrasts: see, for example, Soicher (2013), Bailey and Cameron (2009), Morgan (2007), Bedford and Whitaker (2001), Bailey and Royle (1997), John and Williams (1982), John (1981), as well as Cheng (1978).

Definition 2.7.3. Given a class, $\mathcal{D}(v, b, r, k)$ of designs for a given set (v, b, r, k) of parameters, where the parameters have their usual meaning. On the basis of canonical efficiency factors, $d^+ \in \mathcal{D}$ is said to be A-optimal in $\mathcal{D}$ if the harmonic mean of its canonical efficiency factors is at least as large as that of $d' \in \mathcal{D}$, $\forall\ d' \neq d^+$, where d' is any other competing design in $\mathcal{D}$. That is, the harmonic mean of the canonical efficiency factors of d^+ cannot be less than that of any other competing design in the same class with it.

In particular, for connected designs, the harmonic mean, A, of its canonical efficiency factors is given by:

$$A = \frac{v-1}{\sum_{i=1}^{v-1} \frac{1}{e_i}}$$

Remark. The harmonic mean, A of the canonical efficiency factors of a connected design is identical to the overall average efficiency factor, E_o, based on pairwise treatment differences, given in (2.17): see, for example, Bailey (2004, chapter 4), as well as John and Williams (1995, chapter 2).

Definition 2.7.4. Given a (v, b, r, k) design, d, let d' denote a (v, b, r', k') design obtained from d whose blocks are formed by the complementary treatments in the corresponding block of d, such that $k' = v - k$, and $r' = b - r$. Then, d' is said to be the complementary design of d.

Considering the $(4, 6, 3, 2, 1)$-BIBD, d given in section 2.7, let b_i', $i = 1, 2, ..., 6$ denote the blocks of its complementary design, d'. Then d' is the design with the following block composition:

$b_1' = \{\tau_3, \tau_4\}, b_2' = \{\tau_2, \tau_4\}, b_3' = \{\tau_2, \tau_3\}, b_4' = \{\tau_1, \tau_4\}, b_5' = \{\tau_1, \tau_3\}$, and $b_6' = \{\tau_1, \tau_2\}$; which is also a BIBD. In particular, d' is also, by coincidence, a $(4, 6, 3, 2, 1)$-BIBD.

Remark. Suppose d^{cb} is a complete block design with the same number of treatments, v and blocks, b just like d and d'. Hence $v = k^{cb}$, and $b = r^{cb}$, where k^{cb} and r^{cb} denote the respective block size and replication number of each treatment in d^{cb}. Then $d' = d^{cb} \setminus d$ such that the incidence matrix of d' is $J - N$, where J is the incidence matrix of d^{cb}, which is a $v \times b$ matrix of 1s, and N the incidence matrix of d.

The complement of a BIBD is of necessity a BIBD. This property also hold for a RGD. Also, It is conjectured that if a design is A-optimal, then its complementary design is also A-optimal: see, for example, John and Williams (1982).

Again, suppose $e_i, i = 1, 2, ., ..., v - 1$ are the canonical efficiency factors (c.e.f.s) of d. Let e_i', $i = 1, 2, ..., v - 1$ denote the c.e.f.s of d'. Then,

$$e_i' = 1 - \alpha(1 - e_i)$$

where $\alpha = rk/r'k'$: see, for example, John and Williams (1982).

2.7.2 D-optimality criterion

The D-optimality criterion involves maximizing the geometric mean, D, of the canonical efficiency factors of a design. Equivalently, it minimizes the volume of a confidence ellipsoid containing the estimated treatment contrasts: see, for, example, Soicher (2013), Bailey and Cameron (2009), Morgan (2007), Bedford and Whitaker (2001), Bailey and Royle (1997), John and Williams (1982), and Cheng (1978).

Definition 2.7.5. Let $\mathcal{D}(v, b, r, k)$ denote a class of designs with the same values of v, b, r, k. Based on canonical efficiency factors, a design, d^+ within $\mathcal{D}$ is said to be D-optimal in $\mathcal{D}$ if the geometric mean of its canonical efficiency factors is at least as large as that of any other competing design within the same class. Hence d^+ has a geometric mean efficiency factor which is not less than that of any other competing design in $\mathcal{D}$.

The geometric mean, D, of the canonical efficiency factors of a connected design is given by:

$$D = (\prod_{i=1}^{v-1} e_i)^{\frac{1}{v-1}}$$

Remark. Conjecture 3 of John and Williams (1982) implies that if d^+ is a D-optimal design, then its complementary design is also D-optimal.

2.7.3 E-optimality criterion

The E-optimality criterion seeks to maximize the minimum, E, of the canonical efficiency factors of a design. Equivalently, it minimizes the maximum variance over all normalized treatment contrasts: see, for example, Soicher (2013), Bailey and Cameron (2009), Morgan (2007), Bedford and Whitaker (2001), Bailey and Royle (1997), Jacroux (1980), John and Williams (1982), and Cheng (1978).

Definition 2.7.6. Let d^+ denote a design within the class, $\mathcal{D}(v, b, r, k)$ of designs. Suppose canonical efficiency factors is used as a basis for finding the optimal design in $\mathcal{D}$. Then, d^+ is considered E-optimal in $\mathcal{D}$ if its minimum canonical efficiency factor is at least as large as that of any other competing design within this class. That is, the minimum canonical efficiency factor of d^+ cannot be less than that of any other competing design, d' in $\mathcal{D}$.

Remark. The E-criterion entails that, if $d^+ \in \mathcal{D}$ is E-optimal over $\mathcal{D}$, then

$$\max_{d' \in \mathcal{D}}(\min\{e_1, e_2, ..., e_{v-1}\}) \leq e,$$

where e is the smallest canonical efficiency factor of d^+. Note that if the strict inequality holds, then the E-optimal design in $\mathcal{D}$, d^+ is unique. However, if the equality holds, then there are more than one design in $\mathcal{D}$ that are E-optimal.

There exists another useful criterion for assessing the optimality of a design(s) known as the (M, S)-optimality criterion.

2.7.4 (M, S)-optimality criterion

The (M, S) optimality criterion involves a two-stage optimization procedure (maximization at the first stage and minimization at the second stage). The first stage involves selecting from a given class of designs those ones whose information matrices have maximal trace. At the second stage, the particular design(s) from amongst those with maximal trace of their information matrices whose squared information matrices have minimal trace is then selected, which is/are the (M, S)-optimal design(s): see, for example, John and Williams (1982), Jacroux (1978), as well as Eccleston and Hedayat (1974).

Definition 2.7.7. Let $\mathcal{D}(v, b, r, k)$ denote a class of designs with the same parameters v, b, r, k. Let $\mathcal{D}^*$ be the set of all d^* in $\mathcal{D}$ whose information matrices have maximal trace. Let $d^{*+} \in \mathcal{D}^*$ be such that its squared information matrix has minimal trace over $\mathcal{D}^*$. Then d^{*+} is said to be (M, S)-optimal in $\mathcal{D}$.

Remark. The first stage is the M-optimality, while the second is the S-optimality: see, for example, John and Williams (1982), as well as Eccleston and Hedayat (1974). Thus (M, S)-optimality combines both the M- and S-optimality components.

(M, S)-optimal designs are usually sought for among the class of binary designs, as such designs maximize the trace of their information matrix. Also, an (M, S)-optimal optimal design minimizes the sum of squares of all concurrences between pairs of distinct treatments among the competing designs. Moreover, a BIBD (or RGD), when it exists, is (M, S)-optimal; but when such designs do not exist, some of the concurrences between pairs of distinct treatments will differ by 2, or even more: see, for example, John and Williams (1982).

An (M, S)-optimal design is, usually, a connected design; though it may not always be: see, for example, Jacroux (1978). Suppose an (M, S)-optimal design is connected, the trace of its scaled information matrix is identical to the sum, $\sum_{i=1}^{v-1} e_i$ of its c.e.f.s.; while the sum, $\sum_{i=1}^{v-1} e_i^2$ gives the trace of its squared scaled information matrix.

The (M, S)-optimality, thus, involves selecting a subclass of designs which maximize $\sum_{i=1}^{v-1} e_i$ amongst all the designs in its class, at the first stage, and then choosing from the selected subclass that which minimize $\sum_{i=1}^{v-1} e_i^2$ amongst all designs in the selected subclass, at the second stage.

Given a class, $\mathcal{D}(v, b, k)$ of designs, the existence of an (M, S)-optimal design in $\mathcal{D}$ which is connected $\iff b(k-1) \geq v-1$. Again, the connectedness of an (M, S)-optimal design in $\mathcal{D}$ is guaranteed if $(k-1)(r_{\max} + r_{\min}) \geq v-1$, where $r_{\max}$ and $r_{\min}$ are the respective maximum and minimum number of replications of the treatments in the design: see, for example, Jacroux (1978). In particular, if $\mathcal{D}$ comprises equireplicated designs, the preceding condition reduces to $2r(k-1) \geq v-1$.

Furthermore, considering the class, $\mathcal{D}(v, b, k)$ of binary designs, let $d^+ \in \mathcal{D}$ denote an (M, S)-optimal design in $\mathcal{D}$. Suppose $\mathcal{D}'(v, b, v-k)$ denote the class of complementary designs that correspond to $\mathcal{D}$. Let $d^{+\prime} \in \mathcal{D}'$ denote the complementary design of d^+. Then, $d^{+\prime}$ is (M, S)-optimal in $\mathcal{D}'$. Similar property holds for the dual design of d^+, if $\mathcal{D}$ comprises proper equireplicated designs: see, for example, Jacroux (1978).

John and Williams (1982) conjecture that an (M, S)-optimal design which is A-optimal is also D-optimal; and also that an (M, S)-optimal design that is D-optimal is A-optimal.

2.8 Related work and direction of research

There are many discussions on semi-Latin squares (SLSs) in the literature, leaving behind, the generalized form of it, the semi-Latin rectangle. A pioneering work on semi-Latin rectangles (SLRs) can be found in Bailey and Monod (2001).

These authors constructed efficient $(h \times p)/k$ semi-Latin rectangles for $v = 2n$ treatments, where $h = n, p = 2n, k = 2$ and $2 \leq n \leq 10$, using combinatorics, *via* starters and the cyclic method for constructing balanced tournament designs. Each treatment of their designs appears once in each column and twice in each row. Their constructions, except for the case $n = 2$ yielded designs whose QBDs are RGDs, which have been conjectured to possess optimal properties over other designs in its class, when they exist: see, for example, John and Williams (1982), as well as John and Mitchell (1977).

Uto and Bailey (2020) gives some properties and conditions necessary for a balanced semi-Latin rectangle (BSLR) to exist, giving some algorithms for constructing such designs when each row-column intersection (block) contains $k = 2$ treatments. The algorithms generate designs of the class $h = v$, $p = v\delta$, where $\delta = (v - 1)/2$ for situations that v is odd and $h = v/2$, $p = v(v - 1)/2$ for the case that v is even. They also suggest methods of deriving some other classes of designs from the constructions.

In this work, we concentrate on BSLRs and RGSLRs, with a view to finding some suitable techniques to facilitate the construction of efficient SLRs of various classes and sizes, giving generalizations in some cases using combinatorics via concepts like graph distance, parallel classes, starters, balanced tournament designs (BTDs), difference sets/families, α-resolvable BIBDs, group-divisible designs that are regular-graph designs, as well as undirected terrace and the cyclic constructions. We also employ the concepts of permutation and complementation. Moreover, in some cases, we develop constructions by extending the work of Bailey and Monod (2001).

Chapter 3

Balanced Semi-Latin Rectangles with Block Size Two

3.1 Introduction

We shall, in this chapter, consider semi-Latin rectangles whose quotient block desigsns (QBDs) are balanced incomplete-block designs (BIBDs) and each row-column intersection of the design contains exactly two treatments. We name such designs balanced semi-Latin rectangles (BSLRs) with block size two and give constructions for this class of designs using some combinatorial approaches. We dwell on two concepts, viz, distances (graph distances)–the length of the shortest path between pairs of vertices of a regular n-gon (a regular polygon with n vertices) and parallel classes for a set of v symbols (treatments)–sets of blocks which partition the treatment set. Our constructions consider both odd and even number of treatments; and for each case an algorithmic procedure for constructing the design is given and backed up with some examples. New designs are obtained via some modifications of the procedures and also by transpositions. Moreover, some designs of larger sizes are also obtained. Some illustrative examples are also given for each case. Part of the results obtained have been published: see Uto and Bailey (2020).

3.2 Structure and associated properties of the design

The design under discussion has v treatments arranged in an $h \times p$ array consisting of h rows and p columns, where there are exactly two treatments in each row-column intersection.

Denoting the structure of the design by $(h \times p)/2$; each treatment appears n_c number of times in each column, where $n_c = 2h/v$. Similarly, each treatment appears n_r number of times per row, where $n_r = 2p/v$. Hence, overall, the replication number per treatment is $2hp/v$. We note that n_c and n_r may (or may not) be distinct. Consequently, h and p may (or may not) also be distinct. In particular, h is identical to p if and only if n_c is identical to n_r. Furthermore, for this design, its quotient block design (QBD) which contains hp blocks is an irreducible (or unreduced) BIBD, consisting of all the $\binom{v}{2}$ distinct 2-treatment subsets of v, where each subset is of multiplicity λ, and these constitute the cells entries (blocks) of the design.

3.3 Concepts used in the Construction of the designs

We concentrate on two concepts-distances and parallel classes by utilizing an n-gon, as given in section 3.1.

Definition 3.3.1. Let Λ denote a regular n-gon. Denote by $V = \{1, 2, ..., n\}$ its vertex set. Suppose the vertices are labelled in a cyclic order, with an edge between vertices i and $i + 1$ (reduced modulo n, if necessary). Let $\delta = \lfloor n/2 \rfloor$. For all $i, i' \in V$,$i' \neq i$, we define $d(i, i') \in [1, \delta]$ to be

$$d(i,i') = \begin{cases} n - |i' - i| & \text{if } |i' - i| > \delta, \\ |i' - i| & \text{if } |i' - i| \leq \delta. \end{cases}$$

Then $d(i, i')$ is said to be the distance between the vertices, i and i'.

Remark. $d(i, i')$ is the length of the shortest path connecting the pair, i and i' of vertices.

Definition 3.3.2. Let $\mathcal{V}$ denote the set of treatments of a $(v, 2, \lambda)$-BIBD, Γ, where 2 divides v. Denote by $\mathcal{B} = \{B_j\}_{j=1}^{b}$, its set of blocks; and $r = 2b/v$. Let $\mathcal{A}_l \subset \mathcal{B}$, where $\mathcal{A}_l = \{A_{lm}\}_{m=1}^{v/2}$, A_{lm} being the mth block in $\mathcal{A}_l$, $l = 1, 2, ..., r$. Suppose for all $A_{lm}, A_{lm'} \in \mathcal{A}_l$, $A_{lm} \cap A_{lm'} = \emptyset$, where $m' \neq m$, so that $\left|\bigcup_{m=1}^{v/2} A_{lm}\right| = \sum_{m=1}^{v/2} |A_{lm}| = v$. Then $\mathcal{A}_l$, $l = 1, 2, ..., r$ is said to be a parallel class (or resolution class) in Γ.

$\mathcal{A}_l$ is a set of blocks in Γ which partition $\mathcal{V}$.

Remarks.

- Each treatment appears in exactly one block within each parallel class.
- Each parallel class contains $v/2$ blocks.
- The partition of $\mathcal{B}$ into parallel classes, $\mathcal{A}_1, \mathcal{A}_2, ..., \mathcal{A}_r$ gives a resolution. Hence, Γ satisfying this property is said to be resolvable.

3.4 Basic construction when v is odd

When the set of treatments of the design to be constructed has cardinality equal to an odd number, we approach the construction as follows: By putting $n = v$ in Definition 3.3.1, we have $\delta = (v - 1)/2$. Furthermore, by identifying the treatments with the vertices of a regular v-gon and combining the vertices into pairs that are at unique distances apart, the entries of the cells in the first row of the design are obtained. For each pair, (i, i') of vertices, $d(i, i') \in [1, \delta]$; and for each unique distance, there are v distinct pairs of vertices associated with it, which we generate using a cyclic approach given below. These are then utilized in the construction.

Let the jth pair of vertices at a distance, u apart be denoted by $S_{uj} = \{j, j+u\}$, where $j = 1, 2, ..., v$, and $u = 1, 2, ..., \delta$; with a reduction (modulo v) for each component. Furthermore, let the v distinct pairs of vertices for each u be denoted by $\mathcal{S}_u = \{S_{uj}\}_{j=1}^{v}$.

From the foregoing, it is obvious that, $S_{u1} = \{1, 1+u\}$, $S_{u2} = \{2, 2+u\}$,. . ., $S_{uv} = \{v, v+u\}$. Similarly, $S_{1j} = \{j, j+1\}$, $S_{2j} = \{j, j+2\}$, . . ., $S_{\delta j} = \{j, j+\delta\}$, where there is a reduction (modulo v) for each component.

An algorithmic procedure for constructing the design is presented in section 3.4.1.

3.4.1 An algorithmic procedure for constructing the design using distances

1. Put $S_{uj} = \{j, j+u\}$, where $j = 1, 2, ..., v$ for each $u = 1, 2, ..., \delta$, and with a reduction (modulo v) for each component.

2. Make a Latin square, Δ_u, of order v using the sets $S_{u1}, S_{u2}, ..., S_{uv}$ as symbols, where $u = 1, 2, ..., \delta$.

3. Juxtapose the Latin squares $\Delta_1, \Delta_2, ..., \Delta_\delta$ made in 2., one beside another.

Comments. (1) Using the algorithmic procedure in section 3.4.1, a $(v \times v\delta)/2$ balanced semi-Latin rectangle is obtained. That is, the design has $h = v$ rows and $p = v\delta$ columns. The constructed design is of the form

Δ_1	Δ_2	$\cdots$	Δ_δ

The Latin square, Δ_u, $u = 1, 2, ..., \delta$ can take the form

$\Delta_u =$

S_{u1}	S_{u2}	$\cdots$	S_{uv}
S_{uv}	S_{u1}	$\cdots$	$S_{u,v-1}$
$\vdots$	$\vdots$	$\ddots$	$\vdots$
S_{u2}	S_{u3}	$\cdots$	S_{u1}

Note that, the order of juxtaposition of the Latin squares is immaterial, that is, it need not necessarily follow a natural order. Hence, can take any order.

(2) Furthermore, this algorithm produces designs where each pair of treatments concur once per row and $\lambda = h$ times, overall. Moreover, each treatment appears $v-1$ times per row and $v(v-1)$ times, overall. Thus, the QBD of the constructed design is a $(v, b, r, 2, v)$-BIBD, where $b = v^2(v-1)/2 = hp$ and $r = v(v-1) = 2p$.

The construction is illustrated with the following examples

Example 3.4.1. Let $v = 3$. Then $\delta = 1$, $h = 3$ and $p = 3$. The constructed design is shown in Figure 3.1.

1 2	2 3	3 1
3 1	1 2	2 3
2 3	3 1	1 2

Figure 3.1: A $(3 \times 3)/2$ balanced semi-Latin rectangle for 3 treatments

Remarks.

- Notice that the design in Figure 3.1 has the same number of rows and columns, which is a special case of the semi-Latin rectangle.
- Each treatment appears twice per row and also twice per column, hence, 6 times overall.
- Moreover, each pair of treatments concur 3 times in the design.
- The Quotient block design of this design is a $(3, 9, 6, 2, 3)$-BIBD.

Example 3.4.2. Let $v = 5$. Then $\delta = 2$, $h = 5$ and $p = 10$. The constructed design is presented in Figure 3.2.

Notice that this design takes the form

Δ_1	Δ_2

where

1 2	2 3	3 4	4 5	5 1	1 3	2 4	3 5	4 1	5 2
5 1	1 2	2 3	3 4	4 5	5 2	1 3	2 4	3 5	4 1
4 5	5 1	1 2	2 3	3 4	4 1	5 2	1 3	2 4	3 5
3 4	4 5	5 1	1 2	2 3	3 5	4 1	5 2	1 3	2 4
2 3	3 4	4 5	5 1	1 2	2 4	3 5	4 1	5 2	1 3

Figure 3.2: A $(5 \times 10)/2$ balanced semi-Latin rectangle for 5 treatments

$\Delta_1 =$

1 2	2 3	3 4	4 5	5 1
5 1	1 2	2 3	3 4	4 5
4 5	5 1	1 2	2 3	3 4
3 4	4 5	5 1	1 2	2 3
2 3	3 4	4 5	5 1	1 2

and

$\Delta_2 =$

1	3	2	4	3	5	4	1	5	2
5	2	1	3	2	4	3	5	4	1
4	1	5	2	1	3	2	4	3	5
3	5	4	1	5	2	1	3	2	4
2	4	3	5	4	1	5	2	1	3

Remarks.

- In this example, the number of columns in the design is double the number of rows.
- Each treatment appears 4 times per row and 2 times per column, hence, 20 times overall.
- Moreover, each pair of treatments concurs 5 times in the design.
- The Quotient block design of this design is a $(5, 50, 20, 2, 5)$-BIBD.

Example 3.4.3. Let $v = 7$. Then $\delta = 3$, $h = 7$ and $p = 21$. We obtain the design in Figure 3.3.

Remarks.

- The number of columns in the constructed design is 3 times the number of rows.
- Each treatment appears 6 times per row and 2 times per column, hence, 42 times overall.
- Moreover, each pair of treatments concurs 7 times in the design.
- The Quotient block design of this design is a $(7, 147, 42, 2, 7)$-BIBD.

Example 3.4.4. If $v = 9$, then $\delta = 4$, $h = 9$ and $p = 36$. We obtain the design in Figure 3.4

Remarks.

- In Example 3.4.4, the number of columns in the design is 4 times the number of rows.
- Each treatment appears 8 times per row and 2 times per column, hence, 72 times overall.

1 2	2 3	3 4	4 5	5 6	6 7	7 1	1 3	2 4	3 5	4 6	5 7	6 1	7 2	1 4	2 5	3 6	4 7	5 1	6 2	7 3
7 1	1 2	2 3	3 4	4 5	5 6	6 7	7 2	1 3	2 4	3 5	4 6	5 7	6 1	7 3	1 4	2 5	3 6	4 7	5 1	6 2
6 7	7 1	1 2	2 3	3 4	4 5	5 6	6 1	7 2	1 3	2 4	3 5	4 6	5 7	6 2	7 3	1 4	2 5	3 6	4 7	5 1
5 6	6 7	7 1	1 2	2 3	3 4	4 5	5 7	6 1	7 2	1 3	2 4	3 5	4 6	5 1	6 2	7 3	1 4	2 5	3 6	4 7
4 5	5 6	6 7	7 1	1 2	2 3	3 4	4 6	5 7	6 1	7 2	1 3	2 4	3 5	4 7	5 1	6 2	7 3	1 4	2 5	3 6
3 4	4 5	5 6	6 7	7 1	1 2	2 3	3 5	4 6	5 7	6 1	7 2	1 3	2 4	3 6	4 7	5 1	6 2	7 3	1 4	2 5
2 3	3 4	4 5	5 6	6 7	7 1	1 2	2 4	3 5	4 6	5 7	6 1	7 2	1 3	2 5	3 6	4 7	5 1	6 2	7 3	1 4

Figure 3.3: A $(7 \times 21)/2$ balanced semi-Latin rectangle for 7 treatments

1 2	2 3	3 4	4 5	5 6	6 7	7 8	8 9	9 1	1 3	2 4	3 5	4 6	5 7	6 8	7 9	8 1	9 2	1 4	2 5	3 6	4 7	5 8	6 9	7 1	8 2	9 3	1 5	2 6	3 7	4 8	5 9	6 1	7 2	8 3	9 4
9 1	1 2	2 3	3 4	4 5	5 6	6 7	7 8	8 9	9 2	1 3	2 4	3 5	4 6	5 7	6 8	7 9	8 1	9 3	1 4	2 5	3 6	4 7	5 8	6 9	7 1	8 2	9 4	1 5	2 6	3 7	4 8	5 9	6 1	7 2	8 3
8 9	9 1	1 2	2 3	3 4	4 5	5 6	6 7	7 8	8 1	9 2	1 3	2 4	3 5	4 6	5 7	6 8	7 9	8 2	9 3	1 4	2 5	3 6	4 7	5 8	6 9	7 1	8 3	9 4	1 5	2 6	3 7	4 8	5 9	6 1	7 2
7 8	8 9	9 1	1 2	2 3	3 4	4 5	5 6	6 7	7 9	8 1	9 2	1 3	2 4	3 5	4 6	5 7	6 8	7 1	8 2	9 3	1 4	2 5	3 6	4 7	5 8	6 9	7 2	8 3	9 4	1 5	2 6	3 7	4 8	5 9	6 1
6 7	7 8	8 9	9 1	1 2	2 3	3 4	4 5	5 6	6 8	7 9	8 1	9 2	1 3	2 4	3 5	4 6	5 7	6 9	7 1	8 2	9 3	1 4	2 5	3 6	4 7	5 8	6 1	7 2	8 3	9 4	1 5	2 6	3 7	4 8	5 9
5 6	6 7	7 8	8 9	9 1	1 2	2 3	3 4	4 5	5 7	6 8	7 9	8 1	9 2	1 3	2 4	3 5	4 6	5 8	6 9	7 1	8 2	9 3	1 4	2 5	3 6	4 7	5 9	6 1	7 2	8 3	9 4	1 5	2 6	3 7	4 8
4 5	5 6	6 7	7 8	8 9	9 1	1 2	2 3	3 4	4 6	5 7	6 8	7 9	8 1	9 2	1 3	2 4	3 5	4 7	5 8	6 9	7 1	8 2	9 3	1 4	2 5	3 6	4 8	5 9	6 1	7 2	8 3	9 4	1 5	2 6	3 7
3 4	4 5	5 6	6 7	7 8	8 9	9 1	1 2	2 3	3 5	4 6	5 7	6 8	7 9	8 1	9 2	1 3	2 4	3 6	4 7	5 8	6 9	7 1	8 2	9 3	1 4	2 5	3 7	4 8	5 9	6 1	7 2	8 3	9 4	1 5	2 6
2 3	3 4	4 5	5 6	6 7	7 8	8 9	9 1	1 2	2 4	3 5	4 6	5 7	6 8	7 9	8 1	9 2	1 3	2 5	3 6	4 7	5 8	6 9	7 1	8 2	9 3	1 4	2 6	3 7	4 8	5 9	6 1	7 2	8 3	9 4	1 5

Figure 3.4: A $(9 \times 36)/2$ balanced semi-Latin rectangle for 9 treatments

- Moreover, each pair of treatments concurs 9 times in the design.
- The Quotient block design of this design is a $(9, 324, 72, 2, 9)$-BIBD.

Comments. (1) The Latin squares, Δ_u, $u = 1, 2, ..., \delta$ used in the construction are not necessarily cyclic.

(2) For construction purposes, we introduced the double vertical lines. However, these should be ignored in the course of randomization.

(3) The algorithmic procedure given in Section 3.4.1, when implemented, produces designs with a unique property that each treatment appears $k = 2$ times in each column: see the preceding examples. This has to be so since $h = v$.

3.5 Basic construction when v is even

For those experimental situations where the cardinality of the set of treatments of the design under construction is an even number, we begin by making a resolvable design via combining pairs of treatments (blocks) into parallel classes with the aid of a regular n-gon, where $n = v - 1$. To achieve this, we proceed as follows: one treatment is designated a special symbol, ∞, say, and the remaining $v - 1$ treatments are then identified with the vertices of an n-gon (where $n = v-1$) which in union with the symbol, ∞ are combined into $v/2$ distinct pairs (blocks), which partition the set of treatments to form a parallel class; and overall, there are $r = v - 1$ parallel classes: see, for example, Street and Street (1987, Chapter 2) and Cameron (1994, Chapter 8). The treatment pairs in these parallel classes form the entries of the cells (blocks) of the initial row of the design under construction and are further utilized in the construction.

We denote by $\mathcal{V} = \{1, 2, ..., n\} \cup \{\infty\}$, the treatment set of the design, and $\mathcal{V}^- = \{1, 2, ..., n\}$, the set of vertices of a regular n-gon. Notice that $\mathcal{V} = \mathcal{V}^- \cup \{\infty\}$.

As in definition 3.3.2, we designate $\mathcal{A}_l$, $l = 1, 2, ..., v - 1$ the lth parallel class, and proceed to give an algebraic expression for generating the mth block in $\mathcal{A}_l$ denoted $\mathcal{A}_{lm}$, where $m = 1, 2, ..., v/2$.

For $l = 1, 2, ..., v - 1$, we obtain A_{lm} to be

$$A_{lm} = \begin{cases} \{l, \infty\} & \text{if } m = 1, \\ \{l + m - 1, l - m + 1\} & \text{if } m \in (1, v/2] \end{cases}$$

where each component is reduced modulo $v-1$. Notice that, $A_{11} = \{1, \infty\}$, $A_{21} = \{2, \infty\}$, ..., $A_{v-1,1} = \{v - 1, \infty\}$; and in general, for each l, $A_{l1} = \{l, \infty\}$.

An algorithmic procedure for the construction of the design is presented in section 3.5.1.

3.5.1 An algorithmic procedure for constructing the design using parallel classes

1. Generate $\{A_{lm}\}_{m=1}^{v/2}$, the $v/2$ blocks that make up $\mathcal{A}_l$, where $l = 1, 2, ..., v-1$,

$$A_{lm} = \begin{cases} \{l, \infty\} & \text{if } m = 1, \\ \{l+m-1, l-m+1\} & \text{if } m \in (1, v/2] \end{cases}$$

   and each component is reduced modulo $v-1$.

2. Make a Latin square, Ξ_l of order $v/2$ using the blocks A_{l1}, A_{l2}, . . ., $A_{l,v/2}$, as symbols, where $l = 1, 2, ..., v-1$.

3. Juxtapose the Latin squares $\Xi_1, \Xi_2, ..., \Xi_{v-1}$ made in 2, one beside another.

Comments. (1) The design resulting from implementing the algorithm is an $(h \times p)/2$ BSLR, where $h = v/2$, and $p = v(v-1)/2$. It takes the form

Ξ_1	Ξ_2	$\cdots$	Ξ_{v-1}

(2) The Latin squares, Ξ_l, where $l = 1, 2, ..., v-1$ can take the cyclic form; but this is not a necessity. If it is cyclic, it can be of the form

$$\Xi_l = \begin{array}{|c|c|c|c|} \hline A_{l1} & A_{l2} & \cdots & A_{lx} \\ \hline A_{lx} & A_{l1} & \cdots & A_{l,x-1} \\ \hline \vdots & \vdots & \ddots & \vdots \\ \hline A_{l2} & A_{l3} & \cdots & A_{l1} \\ \hline \end{array}$$

where $x = v/2$. Furthermore, as noted in section 3.4.1, the Latin squares can be juxtaposed in any order, not necessarily in a natural sequence.

(3) Designs produced via the algorithm has $\lambda = 1$ per row and $\lambda = h$, overall. Each treatment is replicated $v-1$ times per row and $v(v-1)/2$ times overall. Hence, its QBD is a $(v, b, r, 2, h)$-BIBD, where $b = (v/2)^2(v-1) = hp$ and $r = v(v-1)/2 = p$.

We illustrate the construction with the following examples.

Example 3.5.1. Let $v = 4$. Then $h = 2$ and $p = 6$. The design takes the form

Ξ_1	Ξ_2	Ξ_3

where

$\Xi_1 =$

1 ∞	2 3
2 3	1 ∞

$\Xi_2 =$

2 ∞	3 1
3 1	2 ∞

and

$\Xi_3 =$

3 ∞	1 2
1 2	3 ∞

Hence, the design is a $(2 \times 6)/2$ BSLR: see Figure 3.5.

1 ∞	2 3	2 ∞	3 1	3 ∞	1 2
2 3	1 ∞	3 1	2 ∞	1 2	3 ∞

Figure 3.5: A $(2 \times 6)/2$ balanced semi-Latin rectangle (BSLR) for 4 treatments

Remarks.

- Notice that each treatment appears 3 times in each row and exactly once in each column. Thus, it appears 6 times, overall.

- Each pair of treatments occurs exactly once per row and 2 times, overall, in the design. Its QBD is a $(4, 12, 6, 2, 2)$-BIBD

Example 3.5.2. Let $v = 6$. Then $h = 3$ and $p = 15$. The design takes the form

Ξ_1	Ξ_2	Ξ_3	Ξ_4	Ξ_5

where

$\Xi_1 =$

1 ∞	2 5	3 4
3 4	1 ∞	2 5
2 5	3 4	1 ∞

$\Xi_2 =$

2 ∞	3 1	4 5
4 5	2 ∞	3 1
3 1	4 5	2 ∞

$\Xi_3 =$

3 ∞	4 2	5 1
5 1	3 ∞	4 2
4 2	5 1	3 ∞

$\Xi_4 =$

4 ∞	5 3	1 2
1 2	4 ∞	5 3
5 3	1 2	4 ∞

$\Xi_5 =$

5 ∞	1 4	2 3
2 3	5 ∞	1 4
1 4	2 3	5 ∞

Hence, we have, as the design, a $(3 \times 15)/2$ BSLR which is presented in Figure 3.6.

1 ∞	2 5	3 4	2 ∞	3 1	4 5	3 ∞	4 2	5 1	4 ∞	5 3	1 2	5 ∞	1 4	2 3
3 4	1 ∞	2 5	4 5	2 ∞	3 1	5 1	3 ∞	4 2	1 2	4 ∞	5 3	2 3	5 ∞	1 4
2 5	3 4	1 ∞	3 1	4 5	2 ∞	4 2	5 1	3 ∞	5 3	1 2	4 ∞	1 4	2 3	5 ∞

Figure 3.6: A $(3 \times 15)/2$ balanced semi-Latin rectangle for 6 treatments

Remarks. Notice that each column of the design contains each treatment exactly once, while each row has each treatment appearing 5 times. Hence, overall, each treatment appears 15 times. Furthermore, each pair of treatments concurs exactly once in each row and 3 times, overall, in the design. The design has as its QBD, a $(6, 45, 15, 2, 3)$-BIBD.

Example 3.5.3. Let $v = 8$. Then $h = 4$ and $p = 28$. The design can be represented in skeletal form as

Ξ_1	Ξ_2	Ξ_3	Ξ_4	Ξ_5	Ξ_6	Ξ_7

We give the full design in Figure 3.7.

Remarks. The number of columns in the design is 7 times the number of rows. For each row, each treatment makes an appearance 7 times; and for each column, it appears exactly once. Hence, overall, each treatment is replicated 28 times. Moreover, the concurrence number per row of the design is unity and is 4, overall. Hence, its QBD is a $(8, 112, 28, 2, 4)$-BIBD.

3.6 Some derivable designs from the basic constructions

Definition 3.6.1. Let Γ_1 and Γ_2 denote two $(h \times p)/k$ SLRs for v treatments. We consider Γ_1 and Γ_2 to be the same if their corresponding cell entries are the same, otherwise they

1 ∞	2 7	3 6	4 5	2 ∞	3 1	4 7	5 6	3 ∞	4 2	5 1	6 7	4 ∞	5 3	6 2	7 1	5 ∞	6 4	7 3	1 2	6 ∞	7 5	1 4	2 3	7 ∞	1 6	2 5	3 4
4 5	1 ∞	2 7	3 6	5 6	2 ∞	3 1	4 7	6 7	3 ∞	4 2	5 1	7 1	4 ∞	5 3	6 2	1 2	5 ∞	6 4	7 3	2 3	6 ∞	7 5	1 4	3 4	7 ∞	1 6	2 5
3 6	4 5	1 ∞	2 7	4 7	5 6	2 ∞	3 1	5 1	6 7	3 ∞	4 2	6 2	7 1	4 ∞	5 3	7 3	1 2	5 ∞	6 4	1 4	2 3	6 ∞	7 5	2 5	3 4	7 ∞	1 6
2 7	3 6	4 5	1 ∞	3 1	4 7	5 6	2 ∞	4 2	5 1	6 7	3 ∞	5 3	6 2	7 1	4 ∞	6 4	7 3	1 2	5 ∞	7 5	1 4	2 3	6 ∞	1 6	2 5	3 4	7 ∞

Figure 3.7: A $(4 \times 28)/2$ balanced semi-Latin rectangle for 8 treatments

are different.

However, if their corresponding cell entries are not the same, it may be possible, in some cases that, performing at least one of the following: a permutation of the rows, a permutation of the columns, and a permutation of the treatments of Γ_1 (or Γ_2) leads to Γ_2 (or Γ_1). In this situation, Γ_1 and Γ_2 are said to be isomorphic, otherwise they are non-isomorphic. More formally, we consider Γ_1 and Γ_2 to be isomorphic if there is at least one of a permutation of the rows, a permutation of the columns, and a permutation of the treatments that takes either of the two designs to the other, otherwise they are non-isomorphic designs. Note that if Γ_1 is isomorphic to Γ_2, then Γ_2 is also isomorphic to Γ_1.

Given the construction procedures for an $(h \times p)/2$ balanced semi-Latin rectangle (BSLR) for v treatments in sections 3.4.1 and 3.5.1, we deduce from them new designs having the same number of treatments by modifying step 3 of these procedures. One of such modifications involves juxtaposing the Latin squares underneath instead of beside. This modification, for instance, produce designs isomorphic to those obtained by a transposition of the designs obtained via the algorithm. We designate any design for a given v obtained by a direct implementation of the algorithmic procedure the parent/basic design and the design obtained by this modification of step 3 of the procedure with juxtaposition(s) done underneath an alternative basic design. Moreover, a transposition of the alternative basic design produces another design with the same number of rows and columns as the basic design which is isomorphic to the basic design.

More designs can also be obtained via appropriate juxtapositions if the set of Latin squares used in the construction has cardinality equal to a nonprime.

Moreover, some designs of larger sizes can also be deduced from the basic (or alternative basic) design by making multiple copies of it and subsequently, juxtaposing them appropriately.

3.6.1 Designs with $h = v(v-1)/2$ rows and $p = v$ (or $v/2$) columns

From our previous constructions which produce basic designs for v treatments in h rows and p columns, the modification described in section 3.6, which involves juxtaposing the Latin squares, this time, one underneath another produces another balanced semi-Latin rectangle for v treatments, which is the alternative basic design. The number of rows and columns of the alternative basic design are in reversed order with that of the basic design.

Suppose v is odd, where the basic design has v treatments, $h = v$ rows and $p = v\delta = v(v-1)/2$ columns, a downward juxtaposition of the Latin squares $\Delta_1, \Delta_2, ..., \Delta_\delta$ produces an alternative basic design with v treatments, where $h = v(v-1)/2$ rows and $p = v$ columns. This design takes the form

$$\begin{array}{|c|}\hline \Delta_1 \\ \hline\hline \Delta_2 \\ \hline\hline \vdots \\ \hline\hline \Delta_\delta \\ \hline \end{array}$$

Similar results follow if v is even, where in this case, the basic design contains v treatments in $h = v/2$ rows and $p = v(v-1)/2$ columns. Hence, juxtaposing the Latin squares, $\Xi_1, \Xi_2, ..., \Xi_{v-1}$ underneath, we obtain an alternative basic design which has v treatments in $v(v-1)/2$ rows and $v/2$ columns. The design takes the form

$$\begin{array}{|c|}\hline \Xi_1 \\ \hline\hline \Xi_2 \\ \hline\hline \vdots \\ \hline\hline \Xi_{v-1} \\ \hline \end{array}$$

Remark. As noted in sections 3.4.1 and 3.5.1, the juxtaposition of the Latin squares must not follow a fixed order.

Notice that the number of rows of the alternative basic design is identical to the number of columns of the basic design and vice versa.

A design of the same size as the one considered in this section can be obtained by simply, transposing the basic design, that is, interchanging the roles of rows and columns in the basic design. The design resulting from this transposition is isomorphic to the alternative basic design. In particular, if each constituent Latin square is symmetric and there is a definite (same) order of juxtaposition of the Latin squares in both the basic and alternative basic designs, then the design obtained via transposition of the basic design is the same as the alternative basic design, since in this case their corresponding blocks (cell entries) are the same.

Similarly, transposing the alternative basic design produces a design which is also of the same size as the basic design and isomorphic to it. These two designs are the same if the aforementioned condition is satisfied.

We now give some examples.

Example 3.6.1. Let $v = 5$ and $\delta = 2$ as in Example 3.4.2. Then we obtain a $(10 \times 5)/2$ BSLR by juxtaposing the Latin square Δ_2 underneath Δ_1. The resulting design is an alternative basic design and is presented in Figure 3.8.

By simply transposing the $(5 \times 10)/2$ BSLR of Figure 3.2, the basic design, another version of a $(10 \times 5)/2$ BSLR results which is isomorphic to the alternative basic design in Figure 3.8 and is presented in Figure 3.9.

Remark. The isomorphism of the designs in Figures 3.8 and 3.9 can be seen by imposing the permutation α, β and I on the columns, rows and treatments, respectively of either design, where

$$\alpha = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 1 & 5 & 4 & 3 & 2 \end{pmatrix}$$

$$\beta = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\ 1 & 5 & 4 & 3 & 2 & 6 & 10 & 9 & 8 & 7 \end{pmatrix}$$

and

$$I = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 1 & 2 & 3 & 4 & 5 \end{pmatrix},$$

the identity permutation.

Now, by transposing the alternative basic design, we obtain another version of the $(5 \times 10)/2$ BSLR: see Figure 3.10. Notice that the design in Figure 3.10 is isomorphic to

1 2	2 3	3 4	4 5	5 1
5 1	1 2	2 3	3 4	4 5
4 5	5 1	1 2	2 3	3 4
3 4	4 5	5 1	1 2	2 3
2 3	3 4	4 5	5 1	1 2
1 3	2 4	3 5	4 1	5 2
5 2	1 3	2 4	3 5	4 1
4 1	5 2	1 3	2 4	3 5
3 5	4 1	5 2	1 3	2 4
2 4	3 5	4 1	5 2	1 3

Figure 3.8: A $(10 \times 5)/2$ balanced semi-Latin rectangle for 5 treatments obtained by juxtaposition underneath

the basic design, and vice versa. An application of the permutation α, β and I to the rows, columns and treatments, respectively of either of these designs reveals this.

Example 3.6.2. Let $v = 7$ and $\delta = 3$ as in Example 3.4.3. We obtain a $(21 \times 7)/2$ BSLR by juxtaposing the Latin squares Δ_1, Δ_2 and Δ_3, one underneath another. The design resulting from this juxtaposition is an alternative basic design and is presented in Figure 3.11.

By transposing the $(7 \times 21)/2$ BSLR of Figure 3.3, the basic design, we obtain another $(21 \times 7)/2$ BSLR, which is isomorphic to the alternative basic design in Figure 3.11). The resulting design is presented in Figure 3.12.

Moreover, we transpose the alternative basic design to obtain another $(7 \times 21)/2$ BSLR: see Figure 3.13, which is isomorphic to the basic design.

Example 3.6.3. Let $v = 9$ and $\delta = 4$ as in Example 3.4.4. Then we obtain a $(36 \times 9)/2$ BSLR by juxtaposing the Latin squares Δ_1, Δ_2, Δ_3 and Δ_4, one underneath another. This design is an alternative basic design: see Figure 3.14.

1	2	5	1	4	5	3	4	2	3
2	3	1	2	5	1	4	5	3	4
3	4	2	3	1	2	5	1	4	5
4	5	3	4	2	3	1	2	5	1
5	1	4	5	3	4	2	3	1	2
1	3	5	2	4	1	3	5	2	4
2	4	1	3	5	2	4	1	3	5
3	5	2	4	1	3	5	2	4	1
4	1	3	5	2	4	1	3	5	2
5	2	4	1	3	5	2	4	1	3

Figure 3.9: A $(10 \times 5)/2$ balanced semi-Latin rectangle for 5 treatments obtained by transposition of the basic design in Figure 3.2

By transposing the $(9 \times 36)/2$ BSLR of Figure 3.4 which serves as the basic design, we obtain another $(36 \times 9)/2$ BSLR isomorphic to the alternative basic design: see Figure 3.15. Furthermore, by transposing the alternative basic design, we obtain another $(9 \times 36)/2$ BSLR: see Figure 3.16, which is isomorphic to the basic design in Figure 3.4.

Example 3.6.4. Let $v = 4$ as in Example 3.5.1. By juxtaposing Ξ_1, Ξ_2 and Ξ_3 underneath, the resulting design is a $(6 \times 2)/2$ BSLR, which is shown in Figure 3.17 .

Notice that, in this example, transposing the basic design in Figure 3.5 leads to a design which is identical to the alternative basic design given in Figure 3.17. Similarly, transposing the alternative basic design results in the basic design. However, these are mere coincidences, and do not happen in general.

We note, as given earlier in the remark in section 3.6.1 that, in particular, the aforementioned property of these designs hold if the juxtaposed Latin squares that make the designs are each symmetric and also the order of their juxtaposition(s) in both the basic and alternative basic designs are the same. For instance, in this example, each of the Latin squares Ξ_1, Ξ_2, and Ξ_3 that make the basic and alternative basic designs is symmetric,

1 2	5 1	4 5	3 4	2 3	1 3	5 2	4 1	3 5	2 4
2 3	1 2	5 1	4 5	3 4	2 4	1 3	5 2	4 1	3 5
3 4	2 3	1 2	5 1	4 5	3 5	2 4	1 3	5 2	4 1
4 5	3 4	2 3	1 2	5 1	4 1	3 5	2 4	1 3	5 2
5 1	4 5	3 4	2 3	1 2	5 2	4 1	3 5	2 4	1 3

Figure 3.10: A $(5 \times 10)/2$ balanced semi-Latin rectangle for 5 treatments obtained via transposition of Figure 3.8

.

and also in both designs the order of juxtapositions follow the sequence Ξ_1, Ξ_2, Ξ_3.

Example 3.6.5. Let $v = 6$ as in Example 3.5.2. Juxtaposing Ξ_1, Ξ_2, Ξ_3, Ξ_4 and Ξ_5 underneath results in a $(15 \times 3)/2$ BSLR, which is the alternative basic design: see Figure 3.18.

Now, by transposing the basic design in Figure 3.6, we obtain another version of the $(15 \times 3)/2$ BSLR presented in Figure 3.19. Notice that, the designs in Figures 3.19 and 3.18 are isomorphic designs.

Moreover, by transposing the design in Figure 3.18, we obtain another $(3 \times 15)/2$ BSLR which is isomorphic to the basic design in Figure 3.6. This is shown in Figure 3.20.

Example 3.6.6. Let $v = 8$ as in Example 3.5.3. We juxtapose Ξ_1, Ξ_2, Ξ_3, Ξ_4, Ξ_5, Ξ_6, and Ξ_7 underneath to obtain a $(28 \times 4)/2$ BSLR, the alternative basic design, and this is shown in Figure 3.21.

A transposition of the basic design in Figure 3.7 gives a design which is isomorphic to the alternative basic design in Figure 3.21. The transposed design is shown in Figure 3.22.

Moreover, a transposition of the alternative basic design also gives another design which is isomorphic to the basic design: see Figure 3.23

1 2	2 3	3 4	4 5	5 6	6 7	7 1
7 1	1 2	2 3	3 4	4 5	5 6	6 7
6 7	7 1	1 2	2 3	3 4	4 5	5 6
5 6	6 7	7 1	1 2	2 3	3 4	4 5
4 5	5 6	6 7	7 1	1 2	2 3	3 4
3 4	4 5	5 6	6 7	7 1	1 2	2 3
2 3	3 4	4 5	5 6	6 7	7 1	1 2
1 3	2 4	3 5	4 6	5 7	6 1	7 2
7 2	1 3	2 4	3 5	4 6	5 7	6 1
6 1	7 2	1 3	2 4	3 5	4 6	5 7
5 7	6 1	7 2	1 3	2 4	3 5	4 6
4 6	5 7	6 1	7 2	1 3	2 4	3 5
3 5	4 6	5 7	6 1	7 2	1 3	2 4
2 4	3 5	4 6	5 7	6 1	7 2	1 3
1 4	2 5	3 6	4 7	5 1	6 2	7 3
7 3	1 4	2 5	3 6	4 7	5 1	6 2
6 2	7 3	1 4	2 5	3 6	4 7	5 1
5 1	6 2	7 3	1 4	2 5	3 6	4 7
4 7	5 1	6 2	7 3	1 4	2 5	3 6
3 6	4 7	5 1	6 2	7 3	1 4	2 5
2 5	3 6	4 7	5 1	6 2	7 3	1 4

Figure 3.11: A $(21 \times 7)/2$ balanced semi-Latin rectangle for 7 treatments obtained by juxtaposition underneath

3.6.2 Designs of the classes $(mv \times nv)/2$ and $(av/2 \times bv/2)/2$, where $mn = \delta$ and $ab = v - 1$

Given our construction (or modified construction) which produces a BSLR for v treatments that is a basic (or an alternative basic) design, we obtain more designs if either δ or $v-1$ is a nonprime corresponding to the case where v is odd or even via some alternative form of juxtapositions, which is bidirectional. This involves juxtaposing the Latin squares in both directions (sideways and underneath) instead of exclusively to one direction as before.

Suppose v is odd. We remind that the basic design has the parameters: $h = v$ rows and $p = v\delta$ columns; while the alternative basic design has $h = v\delta$ rows and $p = v$ columns. Let δ be a nonprime. Furthermore, let $m, n \in \mathbb{Z}$, where $1 < m, n < \delta$ such that $\delta = mn$. Then, for any such m, n, we obtain an $(h^* \times p^*)/2$ BSLR for the same number of treatments as the basic (or alternative basic) design by another modification of step 3 of the procedure in section 3.4.1 via an appropriate juxtaposition of the Latin squares, $\Delta_1, \Delta_2, ..., \Delta_\delta$ some

1	2	7	1	6	7	5	6	4	5	3	4	2	3
2	3	1	2	7	1	6	7	5	6	4	5	3	4
3	4	2	3	1	2	7	1	6	7	5	6	4	5
4	5	3	4	2	3	1	2	7	1	6	7	5	6
5	6	4	5	3	4	2	3	1	2	7	1	6	7
6	7	5	6	4	5	3	4	2	3	1	2	7	1
7	1	6	7	5	6	4	5	3	4	2	3	1	2
1	3	7	2	6	1	5	7	4	6	3	5	2	4
2	4	1	3	7	2	6	1	5	7	4	6	3	5
3	5	2	4	1	3	7	2	6	1	5	7	4	6
4	6	3	5	2	4	1	3	7	2	6	1	5	7
5	7	4	6	3	5	2	4	1	3	7	2	6	1
6	1	5	7	4	6	3	5	2	4	1	3	7	2
7	2	6	1	5	7	4	6	3	5	2	4	1	3
1	4	7	3	6	2	5	1	4	7	3	6	2	5
2	5	1	4	7	3	6	2	5	1	4	7	3	6
3	6	2	5	1	4	7	3	6	2	5	1	4	7
4	7	3	6	2	5	1	4	7	3	6	2	5	1
5	1	4	7	3	6	2	5	1	4	7	3	6	2
6	2	5	1	4	7	3	6	2	5	1	4	7	3
7	3	6	2	5	1	4	7	3	6	2	5	1	4

Figure 3.12: A $(21 \times 7)/2$ balanced semi-Latin rectangle for 7 treatments obtained by transposition of Figure 3.3

.

of them beside and the rest underneath, where $h^* = mv$ and $p^* = nv$.

Notice that, if $m = n = q$, say, then $h^* = p^* = qv$. Hence, the produced design has identical number of rows as columns, which is a special case of the semi-Latin rectangle. Clearly, m is identical to n if and only if δ is a perfect square. Hence, if δ is not a perfect square, m and n are found to be distinct with at least 2 values for each.

Similarly, if v is even, where in this case, the parameters of the basic design are $h = v/2$ rows and $p = v(v-1)/2$ columns; and for the alternative basic design, these are $h = v(v-1)/2$ rows and $p = v/2$ columns. Let $v - 1$ be a nonprime. Furthermore, let $a, b \in \mathbb{Z}$, where $1 < a, b < v - 1$ such that $v - 1 = ab$. Then, for any such a, b, we obtain an $(h^+ \times p^+)/2$ BSLR for the same number of treatments by also modifying step 3 of the procedure in section 3.5.1 via juxtaposing the Latin squares, $\Xi_1, \Xi_2, ..., \Xi_{v-1}$ both beside and underneath as in the case where v is odd. Note that $h^+ = av/2$ and $p^+ = bv/2$.

In particular, if $a = b = c$, say, then $h^+ = p^+ = cv/2$. In this circumstance, as before,

1 2	7 1	6 7	5 6	4 5	3 4	2 3	1 3	7 2	6 1	5 7	4 6	3 5	2 4	1 4	7 3	6 2	5 1	4 7	3 6	2 5
2 3	1 2	7 1	6 7	5 6	4 5	3 4	2 4	1 3	7 2	6 1	5 7	4 6	3 5	2 5	1 4	7 3	6 2	5 1	4 7	3 6
3 4	2 3	1 2	7 1	6 7	5 6	4 5	3 5	2 4	1 3	7 2	6 1	5 7	4 6	3 6	2 5	1 4	7 3	6 2	5 1	4 7
4 5	3 4	2 3	1 2	7 1	6 7	5 6	4 6	3 5	2 4	1 3	7 2	6 1	5 7	4 7	3 6	2 5	1 4	7 3	6 2	5 1
5 6	4 5	3 4	2 3	1 2	7 1	6 7	5 7	4 6	3 5	2 4	1 3	7 2	6 1	5 1	4 7	3 6	2 5	1 4	7 3	6 2
6 7	5 6	4 5	3 4	2 3	1 2	7 1	6 1	5 7	4 6	3 5	2 4	1 3	7 2	6 2	5 1	4 7	3 6	2 5	1 4	7 3
7 1	6 7	5 6	4 5	3 4	2 3	1 2	7 2	6 1	5 7	4 6	3 5	2 4	1 3	7 3	6 2	5 1	4 7	3 6	2 5	1 4

Figure 3.13: A $(7 \times 21)/2$ balanced semi-Latin rectangle for 7 treatments obtained by transposition of Figure 3.11

.

the produced design has identical number of rows as columns. It is obvious that a is identical to b if and only if $v-1$ is a perfect square. Thus, in situations where $v-1$ is not a perfect square, a and b are found to be distinct with at least 2 values for each

Example 3.6.7. We refer to Example 3.4.4, where $v = 9$ and $\delta = 4$. In this example, $m = n = 2$. Hence, $q = 2$ and $h^* = p^* = 18$. The resulting $(18 \times 18)/2$ BSLR is shown in Figure 3.24.

Example 3.6.8. Let $v - 1 = 9$. Notice that, in this example, $a = b = 3$. Hence, $c = 3$. Consequently, $h^+ = p^+ = 15$, and the resulting design is a $(15 \times 15)/2$ BSLR: see Figure 3.25.

Comments. (1) Notice that, in Example 3.6.7, with $\delta = 4$ being the square of a prime, there is only one admissible value for both m and n, which is 2. Notice also that, in Example 3.6.8, 3 is the only admissible value for a and b since 9 is a perfect square and being the square of 3.

(2) Now, suppose $\delta = 12$, which happens if and only if $v = 25$, then there are distinct values for m and n. In particular, the admissible values are $(m, n) = (2, 6), (6, 2), (3, 4)$ and $(4, 3)$. Hence, by an appropriate modification of step 3 of the algorithmic procedure in section section 3.4.1, we obtain a $(50 \times 150)/2$ BSLR and $(75 \times 100)/2$ BSLR for the pairs $(m, n) = (2, 6)$ and $(3, 4)$, respectively; and a $(150 \times 50)/2$ BSLR and $(100 \times 75)/2$ BSLR for the pairs in reversed order.

1 2	2 3	3 4	4 5	5 6	6 7	7 8	8 9	9 1
9 1	1 2	2 3	3 4	4 5	5 6	6 7	7 8	8 9
8 9	9 1	1 2	2 3	3 4	4 5	5 6	6 7	7 8
7 8	8 9	9 1	1 2	2 3	3 4	4 5	5 6	6 7
6 7	7 8	8 9	9 1	1 2	2 3	3 4	4 5	5 6
5 6	6 7	7 8	8 9	9 1	1 2	2 3	3 4	4 5
4 5	5 6	6 7	7 8	8 9	9 1	1 2	2 3	3 4
3 4	4 5	5 6	6 7	7 8	8 9	9 1	1 2	2 3
2 3	3 4	4 5	5 6	6 7	7 8	8 9	9 1	1 2
1 3	2 4	3 5	4 6	5 7	6 8	7 9	8 1	9 2
9 2	1 3	2 4	3 5	4 6	5 7	6 8	7 9	8 1
8 1	9 2	1 3	2 4	3 5	4 6	5 7	6 8	7 9
7 9	8 1	9 2	1 3	2 4	3 5	4 6	5 7	6 8
6 8	7 9	8 1	9 2	1 3	2 4	3 5	4 6	5 7
5 7	6 8	7 9	8 1	9 2	1 3	2 4	3 5	4 6
4 6	5 7	6 8	7 9	8 1	9 2	1 3	2 4	3 5
3 5	4 6	5 7	6 8	7 9	8 1	9 2	1 3	2 4
2 4	3 5	4 6	5 7	6 8	7 9	8 1	9 2	1 3
1 4	2 5	3 6	4 7	5 8	6 9	7 1	8 2	9 3
9 3	1 4	2 5	3 6	4 7	5 8	6 9	7 1	8 2
8 2	9 3	1 4	2 5	3 6	4 7	5 8	6 9	7 1
7 1	8 2	9 3	1 4	2 5	3 6	4 7	5 8	6 9
6 9	7 1	8 2	9 3	1 4	2 5	3 6	4 7	5 8
5 8	6 9	7 1	8 2	9 3	1 4	2 5	3 6	4 7
4 7	5 8	6 9	7 1	8 2	9 3	1 4	2 5	3 6
3 6	4 7	5 8	6 9	7 1	8 2	9 3	1 4	2 5
2 5	3 6	4 7	5 8	6 9	7 1	8 2	9 3	1 4
1 5	2 6	3 7	4 8	5 9	6 1	7 2	8 3	9 4
9 4	1 5	2 6	3 7	4 8	5 9	6 1	7 2	8 3
8 3	9 4	1 5	2 6	3 7	4 8	5 9	6 1	7 2
7 2	8 3	9 4	1 5	2 6	3 7	4 8	5 9	6 1
6 1	7 2	8 3	9 4	1 5	2 6	3 7	4 8	5 9
5 9	6 1	7 2	8 3	9 4	1 5	2 6	3 7	4 8
4 8	5 9	6 1	7 2	8 3	9 4	1 5	2 6	3 7
3 7	4 8	5 9	6 1	7 2	8 3	9 4	1 5	2 6
2 6	3 7	4 8	5 9	6 1	7 2	8 3	9 4	1 5

Figure 3.14: A $(36 \times 9)/2$ balanced semi-Latin rectangle for 9 treatments obtained by juxtaposition underneath

1	2	9	1	8	9	7	8	6	7	5	6	4	5	3	4	2	3
2	3	1	2	9	1	8	9	7	8	6	7	5	6	4	5	3	4
3	4	2	3	1	2	9	1	8	9	7	8	6	7	5	6	4	5
4	5	3	4	2	3	1	2	9	1	8	9	7	8	6	7	5	6
5	6	4	5	3	4	2	3	1	2	9	1	8	9	7	8	6	7
6	7	5	6	4	5	3	4	2	3	1	2	9	1	8	9	7	8
7	8	6	7	5	6	4	5	3	4	2	3	1	2	9	1	8	9
8	9	7	8	6	7	5	6	4	5	3	4	2	3	1	2	9	1
9	1	8	9	7	8	6	7	5	6	4	5	3	4	2	3	1	2
1	3	9	2	8	1	7	9	6	8	5	7	4	6	3	5	2	4
2	4	1	3	9	2	8	1	7	9	6	8	5	7	4	6	3	5
3	5	2	4	1	3	9	2	8	1	7	9	6	8	5	7	4	6
4	6	3	5	2	4	1	3	9	2	8	1	7	9	6	8	5	7
5	7	4	6	3	5	2	4	1	3	9	2	8	1	7	9	6	8
6	8	5	7	4	6	3	5	2	4	1	3	9	2	8	1	7	9
7	9	6	8	5	7	4	6	3	5	2	4	1	3	9	2	8	1
8	1	7	9	6	8	5	7	4	6	3	5	2	4	1	3	9	2
9	2	8	1	7	9	6	8	5	7	4	6	3	5	2	4	1	3
1	4	9	3	8	2	7	1	6	9	5	8	4	7	3	6	2	5
2	5	1	4	9	3	8	2	7	1	6	9	5	8	4	7	3	6
3	6	2	5	1	4	9	3	8	2	7	1	6	9	5	8	4	7
4	7	3	6	2	5	1	4	9	3	8	2	7	1	6	9	5	8
5	8	4	7	3	6	2	5	1	4	9	3	8	2	7	1	6	9
6	9	5	8	4	7	3	6	2	5	1	4	9	3	8	2	7	1
7	1	6	9	5	8	4	7	3	6	2	5	1	4	9	3	8	2
8	2	7	1	6	9	5	8	4	7	3	6	2	5	1	4	9	3
9	3	8	2	7	1	6	9	5	8	4	7	3	6	2	5	1	4
1	5	9	4	8	3	7	2	6	1	5	9	4	8	3	7	2	6
2	6	1	5	9	4	8	3	7	2	6	1	5	9	4	8	3	7
3	7	2	6	1	5	9	4	8	3	7	2	6	1	5	9	4	8
4	8	3	7	2	6	1	5	9	4	8	3	7	2	6	1	5	9
5	9	4	8	3	7	2	6	1	5	9	4	8	3	7	2	6	1
6	1	5	9	4	8	3	7	2	6	1	5	9	4	8	3	7	2
7	2	6	1	5	9	4	8	3	7	2	6	1	5	9	4	8	3
8	3	7	2	6	1	5	9	4	8	3	7	2	6	1	5	9	4
9	4	8	3	7	2	6	1	5	9	4	8	3	7	2	6	1	5

Figure 3.15: A $(36 \times 9)/2$ balanced semi-Latin rectangle for 9 treatments obtained by transposition of Figure 3.4

1 2	9 1	8 9	7 8	6 7	5 6	4 5	3 4	2 3	1 3	9 2	8 1	7 9	6 8	5 7	4 6	3 5	2 4	1 4	9 3	8 2	7 1	6 9	5 8	4 7	3 6	2 5	1 5	9 4	8 3	7 2	6 1	5 9	4 8	3 7	2 6
2 3	1 2	9 1	8 9	7 8	6 7	5 6	4 5	3 4	2 4	1 3	9 2	8 1	7 9	6 8	5 7	4 6	3 5	2 5	1 4	9 3	8 2	7 1	6 9	5 8	4 7	3 6	2 6	1 5	9 4	8 3	7 2	6 1	5 9	4 8	3 7
3 4	2 3	1 2	9 1	8 9	7 8	6 7	5 6	4 5	3 5	2 4	1 3	9 2	8 1	7 9	6 8	5 7	4 6	3 6	2 5	1 4	9 3	8 2	7 1	6 9	5 8	4 7	3 7	2 6	1 5	9 4	8 3	7 2	6 1	5 9	4 8
4 5	3 4	2 3	1 2	9 1	8 9	7 8	6 7	5 6	4 6	3 5	2 4	1 3	9 2	8 1	7 9	6 8	5 7	4 7	3 6	2 5	1 4	9 3	8 2	7 1	6 9	5 8	4 8	3 7	2 6	1 5	9 4	8 3	7 2	6 1	5 9
5 6	4 5	3 4	2 3	1 2	9 1	8 9	7 8	6 7	5 7	4 6	3 5	2 4	1 3	9 2	8 1	7 9	6 8	5 8	4 7	3 6	2 5	1 4	9 3	8 2	7 1	6 9	5 9	4 8	3 7	2 6	1 5	9 4	8 3	7 2	6 1
6 7	5 6	4 5	3 4	2 3	1 2	9 1	8 9	7 8	6 8	5 7	4 6	3 5	2 4	1 3	9 2	8 1	7 9	6 9	5 8	4 7	3 6	2 5	1 4	9 3	8 2	7 1	6 1	5 9	4 8	3 7	2 6	1 5	9 4	8 3	7 2
7 8	6 7	5 6	4 5	3 4	2 3	1 2	9 1	8 9	7 9	6 8	5 7	4 6	3 5	2 4	1 3	9 2	8 1	7 1	6 9	5 8	4 7	3 6	2 5	1 4	9 3	8 2	7 2	6 1	5 9	4 8	3 7	2 6	1 5	9 4	8 3
8 9	7 8	6 7	5 6	4 5	3 4	2 3	1 2	9 1	8 1	7 9	6 8	5 7	4 6	3 5	2 4	1 3	9 2	8 2	7 1	6 9	5 8	4 7	3 6	2 5	1 4	9 3	8 3	7 2	6 1	5 9	4 8	3 7	2 6	1 5	9 4
9 1	8 9	7 8	6 7	5 6	4 5	3 4	2 3	1 2	9 2	8 1	7 9	6 8	5 7	4 6	3 5	2 4	1 3	9 3	8 2	7 1	6 9	5 8	4 7	3 6	2 5	1 4	9 4	8 3	7 2	6 1	5 9	4 8	3 7	2 6	1 5

Figure 3.16: A $(9 \times 36)/2$ balanced semi-Latin rectangle for 9 treatments obtained by transposition of Figure 3.14

.

Similarly, suppose $v - 1 = 15$, then there are distinct admissible values for a and b which form the pairs, viz, $(a, b) = (3, 5)$ and $(5, 3)$. Hence, by an appropriate modification of step 3 of the algorithmic procedure in section 3.5.1, a $(24 \times 40)/2$ BSLR and $(40 \times 24)/2$ BSLR can be obtained.

(3) The aforementioned designs obtained for nonprime values of δ (or $v - 1$), where $1 < m, n < \delta$ (or $1 < a, b < v-1$) are some other possibilities of balanced semi-Latin rectangles obtained by modifying step 3 of the procedure.

However, if we allow $m = 1$ and $n = \delta$, for the designs in the first part of (2), we have a $(25 \times 300)/2$ BSLR, which is precisely, a basic design. Similarly, if we allow $m = \delta$ and $n = 1$, we have a $(300 \times 25)/2$ BSLR, the alternative basic design.

Furthermore, for the designs in the second part of (2), if we allow $a = 1$ and $b = v-1$, we have an $(8 \times 120)/2$ BSLR, which is precisely, the basic design . Similarly, if we allow $a = v-1$ and $b = 1$, we have a $(120 \times 8)/2$ BSLR , the alternative basic design.

1 ∞	2 3
2 3	1 ∞
2 ∞	3 1
3 1	2 ∞
3 ∞	1 2
1 2	3 ∞

Figure 3.17: A $(6 \times 2)/2$ balanced semi-Latin rectangle (BSLR) for 4 treatments obtained by juxtaposition underneath

(4) The double vertical and horizontal lines in the designs show the various points of juxtaposition.

3.6.3 Designs of inflated sizes

Given an $(h \times p)/2$ BSLR for v treatments from our basic (or modified basic) constructions, which is the basic (or alternative basic) design. We obtain some designs of larger sizes having the same number of treatments as the basic (or alternative basic) design by making multiple copies of it, and subsequently, making appropriate juxtaposition(s).

Suppose v is odd. We remind that $h = v$ and $p = v\delta$. Let the derived design be $(h' \times p')/2$, where $h' = ah$ and $p' = bp$ (a and b being positive integers and are not all 1s). We make b copies of the basic design, juxtapose them beside, and subsequently, make a copies of the resulting design and juxtapose them underneath. Alternatively, one can start by making a copies of the basic design, juxtaposing them underneath, and subsequently, making b copies of the resulting design and then juxtaposing them beside. In particular, $a = h'/h$ and $b = p'/p$.

Notice that, if $a = b = 1$, it reduces to the basic design. For the special case, where either $a = 1$ and $b > 1$ or $a > 1$ and $b = 1$, the juxtaposition is one-sided. In particular, if $a = 1$ and $b > 1$, then the construction simply involves making b copies of the basic design and juxtaposing them beside. Conversely, for the situation where $a > 1$ and $b = 1$, a copies of the basic design are made and the juxtaposition is underneath.

1	∞	2	5	3	4
3	4	1	∞	2	5
2	5	3	4	1	∞
2	∞	3	1	4	5
4	5	2	∞	3	1
3	1	4	5	2	∞
3	∞	4	2	5	1
5	1	3	∞	4	2
4	2	5	1	3	∞
4	∞	5	3	1	2
1	2	4	∞	5	3
5	3	1	2	4	∞
5	∞	1	4	2	3
2	3	5	∞	1	4
1	4	2	3	5	∞

Figure 3.18: A $(15 \times 3)/2$ balanced semi-Latin rectangle for 6 treatments obtained by juxtaposition underneath

Now, suppose v is even. Let the derived design be $(h^\dagger \times p^\dagger)/2$, where $h^\dagger = yh$ and $p^\dagger = zp$ (y and z being positive integers and are not all 1s); and reminding that $h = v/2$, $p = v(v-1)/2$. In a similar manner like in the case where v is odd, we make z copies of the basic design, juxtapose them beside, and subsequently, make y copies of the resulting design and juxtapose them underneath. This can also be achieved by first making y copies of the basic design, juxtaposing them underneath, and subsequently, making z copies of the resulting design and then juxtaposing them beside. We note that, $y = h^\dagger/h$ and $z = p^\dagger/p$.

It is obvious that, if $y = z = 1$, we have the basic design. Furthermore, if either $y = 1$ and $z > 1$ or $y > 1$ and $z = 1$, the juxtaposition is one-sided. In particular, if $y = 1$ and $z > 1$, then the construction simply involves making z copies of the basic design and juxtaposing them beside; and if $y > 1$ and $z = 1$, y copies of the basic design are made and the juxtaposition is underneath.

Comment. For the case where v is odd, suppose $p|h'$ and $h|p'$. Let $h' = sp$ and $p' = th$, where s and t are positive integers not all 1s. Then, the alternative basic design can be utilized, viz: make t copies of the alternative basic design, juxtapose them beside, and subsequently, make s copies of the resulting design and juxtapose them underneath.

1 ∞	3 4	2 5
2 5	1 ∞	3 4
3 4	2 5	1 ∞
2 ∞	4 5	3 1
3 1	2 ∞	4 5
4 5	3 1	2 ∞
3 ∞	5 1	4 2
4 2	3 ∞	5 1
5 1	4 2	3 ∞
4 ∞	1 2	5 3
5 3	4 ∞	1 2
1 2	5 3	4 ∞
5 ∞	2 3	1 4
1 4	5 ∞	2 3
2 3	1 4	5 ∞

Figure 3.19: A $(15 \times 3)/2$ balanced semi-Latin rectangle for 6 treatments obtained by transposition of the basic design

However, if these 2 conditions and the preceding ones are met, the basic design as well as the alternative basic design can be utilized for the construction of the derived design.

In a similar manner, for even v, suppose $p|h^{\dagger}$ and $h|p^{\dagger}$. Let $h^{\dagger} = fp$ and $p^{\dagger} = gh$, where f and g are positive integers not all 1s. Then, the alternative basic design can also be utilized by making g copies of the alternative basic design, juxtaposing them beside, and subsequently, making f copies of the resulting design and juxtapose them underneath.

1 ∞	3 4	2 5	2 ∞	4 5	3 1	3 ∞	5 1	4 2	4 ∞	1 2	5 3	5 ∞	2 3	1 4
2 5	1 ∞	3 4	3 1	2 ∞	4 5	4 2	3 ∞	5 1	5 3	4 ∞	1 2	1 4	5 ∞	2 3
3 4	2 5	1 ∞	4 5	3 1	2 ∞	5 1	4 2	3 ∞	1 2	5 3	4 ∞	2 3	1 4	5 ∞

Figure 3.20: A $(3 \times 15)/2$ balanced semi-Latin rectangle for 6 treatments obtained by transposition of the alternative basic design

1	∞	2	7	3	6	4	5
4	5	1	∞	2	7	3	6
3	6	4	5	1	∞	2	7
2	7	3	6	4	5	1	∞
2	∞	3	1	4	7	5	6
5	6	2	∞	3	1	4	7
4	7	5	6	2	∞	3	1
3	1	4	7	5	6	2	∞
3	∞	4	2	5	1	6	7
6	7	3	∞	4	2	5	1
5	1	6	7	3	∞	4	2
4	2	5	1	6	7	3	∞
4	∞	5	3	6	2	7	1
7	1	4	∞	5	3	6	2
6	2	7	1	4	∞	5	3
5	3	6	2	7	1	4	∞
5	∞	6	4	7	3	1	2
1	2	5	∞	6	4	7	3
7	3	1	2	5	∞	6	4
6	4	7	3	1	2	5	∞
6	∞	7	5	1	4	2	3
2	3	6	∞	7	5	1	4
1	4	2	3	6	∞	7	5
7	5	1	4	2	3	6	∞
7	∞	1	6	2	5	3	4
3	4	7	∞	1	6	2	5
2	5	3	4	7	∞	1	6
1	6	2	5	3	4	7	∞

Figure 3.21: A $(28 \times 4)/2$ balanced semi-Latin rectangle for 8 treatments obtained by juxtaposition underneath

1	∞	4	5	3	6	2	7
2	7	1	∞	4	5	3	6
3	6	2	7	1	∞	4	5
4	5	3	6	2	7	1	∞
2	∞	5	6	4	7	3	1
3	1	2	∞	5	6	4	7
4	7	3	1	2	∞	5	6
5	6	4	7	3	1	2	∞
3	∞	6	7	5	1	4	2
4	2	3	∞	6	7	5	1
5	1	4	2	3	∞	6	7
6	7	5	1	4	2	3	∞
4	∞	7	1	6	2	5	3
5	3	4	∞	7	1	6	2
6	2	5	3	4	∞	7	1
7	1	6	2	5	3	4	∞
5	∞	1	2	7	3	6	4
6	4	5	∞	1	2	7	3
7	3	6	4	5	∞	1	2
1	2	7	3	6	4	5	∞
6	∞	2	3	1	4	7	5
7	5	6	∞	2	3	1	4
1	4	7	5	6	∞	2	3
2	3	1	4	7	5	6	∞
7	∞	3	4	2	5	1	6
1	6	7	∞	3	4	2	5
2	5	1	6	7	∞	3	4
3	4	2	5	1	6	7	∞

Figure 3.22: A $(28 \times 4)/2$ balanced semi-Latin rectangle for 8 treatments obtained by transposition of the basic design

1 ∞	4 5	3 6	2 7	2 ∞	5 6	4 7	3 1	3 ∞	6 7	5 1	4 2	4 ∞	7 1	6 2	5 3	5 ∞	1 2	7 3	6 4	6 ∞	2 3	1 4	7 5	7 ∞	3 4	2 5	1 6
2 7	1 ∞	4 5	3 6	3 1	2 ∞	5 6	4 7	4 2	3 ∞	6 7	5 1	5 3	4 ∞	7 1	6 2	6 4	5 ∞	1 2	7 3	7 5	6 ∞	2 3	1 4	1 6	7 ∞	3 4	2 5
3 6	2 7	1 ∞	4 5	4 7	3 1	2 ∞	5 6	5 1	4 2	3 ∞	6 7	6 2	5 3	4 ∞	7 1	7 3	6 4	5 ∞	1 2	1 4	7 5	6 ∞	2 3	2 5	1 6	7 ∞	3 4
4 5	3 6	2 7	1 ∞	5 6	4 7	3 1	2 ∞	6 7	5 1	4 2	3 ∞	7 1	6 2	5 3	4 ∞	1 2	7 3	6 4	5 ∞	2 3	1 4	7 5	6 ∞	3 4	2 5	1 6	7 ∞

Figure 3.23: A $(4 \times 28)/2$ balanced semi-Latin rectangle for 8 treatments obtained by transposition of the alternative basic design

1 2	2 3	3 4	4 5	5 6	6 7	7 8	8 9	9 1	1 3	2 4	3 5	4 6	5 7	6 8	7 9	8 1	9 2
9 1	1 2	2 3	3 4	4 5	5 6	6 7	7 8	8 9	9 2	1 3	2 4	3 5	4 6	5 7	6 8	7 9	8 1
8 9	9 1	1 2	2 3	3 4	4 5	5 6	6 7	7 8	8 1	9 2	1 3	2 4	3 5	4 6	5 7	6 8	7 9
7 8	8 9	9 1	1 2	2 3	3 4	4 5	5 6	6 7	7 9	8 1	9 2	1 3	2 4	3 5	4 6	5 7	6 8
6 7	7 8	8 9	9 1	1 2	2 3	3 4	4 5	5 6	6 8	7 9	8 1	9 2	1 3	2 4	3 5	4 6	5 7
5 6	6 7	7 8	8 9	9 1	1 2	2 3	3 4	4 5	5 7	6 8	7 9	8 1	9 2	1 3	2 4	3 5	4 6
4 5	5 6	6 7	7 8	8 9	9 1	1 2	2 3	3 4	4 6	5 7	6 8	7 9	8 1	9 2	1 3	2 4	3 5
3 4	4 5	5 6	6 7	7 8	8 9	9 1	1 2	2 3	3 5	4 6	5 7	6 8	7 9	8 1	9 2	1 3	2 4
2 3	3 4	4 5	5 6	6 7	7 8	8 9	9 1	1 2	2 4	3 5	4 6	5 7	6 8	7 9	8 1	9 2	1 3
1 4	2 5	3 6	4 7	5 8	6 9	7 1	8 2	9 3	1 5	2 6	3 7	4 8	5 9	6 1	7 2	8 3	9 4
9 3	1 4	2 5	3 6	4 7	5 8	6 9	7 1	8 2	9 4	1 5	2 6	3 7	4 8	5 9	6 1	7 2	8 3
8 2	9 3	1 4	2 5	3 6	4 7	5 8	6 9	7 1	8 3	9 4	1 5	2 6	3 7	4 8	5 9	6 1	7 2
7 1	8 2	9 3	1 4	2 5	3 6	4 7	5 8	6 9	7 2	8 3	9 4	1 5	2 6	3 7	4 8	5 9	6 1
6 9	7 1	8 2	9 3	1 4	2 5	3 6	4 7	5 8	6 1	7 2	8 3	9 4	1 5	2 6	3 7	4 8	5 9
5 8	6 9	7 1	8 2	9 3	1 4	2 5	3 6	4 7	5 9	6 1	7 2	8 3	9 4	1 5	2 6	3 7	4 8
4 7	5 8	6 9	7 1	8 2	9 3	1 4	2 5	3 6	4 8	5 9	6 1	7 2	8 3	9 4	1 5	2 6	3 7
3 6	4 7	5 8	6 9	7 1	8 2	9 3	1 4	2 5	3 7	4 8	5 9	6 1	7 2	8 3	9 4	1 5	2 6
2 5	3 6	4 7	5 8	6 9	7 1	8 2	9 3	1 4	2 6	3 7	4 8	5 9	6 1	7 2	8 3	9 4	1 5

Figure 3.24: An $(18 \times 18)/2$ balanced semi-Latin rectangle for 9 treatments obtained by modifying step 3 of the algorithmic procedure

1 ∞	2 9	3 8	4 7	5 6	2 ∞	3 1	4 9	5 8	6 7	3 ∞	4 2	5 1	6 9	7 8
5 6	1 ∞	2 9	3 8	4 7	6 7	2 ∞	3 1	4 9	5 8	7 8	3 ∞	4 2	5 1	6 9
4 7	5 6	1 ∞	2 9	3 8	5 8	6 7	2 ∞	3 1	4 9	6 9	7 8	3 ∞	4 2	5 1
3 8	4 7	5 6	1 ∞	2 9	4 9	5 8	6 7	2 ∞	3 1	5 1	6 9	7 8	3 ∞	4 2
2 9	3 8	4 7	5 6	1 ∞	3 1	4 9	5 8	6 7	2 ∞	4 2	5 1	6 9	7 8	3 ∞
4 ∞	5 3	6 2	7 1	8 9	5 ∞	6 4	7 3	8 2	9 1	6 ∞	7 5	8 4	9 3	1 2
8 9	4 ∞	5 3	6 2	7 1	9 1	5 ∞	6 4	7 3	8 2	1 2	6 ∞	7 5	8 4	9 3
7 1	8 9	4 ∞	5 3	6 2	8 2	9 1	5 ∞	6 4	7 3	9 3	1 2	6 ∞	7 5	8 4
6 2	7 1	8 9	4 ∞	5 3	7 3	8 2	9 1	5 ∞	6 4	8 4	9 3	1 2	6 ∞	7 5
5 3	6 2	7 1	8 9	4 ∞	6 4	7 3	8 2	9 1	5 ∞	7 5	8 4	9 3	1 2	6 ∞
7 ∞	8 6	9 5	1 4	2 3	8 ∞	9 7	1 6	2 5	3 4	9 ∞	1 8	2 7	3 6	4 5
2 3	7 ∞	8 6	9 5	1 4	3 4	8 ∞	9 7	1 6	2 5	4 5	9 ∞	1 8	2 7	3 6
1 4	2 3	7 ∞	8 6	9 5	2 5	3 4	8 ∞	9 7	1 6	3 6	4 5	9 ∞	1 8	2 7
9 5	1 4	2 3	7 ∞	8 6	1 6	2 5	3 4	8 ∞	9 7	2 7	3 6	4 5	9 ∞	1 8
8 6	9 5	1 4	2 3	7 ∞	9 7	1 6	2 5	3 4	8 ∞	1 8	2 7	3 6	4 5	9 ∞

Figure 3.25: A $(15 \times 15)/2$ balanced semi-Latin rectangle for 10 treatments obtained by slightly modifying step 3 of the algorithmic procedure

Again, if these 2 conditions and the preceding ones are met, the basic design as well as the alternative basic design can be used for the construction.

We give some examples.

Case 1: Some examples when v is odd

Example 3.6.9. To make, for instance, a $(3 \times 6)/2$ BSLR for 3 treatments; notice that, the conditions for using the basic design and alternative basic design in the construction are satisfied, where $a = s = 1$ and $b = t = 2$ such that $h' = h = p$ and $p' = 2p = 2h$. Notice also that, $\delta = 1$, and we have a trivial case, where the alternative basic design is identical to the basic design.

We make $b = t = 2$ copies of the $(3 \times 3)/2$ BSLR in Figure 3.1 which serves as both the basic and alternative basic design, and then juxtapose them beside. The resulting design is shown in Figure 3.26, where the double vertical lines show the point of juxtaposition.

In a similar manner, to obtain a $(6 \times 3)/2$ BSLR for the same number of treatments as above; notice that $a = s = 2$ and $b = t = 1$ such that $h' = 2h = 2p$ and $p' = p = h$. Thus, we make $a = s = 2$ copies of the basic design/alternative basic design and juxtapose underneath: see Figure 3.27.

1 2	2 3	3 1	1 2	2 3	3 1
3 1	1 2	2 3	3 1	1 2	2 3
2 3	3 1	1 2	2 3	3 1	1 2

Figure 3.26: A $(3 \times 6)/2$ balanced semi-Latin rectangle for 3 treatments

1 2	2 3	3 1
3 1	1 2	2 3
2 3	3 1	1 2
1 2	2 3	3 1
3 1	1 2	2 3
2 3	3 1	1 2

Figure 3.27: A $(6 \times 3)/2$ balanced semi-Latin rectangle for 3 treatments

Remark. Another $(6 \times 3)/2$ BSLR for 3 treatments can be obtained by transposing the $(3 \times 6)/2$ BSLR with 3 treatments. Similarly, transposing the $(6 \times 3)/2$ BSLR obtained by juxtaposition underneath produces another $(3 \times 6)/2$ BSLR.

If interest is to make, say, a $(3 \times 12)/2$ BSLR, this can be achieved by making either 4 copies of the basic design and juxtaposing them beside or by simply making 2 copies of the $(3 \times 6)/2$ BSLR and also juxtaposing them beside.

Comment. The double vertical and horizontal lines are used for construction purposes and show the points of juxtaposition.

Example 3.6.10. Suppose we wish to make a $(6 \times 9)/2$ BSLR for 3 treatments, we start by making $b = 3$ copies of the basic design and juxtaposing them beside: this gives a $(3 \times 9)/2$ BSLR, which we then make $a = 2$ copies of it and juxtapose underneath to obtain the required design. Alternatively, this construction can be approached by first making $a = 2$ copies of the basic design and juxtaposing them underneath, which gives rise to a $(6 \times 3)/2$ BSLR; and subsequently, $b = 3$ copies of the resulting $(6 \times 3)/2$ BSLR are then made and juxtaposed beside to obtain the required design. The required $(6 \times 9)/2$ BSLR is shown in Figure 3.28.

1 2	2 3	3 1	1 2	2 3	3 1	1 2	2 3	3 1
3 1	1 2	2 3	3 1	1 2	2 3	3 1	1 2	2 3
2 3	3 1	1 2	2 3	3 1	1 2	2 3	3 1	1 2
1 2	2 3	3 1	1 2	2 3	3 1	1 2	2 3	3 1
3 1	1 2	2 3	3 1	1 2	2 3	3 1	1 2	2 3
2 3	3 1	1 2	2 3	3 1	1 2	2 3	3 1	1 2

Figure 3.28: A $(6 \times 9)/2$ balanced semi-Latin rectangle for 3 treatments

Example 3.6.11. Suppose interest is to make a $(5 \times 30)/2$ BSLR for 5 treatments, we make 3 copies of the $(5 \times 10)/2$ BSLR in Figure 3.2, which serves as the basic design in this case and juxtapose them beside. Clearly, in this example, $a = 1$ and $b = 3$. However, $p \nmid h'$. Hence the alternative basic design cannot be used for the construction. The desired design is presented in Figure 3.29.

Similarly, a $(20 \times 10)/2$ BSLR for 5 treatments can be made by first making 4 copies of the parent design and subsequently, juxtaposing them underneath (since $a = 4$ and $b = 1$): see Figure 3.30.

Remark. By making $t = 2$ copies of the alternative basic design, juxtaposing them beside, and subsequently, making $s = 2$ copies of the resulting design and juxtapose them underneath gives another $(20 \times 10)/2$ BSLR for 5 treatments.

1 2	2 3	3 4	4 5	5 1	1 3	2 4	3 5	4 1	5 2	1 2	2 3	3 4	4 5	5 1	1 3	2 4	3 5	4 1	5 2	1 2	2 3	3 4	4 5	5 1	1 3	2 4	3 5	4 1	5 2
5 1	1 2	2 3	3 4	4 5	5 2	1 3	2 4	3 5	4 1	5 1	1 2	2 3	3 4	4 5	5 2	1 3	2 4	3 5	4 1	5 1	1 2	2 3	3 4	4 5	5 2	1 3	2 4	3 5	4 1
4 5	5 1	1 2	2 3	3 4	4 1	5 2	1 3	2 4	3 5	4 5	5 1	1 2	2 3	3 4	4 1	5 2	1 3	2 4	3 5	4 5	5 1	1 2	2 3	3 4	4 1	5 2	1 3	2 4	3 5
3 4	4 5	5 1	1 2	2 3	3 5	4 1	5 2	1 3	2 4	3 4	4 5	5 1	1 2	2 3	3 5	4 1	5 2	1 3	2 4	3 4	4 5	5 1	1 2	2 3	3 5	4 1	5 2	1 3	2 4
2 3	3 4	4 5	5 1	1 2	2 4	3 5	4 1	5 2	1 3	2 3	3 4	4 5	5 1	1 2	2 4	3 5	4 1	5 2	1 3	2 3	3 4	4 5	5 1	1 2	2 4	3 5	4 1	5 2	1 3

Figure 3.29: A $(5 \times 30)/2$ balanced semi-Latin rectangle for 5 treatments

Case 2: Some examples when v is even

Example 3.6.12. To make, for instance, a $(2 \times 12)/2$ BSLR for 4 treatments, we make $z = 2$ copies of the $(2 \times 6)/2$ BSLR in Figure 3.5 which serves as the basic design and then juxtapose them beside. The resulting design is shown in Figure 3.31, where the double vertical lines show the point of juxtaposition.

In a similar manner, to obtain a $(4 \times 6)/2$ BSLR for the same number of treatments as above, we also make 2 copies of the basic design, but this time, we juxtapose them underneath as shown in Figure 3.32

Notice that, in this example, $h' = h$ and $p' = 2p$ such that $y = 1$ and $z = 2$, for the earlier design; while for the latter design, $h' = 2h$ and $p' = p$ such that $y = 2$ and $z = 1$.

Comments. (1) To make, for instance, a $(6 \times 4)/2$ BSLR. Notice that $h^\dagger = p$, hence $f = 1$. Similarly, $p^\dagger = 2h$ such that $g = 2$. Hence, we simply make $g = 2$ copies of the alternative basic design shown in Figure 3.17 and juxtapose them beside: see Figure 3.33. Alternatively, the same design can be obtained by a transposition of the $(4 \times 6)/2$ BSLR. It is obvious that $p \nmid p^\dagger$, hence the basic design cannot be used directly.

(2) The triple vertical lines in Figure 3.33 is for purposes of construction; it shows the point of juxtaposition of the 2 copies of the alternative basic design.

(3) Suppose interest is to make, say, a $(6 \times 6)/2$ BSLR, this can be achieved by making 3 copies of the basic design and juxtaposing them underneath. Another version of this design can be obtained by making 3 copies of the alternative basic design and subsequently, juxtaposing them beside: see Figures 3.34 and 3.35, respectively.

Example 3.6.13. Suppose interest is to make a $(15 \times 15)/2$ BSLR for 6 treatments. Notice that $y = 5$, $z = 1$, $f = 1$ and $g = 5$. Hence, both the basic design and the alternative basic design can be utilized just like in the construction of the $(6 \times 6)/2$ BSLR in the comments section of Example 3.6.12.

1 2	2 3	3 4	4 5	5 1	1 3	2 4	3 5	4 1	5 2
5 1	1 2	2 3	3 4	4 5	5 2	1 3	2 4	3 5	4 1
4 5	5 1	1 2	2 3	3 4	4 1	5 2	1 3	2 4	3 5
3 4	4 5	5 1	1 2	2 3	3 5	4 1	5 2	1 3	2 4
2 3	3 4	4 5	5 1	1 2	2 4	3 5	4 1	5 2	1 3
1 2	2 3	3 4	4 5	5 1	1 3	2 4	3 5	4 1	5 2
5 1	1 2	2 3	3 4	4 5	5 2	1 3	2 4	3 5	4 1
4 5	5 1	1 2	2 3	3 4	4 1	5 2	1 3	2 4	3 5
3 4	4 5	5 1	1 2	2 3	3 5	4 1	5 2	1 3	2 4
2 3	3 4	4 5	5 1	1 2	2 4	3 5	4 1	5 2	1 3
1 2	2 3	3 4	4 5	5 1	1 3	2 4	3 5	4 1	5 2
5 1	1 2	2 3	3 4	4 5	5 2	1 3	2 4	3 5	4 1
4 5	5 1	1 2	2 3	3 4	4 1	5 2	1 3	2 4	3 5
3 4	4 5	5 1	1 2	2 3	3 5	4 1	5 2	1 3	2 4
2 3	3 4	4 5	5 1	1 2	2 4	3 5	4 1	5 2	1 3
1 2	2 3	3 4	4 5	5 1	1 3	2 4	3 5	4 1	5 2
5 1	1 2	2 3	3 4	4 5	5 2	1 3	2 4	3 5	4 1
4 5	5 1	1 2	2 3	3 4	4 1	5 2	1 3	2 4	3 5
3 4	4 5	5 1	1 2	2 3	3 5	4 1	5 2	1 3	2 4
2 3	3 4	4 5	5 1	1 2	2 4	3 5	4 1	5 2	1 3

Figure 3.30: A $(20 \times 10)/2$ balanced semi-Latin rectangle for 5 treatments

1 ∞	2 3	2 ∞	3 1	3 ∞	1 2	1 ∞	2 3	2 ∞	3 1	3 ∞	1 2
2 3	1 ∞	3 1	2 ∞	1 2	3 ∞	2 3	1 ∞	3 1	2 ∞	1 2	3 ∞

Figure 3.31: A $(2 \times 12)/2$ balanced semi-Latin rectangle (BSLR) for 4 treatments

1 ∞	2 3	2 ∞	3 1	3 ∞	1 2
2 3	1 ∞	3 1	2 ∞	1 2	3 ∞
1 ∞	2 3	2 ∞	3 1	3 ∞	1 2
2 3	1 ∞	3 1	2 ∞	1 2	3 ∞

Figure 3.32: A $(4 \times 6)/2$ balanced semi-Latin rectangle (BSLR) for 4 treatments

1 ∞	2 3	1 ∞	2 3
2 3	1 ∞	2 3	1 ∞
2 ∞	3 1	2 ∞	3 1
3 1	2 ∞	3 1	2 ∞
3 ∞	1 2	3 ∞	1 2
1 2	3 ∞	1 2	3 ∞

Figure 3.33: A $(6 \times 4)/2$ balanced semi-Latin rectangle (BSLR) for 4 treatments obtained by juxtaposition of 2 copies of the alternative basic design beside

We make $y = 5$ copies of the $(3 \times 15)/2$ BSLR in Figure 3.6, which serves as the basic design in this case, and juxtapose them underneath: see Figure 3.36. Another version of this design can be obtained by making $g = 5$ copies of the alternative basic design shown in Figure 3.18 and juxtaposing them beside. The resulting design is presented in Figure 3.37.

Comment. The double horizontal and vertical lines in Figures 3.36 and 3.37 show the respective point of juxtapositions of copies of the basic and alternative basic designs.

Example 3.6.14. To make a $(9 \times 30)/2$ BSLR for 6 treatments via juxtaposition, we

1	∞	2	3	2	∞	3	1	3	∞	1	2
2	3	1	∞	3	1	2	∞	1	2	3	∞
1	∞	2	3	2	∞	3	1	3	∞	1	2
2	3	1	∞	3	1	2	∞	1	2	3	∞
1	∞	2	3	2	∞	3	1	3	∞	1	2
2	3	1	∞	3	1	2	∞	1	2	3	∞

Figure 3.34: A $(6 \times 6)/2$ balanced semi-Latin rectangle (BSLR) for 4 treatments obtained by juxtaposition of 3 copies of the basic design underneath

1	∞	2	3	1	∞	2	3	1	∞	2	3
2	3	1	∞	2	3	1	∞	2	3	1	∞
2	∞	3	1	2	∞	3	1	2	∞	3	1
3	1	2	∞	3	1	2	∞	3	1	2	∞
3	∞	1	2	3	∞	1	2	3	∞	1	2
1	2	3	∞	1	2	3	∞	1	2	3	∞

Figure 3.35: A $(6 \times 6)/2$ balanced semi-Latin rectangle (BSLR) for 4 treatments obtained by juxtaposition of 3 copies of the alternative basic design beside

observe that $p = 15$ and $h^{\dagger} = 9$, hence $p \nmid h^{\dagger}$ and we cannot use the alternative basic design. However, $h^{\dagger} = 3h$ and $p^{\dagger} = 2p$, giving $y = 3$ and $z = 2$. Thus, we utilize the

1 ∞	2 5	3 4	2 ∞	3 1	4 5	3 ∞	4 2	5 1	4 ∞	5 3	1 2	5 ∞	1 4	2 3
3 4	1 ∞	2 5	4 5	2 ∞	3 1	5 1	3 ∞	4 2	1 2	4 ∞	5 3	2 3	5 ∞	1 4
2 5	3 4	1 ∞	3 1	4 5	2 ∞	4 2	5 1	3 ∞	5 3	1 2	4 ∞	1 4	2 3	5 ∞
1 ∞	2 5	3 4	2 ∞	3 1	4 5	3 ∞	4 2	5 1	4 ∞	5 3	1 2	5 ∞	1 4	2 3
3 4	1 ∞	2 5	4 5	2 ∞	3 1	5 1	3 ∞	4 2	1 2	4 ∞	5 3	2 3	5 ∞	1 4
2 5	3 4	1 ∞	3 1	4 5	2 ∞	4 2	5 1	3 ∞	5 3	1 2	4 ∞	1 4	2 3	5 ∞
1 ∞	2 5	3 4	2 ∞	3 1	4 5	3 ∞	4 2	5 1	4 ∞	5 3	1 2	5 ∞	1 4	2 3
3 4	1 ∞	2 5	4 5	2 ∞	3 1	5 1	3 ∞	4 2	1 2	4 ∞	5 3	2 3	5 ∞	1 4
2 5	3 4	1 ∞	3 1	4 5	2 ∞	4 2	5 1	3 ∞	5 3	1 2	4 ∞	1 4	2 3	5 ∞
1 ∞	2 5	3 4	2 ∞	3 1	4 5	3 ∞	4 2	5 1	4 ∞	5 3	1 2	5 ∞	1 4	2 3
3 4	1 ∞	2 5	4 5	2 ∞	3 1	5 1	3 ∞	4 2	1 2	4 ∞	5 3	2 3	5 ∞	1 4
2 5	3 4	1 ∞	3 1	4 5	2 ∞	4 2	5 1	3 ∞	5 3	1 2	4 ∞	1 4	2 3	5 ∞
1 ∞	2 5	3 4	2 ∞	3 1	4 5	3 ∞	4 2	5 1	4 ∞	5 3	1 2	5 ∞	1 4	2 3
3 4	1 ∞	2 5	4 5	2 ∞	3 1	5 1	3 ∞	4 2	1 2	4 ∞	5 3	2 3	5 ∞	1 4
2 5	3 4	1 ∞	3 1	4 5	2 ∞	4 2	5 1	3 ∞	5 3	1 2	4 ∞	1 4	2 3	5 ∞

Figure 3.36: A $(15 \times 15)/2$ balanced semi-Latin rectangle for 6 treatments obtained via the basic design

1 ∞	2 5	3 4	1 ∞	2 5	3 4	1 ∞	2 5	3 4	1 ∞	2 5	3 4	1 ∞	2 5	3 4
3 4	1 ∞	2 5	3 4	1 ∞	2 5	3 4	1 ∞	2 5	3 4	1 ∞	2 5	3 4	1 ∞	2 5
2 5	3 4	1 ∞	2 5	3 4	1 ∞	2 5	3 4	1 ∞	2 5	3 4	1 ∞	2 5	3 4	1 ∞
2 ∞	3 1	4 5	2 ∞	3 1	4 5	2 ∞	3 1	4 5	2 ∞	3 1	4 5	2 ∞	3 1	4 5
4 5	2 ∞	3 1	4 5	2 ∞	3 1	4 5	2 ∞	3 1	4 5	2 ∞	3 1	4 5	2 ∞	3 1
3 1	4 5	2 ∞	3 1	4 5	2 ∞	3 1	4 5	2 ∞	3 1	4 5	2 ∞	3 1	4 5	2 ∞
3 ∞	4 2	5 1	3 ∞	4 2	5 1	3 ∞	4 2	5 1	3 ∞	4 2	5 1	3 ∞	4 2	5 1
5 1	3 ∞	4 2	5 1	3 ∞	4 2	5 1	3 ∞	4 2	5 1	3 ∞	4 2	5 1	3 ∞	4 2
4 2	5 1	3 ∞	4 2	5 1	3 ∞	4 2	5 1	3 ∞	4 2	5 1	3 ∞	4 2	5 1	3 ∞
4 ∞	5 3	1 2	4 ∞	5 3	1 2	4 ∞	5 3	1 2	4 ∞	5 3	1 2	4 ∞	5 3	1 2
1 2	4 ∞	5 3	1 2	4 ∞	5 3	1 2	4 ∞	5 3	1 2	4 ∞	5 3	1 2	4 ∞	5 3
5 3	1 2	4 ∞	5 3	1 2	4 ∞	5 3	1 2	4 ∞	5 3	1 2	4 ∞	5 3	1 2	4 ∞
5 ∞	1 4	2 3	5 ∞	1 4	2 3	5 ∞	1 4	2 3	5 ∞	1 4	2 3	5 ∞	1 4	2 3
2 3	5 ∞	1 4	2 3	5 ∞	1 4	2 3	5 ∞	1 4	2 3	5 ∞	1 4	2 3	5 ∞	1 4
1 4	2 3	5 ∞	1 4	2 3	5 ∞	1 4	2 3	5 ∞	1 4	2 3	5 ∞	1 4	2 3	5 ∞

Figure 3.37: A $(15 \times 15)/2$ balanced semi-Latin rectangle for 6 treatments obtained via the alternative basic design

1 ∞	2 5	3 4	2 ∞	3 1	4 5	3 ∞	4 2	5 1	4 ∞	5 3	1 2	5 ∞	1 4	2 3	1 ∞	2 5	3 4	2 ∞	3 1	4 5	3 ∞	4 2	5 1	4 ∞	5 3	1 2	5 ∞	1 4	2 3
3 4	1 ∞	2 5	4 5	2 ∞	3 1	5 1	3 ∞	4 2	1 2	4 ∞	5 3	2 3	5 ∞	1 4	3 4	1 ∞	2 5	4 5	2 ∞	3 1	5 1	3 ∞	4 2	1 2	4 ∞	5 3	2 3	5 ∞	1 4
2 5	3 4	1 ∞	3 1	4 5	2 ∞	4 2	5 1	3 ∞	5 3	1 2	4 ∞	1 4	2 3	5 ∞	2 5	3 4	1 ∞	3 1	4 5	2 ∞	4 2	5 1	3 ∞	5 3	1 2	4 ∞	1 4	2 3	5 ∞
1 ∞	2 5	3 4	2 ∞	3 1	4 5	3 ∞	4 2	5 1	4 ∞	5 3	1 2	5 ∞	1 4	2 3	1 ∞	2 5	3 4	2 ∞	3 1	4 5	3 ∞	4 2	5 1	4 ∞	5 3	1 2	5 ∞	1 4	2 3
3 4	1 ∞	2 5	4 5	2 ∞	3 1	5 1	3 ∞	4 2	1 2	4 ∞	5 3	2 3	5 ∞	1 4	1 ∞	2 5	3 4	2 ∞	3 1	4 5	3 ∞	4 2	5 1	4 ∞	5 3	1 2	5 ∞	1 4	2 3
2 5	3 4	1 ∞	3 1	4 5	2 ∞	4 2	5 1	3 ∞	5 3	1 2	4 ∞	1 4	2 3	5 ∞	2 5	3 4	1 ∞	3 1	4 5	2 ∞	4 2	5 1	3 ∞	5 3	1 2	4 ∞	1 4	2 3	5 ∞
1 ∞	2 5	3 4	2 ∞	3 1	4 5	3 ∞	4 2	5 1	4 ∞	5 3	1 2	5 ∞	1 4	2 3	1 ∞	2 5	3 4	2 ∞	3 1	4 5	3 ∞	4 2	5 1	4 ∞	5 3	1 2	5 ∞	1 4	2 3
3 4	1 ∞	2 5	4 5	2 ∞	3 1	5 1	3 ∞	4 2	1 2	4 ∞	5 3	2 3	5 ∞	1 4	3 4	1 ∞	2 5	4 5	2 ∞	3 1	5 1	3 ∞	4 2	1 2	4 ∞	5 3	2 3	5 ∞	1 4
2 5	3 4	1 ∞	3 1	4 5	2 ∞	4 2	5 1	3 ∞	5 3	1 2	4 ∞	1 4	2 3	5 ∞	2 5	3 4	1 ∞	3 1	4 5	2 ∞	4 2	5 1	3 ∞	5 3	1 2	4 ∞	1 4	2 3	5 ∞

Figure 3.38: A $(9 \times 30)/2$ balanced semi-Latin rectangle for 6 treatments

basic design by making 2 copies of it, juxtaposing them beside, and subsequently, making 3 copies of the resulting design and juxtaposing them underneath: see Figure 3.38

Remark. The juxtaposition is in two stages. The design resulting from the first stage of juxtaposition is a $(3 \times 30)/2$ BSLR. A juxtaposition of 3 copies of it gives the required $(9 \times 30)/2$ BSLR.

In a similar way, the same design can be obtained by first making 3 copies of the basic design, juxtaposing them underneath, and subsequently, making 2 copies of the resulting design and then juxtaposing them beside. In this case, a $(9 \times 15)/2$ BSLR is obtained at the first stage of juxtaposition, whose 2 copies juxtaposed beside produces the desired design.

Chapter 4

Balanced Semi-Latin Rectangles with Larger Block Sizes

4.1 Introduction

This chapter focuses on balanced semi-Latin rectangles with block sizes greater than two, that is, there are more than two treatments in each row-column intersection of the design. Just like the designs discussed in Chapter 3, for these designs, their quotient block designs are BIBDs. We denote the structure of this design for a given number, v, of treatments by $(h \times p)/k$, where $k > 2$. Each treatment appears khp/v times, overall, in the design. It appears $n_r = kp/v$ times in each row and $n_c = kh/v$ times in each column. Furthermore, as noted in section 3.2, the values of h and p are not necessarily distinct.

Some algorithmic procedures are given for constructing designs of different classes. More designs are obtained via some modifications of the algorithms, by transpositions and also by employing complementation of different kinds. Designs of larger sizes are also obtained by making multiple copies of designs of smaller sizes and then putting them in an array of appropriate size. In some cases, we also make use of Latin squares of different compositions to make more designs of larger sizes that have identical number of rows and columns if certain conditions are satisfied. We give, in addition, some examples to illustrate the constructions.

4.2 Construction Approaches

We give some constructions for these designs using some procedures such as a modified version of the distance approach we used in the preceding chapter. Some concepts such as distance, difference sets/difference families, affine resolvability and complementation are also utilized in the construction. For those classes of designs that we give a direct construction for, having v treatments, h rows, p columns and block (cell) size k, we obtain an equivalent design–having the same value of h, p, and v, though with block size, k' via block complementation, where $k' = v - k$. This involves taking a BSLR for a given h, p, v and k obtained by direct construction, and putting in each cell those treatments that are missing from it. We adopt this approach, in particular, when $k' > k$.

If $k' = k$, then $n_c = 1$ if and only if $h = 2$. In this case, we adopt another form of complementation for the construction, we name it column complementation. This involves filling the cells in row 1 of a $2 \times p$ array with appropriate entries (where the entries of these cells form the p blocks of a BIBD) and then putting in the cell in row 2 of each column those treatments that are missing from the cell directly above it.

We also employ another form of complementation that we name row complementation. This would be used to obtain construction for designs with $p = 2$ and $k' = k$, hence $n_r = 1$. We note that if $k' = k$, then $n_r = 1$ if and only if $p = 2$. The procedure is akin to the construction by column complementation and involves filling each cell in column 2 of an $h \times 2$ array by the complement of the set of treatments of the cell in column 1 of the same row, where the entries of the h cells in column 1 form the h blocks of a BIBD.

We note that, in general, if $k' = k$, then $n_c = t$ if and only if $h = 2t$, where $t = 1, 2, ...$ Similarly, if $k' = k$, then $n_r = u$ if and only if $p = 2u$, where $u = 1, 2, ...$

4.3 Constructions based on distances

Let $V = \{1, 2, ..., v\}$ denote the set of treatments of the design under construction, where v is odd. We obtain constructions for BSLRs with $k = 3$. We begin by identifying the treatments with the vertices of a regular v-gon and then forming triples/blocks by combining each vertex, i with adjacent vertices i' and i'', each being equidistant from i, with distance, $d(i, i') = d(i, i'') = 1$, where $i, i', i'' = 1, 2, ..., v$, $i \neq i' \neq i''$. This is repeated for all values of i with the nonadjacent vertices, i' and i'' for which $d(i, i') = d(i, i'') = 2, 3, ..., \delta$, where $\delta = (v-1)/2$, since v is odd. Hence, overall, $v\delta$ triples are generated and utilized in the construction.

Given a vertex i with 2 distinct vertices, i' and i'', each being equidistant from i and which combine with i to form the triple, $\{i, i', i''\}$, where $i, i', i'' = 1, 2, ..., v$, $i \neq i' \neq i''$. For each $l \in \{1, 2, ..., \delta\}$, let S_{lw} denote the wth triple formed such that $d(i, i') = d(i, i'') = l$. Then $S_{lw} = \{w, w + l, w - l\}$, where each component is reduced modulo v, for all $l = 1, 2, 3, ..., \delta$. and $w = 1, 2, ..., v$.

Notice that, $S_{l1} = \{1, 1 + l, 1 - l\}$, $S_{l2} = \{2, 2 + l, 2 - l\}$, . . ., $S_{lv} = \{v, v + l, v - l\}$. Similarly, $S_{1w} = \{w, w + 1, w - 1\}$, $S_{2w} = \{w, w + 2, w - 2\}$, . . ., $S_{\delta w} = \{w, w + \delta, w - \delta\}$.

For each $l = 1, 2, 3, ..., \delta$, a Latin square of order v is made using S_{l1}, S_{l2},...,S_{lv} as symbols and these are then inserted into an array of appropriate size to obtain the required design.

Theorem 4.3.1. *The set, $\{S_{lw}\}_{w=1}^{v}$, of all generated triples that facilitates the construction of the balanced semi-Latin rectangle, where $l = 1, 2, ..., \delta$; $\delta = (v-1)/2$ and v is* *odd forms a 3-resolvable* $(v, v\delta, 3\delta, 3, 3)$*-BIBD, where* δ *denote the number of 3-resolution classes.*

Proof. Let $V = \{1, 2, ..., v\}$ denote the set of vertices of a v-gon which corresponds to the set of treatments of a design. Relabel these vertices in a cyclic order as $i, i+1, i+2, ..., i+(\delta-1), i+\delta, i-\delta, i-(\delta-1), ..., i-2, i-1$, where $i \in V$, and the addition/subtraction is performed modulo v. For each $i \in V$ and $i' \in V\backslash\{i\}$, it follows that $i' = i \pm d(i, i')$, where $d(i, i')$ denote the distance between the vertices i and i', defined by

$$d(i, i') = \begin{cases} |i' - i| & \text{if } |i' - i| \leq \delta, \\ \\ v - |i' - i| & \text{if } |i' - i| > \delta. \end{cases} \tag{4.1}$$

In particular, for all $i < i'$,

$$i' = \begin{cases} i + d(i, i') & \text{if } |i' - i| \leq \delta, \\ \\ i - d(i, i') & \text{if } |i' - i| > \delta. \end{cases}$$

Similarly, if $i > i'$, the positive and negative signs are exchanged, that is,

$$i' = \begin{cases} i - d(i, i') & \text{if } |i' - i| \leq \delta, \\ \\ i + d(i, i') & \text{if } |i' - i| > \delta. \end{cases}$$

By the construction, for each $l \in \{1, 2, 3, ..., \delta\}$, each vertex, i associates with the pair $(i-l, i+l)$, which are equidistant from it to form a triple/block, $\{i, i+l, i-l\}$, where $i = 1, 2, ..., v$ (hence v triples are formed) such that $d(i, i+l) = d(i, i-l) = l$. Clearly, each of the vertices $i+l$ and $i-l$ is distinct from i, since $l \neq 0$. Similarly, $d(i-l, i+l) = 2l$ or $v - 2l$, which are both nonzero, since $2l$ is even and v is odd, and also $l \neq 0$. Obviously, both $i-l$ and $i+l$ are also distinct from each other, for all $i \in V$, hence no two vertices within a set of triples are identical, which makes the design binary. Now, since $d(i, i+l) = d(i, i-l) = l$, for all l, then it follows that, i and $i+l$ associate with $i-l$ and $i+2l$. Similarly, i and $i-l$ associate with $i+l$ and $i-2l$. Notice that, in these cases, $d(i-l, i+2l) = d(i+l, i-2l)$, which by (4.1) has the value $3l$ if $3l \leq \delta$, such that $l \leq \delta/3$; or has the value $v - 3l$, if $3l > \delta$, such that $l > \delta/3$. Suppose $l = v/3$. Then $l > \delta/3$, since $v > \delta$ such that $d(i-l, i+2l) = d(i+l, i-2l)$ has the value $v - 3l$. Now, if $l = v/3$, then $v - 3l = 0$. Conversely, if $v - 3l = 0$, then $l = v/3$. Hence $v - 3l = 0$ if and only if $l = v/3$. Thus, the vertices, $i - l$ and $i + 2l$ are identical if and only if $l = v/3$, hence under this condition, each forms identical triple with the pair, $(i, i + l)$ of vertices. Similarly, the vertices $i + l$ and $i - 2l$ are also identical if and only if $l = v/3$, and under this condition, each forms identical triple with the pair, $(i, i - l)$ of vertices.

Furthermore, since v is odd, then $v - 2$ is also odd. Suppose i and $i + l$ are two vertices. Then by the construction, there exists a vertex, i^*, say, such that $d(i, i^*) = d(i + l, i^*)$, that is, i and $i + l$ are equidistant from i^*.

Similarly, suppose i and $i - l$ are two vertices. Then i^* is such that $d(i, i^*) = d(i - l, i^*)$.

From the foregoing discussion, it follows that the pair, $(i, i + l)$ of vertices forms sets of triples with $i - l$, $i + 2l$ and i^*, viz, $\{i, i + l, i - l\}$, $\{i, i + l, i + 2l\}$, and $\{i, i + l, i^*\}$. Similarly, the pair $(i, i - l)$ forms the triples $\{i, i - l, i - 2l\}$, $\{i, i - l, i + l\}$, and $\{i, i - l, i^*\}$. Thus every pair of vertices appear together in 3 triples (blocks), making $\lambda = 3$, where only the pairs $(i, i \pm v/3)$, for any $i \in V$ form identical sets of triples each time they appear together.

Since, for each $l \in \{1, 2, 3, ..., \delta\}$, there are v triples (blocks), then there are $v\delta$ blocks, overall. Moreover, for each $l \in \{1, 2, 3, ..., \delta\}$, each vertex (treatment), $i \in V$ appears in 3 blocks, hence it appears in 3δ blocks, overall. There are 3 plots in each block, hence, for each l, there are $3v$ plots, and overall, there are $3v\delta$ plots. Moreover, each treatment appears at most once in each block. Since each treatment appears in 3δ blocks/plots in the design, then, overall, the v distinct treatments appear in $3\delta v$ plots, which is identical to the total number of plots.

Hence the design has v vertices/treatments arranged in $v\delta$ incomplete blocks of size 3 which are divided into δ 3-resolution classes, where each $l \in \{1, 2, ..., \delta\}$ corresponds to a 3-resolution class and contains v blocks, and each treatment appears in 3 blocks within each 3-resolution class, hence 3δ times, overall. Furthermore, overall, each pair of treatments appears together in 3 blocks. It follows that the set of all $v\delta$ triples form a 3-resolvable $(v, v\delta, 3\delta, 3, 3)$-BIBD; and by putting $i = w$, for all $i \in V$, the theorem follows.

■

Corollary 4.3.1. *The BIBD has repeated blocks, each with multiplicity 3 if and only if* $l = v/3$. *Consequently, the support size, the number of distinct blocks in the BIBD is* $v(\delta - 1) + v/3 = v\delta - 2v/3$, *where* $\delta = (v - 1)/2$.

Corollary 4.3.2. *If* $3|v$, *then by Corollary 4.3.1, the number of distinct blocks in the 3-resolution class associated with* $l = v/3$ *is precisely* $v/3$. *It follows from Corollary 4.3.1 that the number of distinct blocks in the BIBD is* $2v/3$ *less the total number of blocks.*

4.3.1 Construction for designs of the class $(v \times \delta v)/3$, where $\delta = (v-1)/2 > 1$ and v is odd

An array of size $v \times \delta v$ is created whose columns are divided into δ equal subdivisions, separated by double vertical lines and the Latin squares are then inserted, one to each subdivision.

An algorithmic procedure for constructing the design

1. For each $l = 1, 2, 3, ..., \delta$, make a Latin square of order v with $\{S_{lw}\}_{w=1}^{v}$ as the symbol set, where $S_{lw} = \{w, w + l, w - l\}$ and each component is reduced modulo v.

2. Create a $v \times \delta v$ array and divide its columns into δ equal subdivisions of v columns each, separated by double vertical lines.

3. Insert the Latin squares made in step 1 into the array, one in each subdivision.

   Remark. The insertion of the Latin squares into the array does not need to follow a definite order, that is, any of the Latin squares can be inserted in any subdivision of the array.

   The constructed design has as its QBD a $(v, v^2\delta, 3v\delta, 3, 3v)$-BIBD. Hence it is a BSLR.

   For row 1 of the design, $S_{l1} = \{1, 1 + l, 1 - l\}$ gives the entries of the cell (block) in column 1 of the lth subdivision, where $l = 1, 2, 3, ..., \delta$; and in general, for this row, $S_{lw} = \{w, w + l, w - l\}$ gives the entries of the cell in column w of the lth subdivision, where $w = 1, 2, ..., v$. Notice that S_{lw} is identical to $S_{l1} + (w - 1)$, where $w > 1$, that is, for instance, $S_{l2} = S_{l1} + 1$, $S_{l3} = S_{l1} + 2 = (S_{l1} + 1) + 1 = S_{l2} + 1$, $S_{l4} = S_{l1} + 3 = (S_{l1} + 2) + 1 = S_{l3} + 1$, and so on. Hence the blocks in each subdivision can be generated by a cyclic development of its initial block.

Example 4.3.1. Let $v = 5$, then $\delta = 2$. Hence the design is of size $(5 \times 10)/3$: see Figure 4.1.

1 2 5	2 3 1	3 4 2	4 5 3	5 1 4	1 3 4	2 4 5	3 5 1	4 1 2	5 2 3
5 1 4	1 2 5	2 3 1	3 4 2	4 5 3	5 2 3	1 3 4	2 4 5	3 5 1	4 1 2
4 5 3	5 1 4	1 2 5	2 3 1	3 4 2	4 1 2	5 2 3	1 3 4	2 4 5	3 5 1
3 4 2	4 5 3	5 1 4	1 2 5	2 3 1	3 5 1	4 1 2	5 2 3	1 3 4	2 4 5
2 3 1	3 4 2	4 5 3	5 1 4	1 2 5	2 4 5	3 5 1	4 1 2	5 2 3	1 3 4

Figure 4.1: A $(5 \times 10)/3$ balanced semi-Latin rectangle for 5 treatments

Example 4.3.2. Let $v = 7$, then $\delta = 3$. Hence the design is of size $(7 \times 21)/3$: see Figure 4.2.

Example 4.3.3. Let $v = 9$, then $\delta = 4$. Hence the design is of size $(9 \times 36)/3$: see Figure 4.3.

1	2	3	4	5	6	7	1	2	3	4	5	6	7	1	2	3	4	5	6	7
2	3	4	5	6	7	1	3	4	5	6	7	1	2	4	5	6	7	1	2	3
7	1	2	3	4	5	6	6	7	1	2	3	4	5	5	6	7	1	2	3	4
7	1	2	3	4	5	6	7	1	2	3	4	5	6	7	1	2	3	4	5	6
1	2	3	4	5	6	7	2	3	4	5	6	7	1	3	4	5	6	7	1	2
6	7	1	2	3	4	5	5	6	7	1	2	3	4	4	5	6	7	1	2	3
6	7	1	2	3	4	5	6	7	1	2	3	4	5	6	7	1	2	3	4	5
7	1	2	3	4	5	6	1	2	3	4	5	6	7	2	3	4	5	6	7	1
5	6	7	1	2	3	4	4	5	6	7	1	2	3	3	4	5	6	7	1	2
5	6	7	1	2	3	4	5	6	7	1	2	3	4	5	6	7	1	2	3	4
6	7	1	2	3	4	5	7	1	2	3	4	5	6	1	2	3	4	5	6	7
4	5	6	7	1	2	3	3	4	5	6	7	1	2	2	3	4	5	6	7	1
4	5	6	7	1	2	3	4	5	6	7	1	2	3	4	5	6	7	1	2	3
5	6	7	1	2	3	4	6	7	1	2	3	4	5	7	1	2	3	4	5	6
3	4	5	6	7	1	2	2	3	4	5	6	7	1	1	2	3	4	5	6	7
3	4	5	6	7	1	2	3	4	5	6	7	1	2	3	4	5	6	7	1	2
4	5	6	7	1	2	3	5	6	7	1	2	3	4	6	7	1	2	3	4	5
2	3	4	5	6	7	1	1	2	3	4	5	6	7	7	1	2	3	4	5	6
2	3	4	5	6	7	1	2	3	4	5	6	7	1	2	3	4	5	6	7	1
3	4	5	6	7	1	2	4	5	6	7	1	2	3	5	6	7	1	2	3	4
1	2	3	4	5	6	7	7	1	2	3	4	5	6	6	7	1	2	3	4	5

Figure 4.2: A $(7 \times 21)/3$ balanced semi-Latin rectangle for 7 treatments

4.3.2 Designs of the class $(\delta v \times v)/3$

Given the algorithmic procedure in section 4.3.1, where there are v treatments in v rows and δv columns. Let the array size in step 2 be $\delta v \times v$ such that the rows are divided into δ equal subdivisions of v rows each, separated by double horizontal lines. Then this modification produces the desired design, which is a BSLR for v treatments in $v\delta$ rows and v columns.

For instance, in Example 4.3.2, where $v = 7$, and $\delta = 3$. Putting the Latin squares (the order of doing this being immaterial) in a (21×7) array which is partitioned into 3 subdivisions with respect to the rows and separated by double horizontal lines, where each subdivision has 7 rows produces a BSLR for 7 treatments in 21 rows and 7 columns: see Figure 4.4.

Remark. Another design of the same size can be obtained by a transposition of the design in Figure 4.2. The design in Figure 4.4 and that obtainable by transposing the design in Figure 4.2 are isomorphic.

1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9
2	3	4	5	6	7	8	9	1	3	4	5	6	7	8	9	1	2	4	5	6	7	8	9	1	2	3	5	6	7	8	9	1	2	3	4
9	1	2	3	4	5	6	7	8	8	9	1	2	3	4	5	6	7	7	8	9	1	2	3	4	5	6	6	7	8	9	1	2	3	4	5
9	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8
1	2	3	4	5	6	7	8	9	2	3	4	5	6	7	8	9	1	3	4	5	6	7	8	9	1	2	4	5	6	7	8	9	1	2	3
8	9	1	2	3	4	5	6	7	7	8	9	1	2	3	4	5	6	6	7	8	9	1	2	3	4	5	5	6	7	8	9	1	2	3	4
8	9	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7
9	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8	9	2	3	4	5	6	7	8	9	1	3	4	5	6	7	8	9	1	2
7	8	9	1	2	3	4	5	6	6	7	8	9	1	2	3	4	5	5	6	7	8	9	1	2	3	4	4	5	6	7	8	9	1	2	3
7	8	9	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6
8	9	1	2	3	4	5	6	7	9	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8	9	2	3	4	5	6	7	8	9	1
6	7	8	9	1	2	3	4	5	5	6	7	8	9	1	2	3	4	4	5	6	7	8	9	1	2	3	3	4	5	6	7	8	9	1	2
6	7	8	9	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	1	2	3	4	5
7	8	9	1	2	3	4	5	6	8	9	1	2	3	4	5	6	7	9	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8	9
5	6	7	8	9	1	2	3	4	4	5	6	7	8	9	1	2	3	3	4	5	6	7	8	9	1	2	2	3	4	5	6	7	8	9	1
5	6	7	8	9	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	1	2	3	4
6	7	8	9	1	2	3	4	5	7	8	9	1	2	3	4	5	6	8	9	1	2	3	4	5	6	7	9	1	2	3	4	5	6	7	8
4	5	6	7	8	9	1	2	3	3	4	5	6	7	8	9	1	2	2	3	4	5	6	7	8	9	1	1	2	3	4	5	6	7	8	9
4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	1	2	3
5	6	7	8	9	1	2	3	4	6	7	8	9	1	2	3	4	5	7	8	9	1	2	3	4	5	6	8	9	1	2	3	4	5	6	7
3	4	5	6	7	8	9	1	2	2	3	4	5	6	7	8	9	1	1	2	3	4	5	6	7	8	9	9	1	2	3	4	5	6	7	8
3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	1	2
4	5	6	7	8	9	1	2	3	5	6	7	8	9	1	2	3	4	6	7	8	9	1	2	3	4	5	7	8	9	1	2	3	4	5	6
2	3	4	5	6	7	8	9	1	1	2	3	4	5	6	7	8	9	9	1	2	3	4	5	6	7	8	8	9	1	2	3	4	5	6	7
2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	1
3	4	5	6	7	8	9	1	2	4	5	6	7	8	9	1	2	3	5	6	7	8	9	1	2	3	4	6	7	8	9	1	2	3	4	5
1	2	3	4	5	6	7	8	9	9	1	2	3	4	5	6	7	8	8	9	1	2	3	4	5	6	7	7	8	9	1	2	3	4	5	6

Figure 4.3: A $(9 \times 36)/3$ balanced semi-Latin rectangle for 9 treatments

4.3.3 More designs

If δ which specifies the number of Latin squares involved in the construction is a nonprime, more possibilities can be obtained by adapting the procedure in section 4.3.1, via adjusting the array size by taking into consideration each pair of factors (a pair of distinct factors used differently when in reversed order) of the number of Latin squares, the order of each Latin square and the total number of blocks in the QBD of the basic design, the design obtained by implementing the procedure in section 4.3.1.

For instance, 4 Latin squares are involved in the construction of the $(9 \times 36)/3$ BSLR in Figure 4.3. Since 4 is a product of 2 and 2, then the 4 Latin squares can be put in an (18×18) array, thereby obtaining another possibility of a BSLR for 9 treatments whose QBD has 324 blocks just like the basic design in Figure 4.3. The resulting $(18 \times 18)/3$ BSLR is shown in Figure 4.5.

1 2 7	2 3 1	3 4 2	4 5 3	5 6 4	6 7 5	7 1 6
7 1 6	1 2 7	2 3 1	3 4 2	4 5 3	5 6 4	6 7 5
6 7 5	7 1 6	1 2 7	2 3 1	3 4 2	4 5 3	5 6 4
5 6 4	6 7 5	7 1 6	1 2 7	2 3 1	3 4 2	4 5 3
4 5 3	5 6 4	6 7 5	7 1 6	1 2 7	2 3 1	3 4 2
3 4 2	4 5 3	5 6 4	6 7 5	7 1 6	1 2 7	2 3 1
2 3 1	3 4 2	4 5 3	5 6 4	6 7 5	7 1 6	1 2 7
1 3 6	2 4 7	3 5 1	4 6 2	5 7 3	6 1 4	7 2 5
7 2 5	1 3 6	2 4 7	3 5 1	4 6 2	5 7 3	6 1 4
6 1 4	7 2 5	1 3 6	2 4 7	3 5 1	4 6 2	5 7 3
5 7 3	6 1 4	7 2 5	1 3 6	2 4 7	3 5 1	4 6 2
4 6 2	5 7 3	6 1 4	7 2 5	1 3 6	2 4 7	3 5 1
3 5 1	4 6 2	5 7 3	6 1 4	7 2 5	1 3 6	2 4 7
2 4 7	3 5 1	4 6 2	5 7 3	6 1 4	7 2 5	1 3 6
1 4 5	2 5 6	3 6 7	4 7 1	5 1 2	6 2 3	7 3 4
7 3 4	1 4 5	2 5 6	3 6 7	4 7 1	5 1 2	6 2 3
6 2 3	7 3 4	1 4 5	2 5 6	3 6 7	4 7 1	5 1 2
5 1 2	6 2 3	7 3 4	1 4 5	2 5 6	3 6 7	4 7 1
4 7 1	5 1 2	6 2 3	7 3 4	1 4 5	2 5 6	3 6 7
3 6 7	4 7 1	5 1 2	6 2 3	7 3 4	1 4 5	2 5 6
2 5 6	3 6 7	4 7 1	5 1 2	6 2 3	7 3 4	1 4 5

Figure 4.4: A $(21 \times 7)/3$ balanced semi-Latin rectangle for 7 treatments

1 2 9	2 3 1	3 4 2	4 5 3	5 6 4	6 7 5	7 8 6	8 9 7	9 1 8	1 3 8	2 4 9	3 5 1	4 6 2	5 7 3	6 8 4	7 9 5	8 1 6	9 2 7
9 1 8	1 2 9	2 3 1	3 4 2	4 5 3	5 6 4	6 7 5	7 8 6	8 9 7	9 2 7	1 3 8	2 4 9	3 5 1	4 6 2	5 7 3	6 8 4	7 9 5	8 1 6
8 9 7	9 1 8	1 2 9	2 3 1	3 4 2	4 5 3	5 6 4	6 7 5	7 8 6	8 1 6	9 2 7	1 3 8	2 4 9	3 5 1	4 6 2	5 7 3	6 8 4	7 9 5
7 8 6	8 9 7	9 1 8	1 2 9	2 3 1	3 4 2	4 5 3	5 6 4	6 7 5	7 9 5	8 1 6	9 2 7	1 3 8	2 4 9	3 5 1	4 6 2	5 7 3	6 8 4
6 7 5	7 8 6	8 9 7	9 1 8	1 2 9	2 3 1	3 4 2	4 5 3	5 6 4	6 8 4	7 9 5	8 1 6	9 2 7	1 3 8	2 4 9	3 5 1	4 6 2	5 7 3
5 6 4	6 7 5	7 8 6	8 9 7	9 1 8	1 2 9	2 3 1	3 4 2	4 5 3	5 7 3	6 8 4	7 9 5	8 1 6	9 2 7	1 3 8	2 4 9	3 5 1	4 6 2
4 5 3	5 6 4	6 7 5	7 8 6	8 9 7	9 1 8	1 2 9	2 3 1	3 4 2	4 6 2	5 7 3	6 8 4	7 9 5	8 1 6	9 2 7	1 3 8	2 4 9	3 5 1
3 4 2	4 5 3	5 6 4	6 7 5	7 8 6	8 9 7	9 1 8	1 2 9	2 3 1	3 5 1	4 6 2	5 7 3	6 8 4	7 9 5	8 1 6	9 2 7	1 3 8	2 4 9
2 3 1	3 4 2	4 5 3	5 6 4	6 7 5	7 8 6	8 9 7	9 1 8	1 2 9	2 4 9	3 5 1	4 6 2	5 7 3	6 8 4	7 9 5	8 1 6	9 2 7	1 3 8
1 4 7	2 5 8	3 6 9	4 7 1	5 8 2	6 9 3	7 1 4	8 2 5	9 3 6	1 5 6	2 6 7	3 7 8	4 8 9	5 9 1	6 1 2	7 2 3	8 3 4	9 4 5
9 3 6	1 4 7	2 5 8	3 6 9	4 7 1	5 8 2	6 9 3	7 1 4	8 2 5	9 4 5	1 5 6	2 6 7	3 7 8	4 8 9	5 9 1	6 1 2	7 2 3	8 3 4
8 2 5	9 3 6	1 4 7	2 5 8	3 6 9	4 7 1	5 8 2	6 9 3	7 1 4	8 3 4	9 4 5	1 5 6	2 6 7	3 7 8	4 8 9	5 9 1	6 1 2	7 2 3
7 1 4	8 2 5	9 3 6	1 4 7	2 5 8	3 6 9	4 7 1	5 8 2	6 9 3	7 2 3	8 3 4	9 4 5	1 5 6	2 6 7	3 7 8	4 8 9	5 9 1	6 1 2
6 9 3	7 1 4	8 2 5	9 3 6	1 4 7	2 5 8	3 6 9	4 7 1	5 8 2	6 1 2	7 2 3	8 3 4	9 4 5	1 5 6	2 6 7	3 7 8	4 8 9	5 9 1
5 8 2	6 9 3	7 1 4	8 2 5	9 3 6	1 4 7	2 5 8	3 6 9	4 7 1	5 9 1	6 1 2	7 2 3	8 3 4	9 4 5	1 5 6	2 6 7	3 7 8	4 8 9
4 7 1	5 8 2	6 9 3	7 1 4	8 2 5	9 3 6	1 4 7	2 5 8	3 6 9	4 8 9	5 9 1	6 1 2	7 2 3	8 3 4	9 4 5	1 5 6	2 6 7	3 7 8
3 6 9	4 7 1	5 8 2	6 9 3	7 1 4	8 2 5	9 3 6	1 4 7	2 5 8	3 7 8	4 8 9	5 9 1	6 1 2	7 2 3	8 3 4	9 4 5	1 5 6	2 6 7
2 5 8	3 6 9	4 7 1	5 8 2	6 9 3	7 1 4	8 2 5	9 3 6	1 4 7	2 6 7	3 7 8	4 8 9	5 9 1	6 1 2	7 2 3	8 3 4	9 4 5	1 5 6

Figure 4.5: A $(18 \times 18)/3$ balanced semi-Latin rectangle for 9 treatments

4.4 Constructions based on difference sets/difference families

4.4.1 Preliminaries

Let $G = (\mathbb{Z}_v, +)$ denote a group formed by Z_v, the set of integers (mod v). We had previously defined a (v, k, λ)-difference set for $\mathbb{Z}_v$ to be a k–element subset of $\mathbb{Z}_v$ for which the differences between all possible pairs of its elements give all the non-zero elements of $\mathbb{Z}_v$, each of them exactly λ times : see Definition 2.2.16, see also Stinson (2004, Chapter 3). We note that a (v, k, λ)-difference set for $\mathbb{Z}_v$ does not always exist. If it exists, then $\frac{\lambda(v-1)}{k(k-1)} = 1$, where k is the size of the difference set and satisfies $2 \leq k < v$, λ (> 0) is the index. Hence, if a difference set exists, it generates a symmetric (v, k, λ)-BIBD, denoted (v, k, λ)-SBIBD via a cyclic development of the difference set which involves successive addition of an element of Z_v.

Furthermore, a difference family (which has more than one set) generalizes the concept of difference set. Let $\beta = \frac{\lambda(v-1)}{k(k-1)}$. If a (v, k, λ)-difference family for $\mathbb{Z}_v$ exists, then $\beta > 1$, where $\beta \in \mathbb{Z}$ specifies the number of sets in the family: see, for example, Stinson (2004, Chapter 3), for discussions and examples of difference sets and difference families.

Notice that, for a difference set, $\beta = 1$.

Example 4.4.1. The set $\{1, 2, 4\}$ is a (7, 3, 1)-difference set in $(\mathbb{Z}_7, +)$, thus it generates a (7, 3, 1)-SBIBD. Notice that $\beta = 1$ and the differences (modulo 7) between all possible pairs of elements of the set yield ±1,±2 and ±3, that is, elements of the set $\{1, 2, 3, 4, 5, 6\}$, which is precisely the set $\mathbb{Z}_7 \backslash \{0\}$, where each element appears exactly once.

Similarly, $\{1, 2, 6, 12\}$ is a (13,4,1)-difference set in $(\mathbb{Z}_{13}, +)$, thus it generates a (13, 4, 1)-SBIBD. Some other examples include: $\{1, 2, 3\}$, which is a (4, 3, 2)-difference set in $(\mathbb{Z}_4, +)$; $\{1, 2, 7, 9, 19\}$, which is a (21, 5, 1)-difference set in $(\mathbb{Z}_{21}, +)$; $\{1, 3, 4, 5, 9\}$ which is an (11, 5, 2)-difference set in $(\mathbb{Z}_{11}, +)$; $\{1, 2, 3, 5, 6, 9, 11\}$ which is a (15,7,3)-difference set in $(\mathbb{Z}_{15}, +)$; and $\{1, 7, 9, 10, 12, 16, 26, 33, 34\}$, a (37,9,2)-difference set in $(\mathbb{Z}_{37}, +)$.

Furthermore, the sets $\{1, 2, 5\}$ and $\{2, 4, 10\}$ constitute a (13, 3, 1)-difference family in $(\mathbb{Z}_{13}, +)$, and this generates a (13, 3, 1)-BIBD. Notice that $\beta = 2$ and the differences between all possible pairs of elements (modulo 13) with respect to the 1st set give: ±1,±3 and ±4, which is the set $\{1, 3, 4, 9, 10, 12\}$. For the 2nd set, the differences are ±2,±6 and ±8 which constitute the set $\{2, 5, 6, 7, 8, 11\}$. Let $A = \{1, 3, 4, 9, 10, 12\}$ and $B = \{2, 5, 6, 7, 8, 11\}$. Then, it is obvious that $A \cup B = \mathbb{Z}_{13} \backslash \{0\}$, each element of the set appearing exactly once.

Also, $\{\{1, 2, 4\}, \{3, 5, 6\}\}$ is a (7, 3, 2)-difference family in $(\mathbb{Z}_7, +)$, and thus generates a (7, 3, 2)-BIBD; $\{\{1, 2, 4, 25\}, \{1, 11, 19, 31\}, \{1, 5, 27, 33\}\}$ is a (37, 4, 1)-difference family in $(\mathbb{Z}_{37}, +)$, and thus generates a (37,4, 1)-BIBD; and $\{\{1, 3, 5\}, \{2, 6, 3\}, \{3, 2, 1\}, \{4, 2, 1\}\}$ is a (7, 3, 4)-difference family in $(\mathbb{Z}_7, +)$, hence generates a (7, 3, 4)-BIBD.

Definition 4.4.1. Let S denote a (v, k, λ)-difference set in $(\mathbb{Z}_v, +)$. Then for all $j \in \mathbb{Z}_v$,

$S + j = \{i + j : i \in S\}$ is said to be a translate of S.

Remark. We regard the set of integers modulo v as $\{1, 2, , ..., v\}$. Notice that if $j = v$, then $S + j = S$, which makes S a translate of itself. Furthermore, the set of all v translates of S gives the block set of a symmetric (v, k, λ)-BIBD.

4.4.2 Construction Procedure

Given the treatment set, $\mathcal{V} = \{1, 2, ..., v\}$ of a design. Suppose a (v, k, λ)-difference set (or difference family) exists in $(\mathbb{Z}_v, +)$. Then we begin by utilizing the difference set (or difference family) to obtain a (v, k, λ)-BIBD *via* successive addition of 1 to each element of the set(s), reduced mod v, and subsequently, utilizing the blocks of the BIBD in our construction of BSLR.

In general, the number of blocks generated for the BIBD is βv, where $\beta v \geq v$. In particular, with a difference set, v distinct blocks are generated. Similarly, if a difference family exists, each set in the family generates v blocks, and overall, βv $(> v)$ blocks are generated.

Then, with the BIBD, some BSLRs of appropriate sizes can be constructed.

Let S denote a (v, k, λ)-difference set and S_j, its jth translate, where $j = 1, 2, ..., v$. For all $j \in \{1, 2, ..., v\}$, define $S_j = S + (j - 1)$, where $S + (j - 1) = \{m + (j - 1) : m \in S\}$. Notice that there are v translates of S, viz, S_1,S_2,...,S_v, where $S_1 = S$, $S_2 = S + 1$,...,$S_v = S + (v - 1)$. Similarly, let A_y denote the yth member set of a (v, k, λ)-difference family, where $y = 1, 2, ..., \beta$. Let A_{yj} denote the jth translate of A_y, where $j = 1, 2, ..., v$. For all $y \in \{1, 2, ..., \beta\}$ and $j \in \{1, 2, ..., v\}$, define $A_{yj} = A_y + (j - 1)$, where $A_y + (j - 1) = \{n + (j - 1) : n \in A_y\}$. Notice that, for all y, there are v translates of A_y, which are A_{y1}, A_{y2},..., A_{yv}, where $A_{y1} = A_y$, $A_{y2} = A_y + 1$,..., $A_{yv} = A_y + (v - 1)$. In particular, $A_{11} = A_1$, $A_{21} = A_2$,..., $A_{\beta 1} = A_\beta$.

In the case that a difference set exists, the v translates of S, that is, S_1,S_2,...,S_v are then used as symbols to make a Latin square of order v. Similarly, if a difference family exists, then for all y, A_{y1}, A_{y2},..., A_{yv} are used as symbols to make a Latin square. These Latin squares are then inserted into an array of appropriate size to obtain the desired design.

4.4.3 Construction for designs of the class $(v \times \beta v)/k$

A $(v \times \beta v)/k$ array is created and the columns are divided into β subdivisions, each subdivision having v columns and separated by double vertical lines. Subsequently, each Latin square is inserted into a subdivision to give the desired design.

An algorithmic procedure for the construction

1. Identify a (v, k, λ)-difference set (or difference family) in $(\mathbb{Z}_v, +)$, if it exists

2. If a difference set exists, for all $j \in \{1, 2, ..., v\}$, put $S_j = S+(j-1)$, where $S+(j-1) = \{m + (j-1) : m \in S\}$. But if a difference family exists, for all $y \in \{1, 2, ..., \beta\}$ and $j \in \{1, 2, ..., v\}$, put $A_{yj} = A_y + (j-1)$, where $A_y + (j-1) = \{n + (j-1) : n \in A_y\}$, and $\beta = \frac{\lambda(v-1)}{k(k-1)}$.

3. In the case where a difference set exists, make a Latin square of order v using S_1,S_2,...,S_v as symbols; and in the case where a difference family exists, for all $y \in \{1, 2, ..., \beta\}$, make a Latin square of order v with A_{y1}, A_{y2},..., A_{yv} as symbols.

4. Create a $(v \times \beta v)$ array and divide its columns into β equal subdivisions of v columns each, separated by double vertical lines.

5. Insert the Latin squares made in step 3 into the array, one in each subdivision.

Remark. In general, β Latin squares are required for the construction. Just like in section 4.3.1, any of the Latin squares can be inserted in any subdivision of the array. Notice that, if a difference set is used for the construction, then $\beta = 1$ such that $\beta v = v$. Hence, the number of columns in the array of the design is identical to the number of rows, which is clearly, a design with the same number of rows as columns.

Example 4.4.2. Let $v = 7$. For $k = 3$, we recognize that there exists a $(7, 3, 1)$-difference set in $(\mathbb{Z}_7, +)$, which is $\{1, 2, 4\}$: see Example 4.4.1. Notice that $\beta = 1$, hence $\beta v = 7$. From this set of parameters, we obtain a $(7 \times 7)/3$ BSLR for 7 treatments. A direct implementation of the algorithmic procedure produces the desired design: see Figure 4.6.

1 2 4	2 3 5	3 4 6	4 5 7	5 6 1	6 7 2	7 1 3
7 1 3	1 2 4	2 3 5	3 4 6	4 5 7	5 6 1	6 7 2
6 7 2	7 1 3	1 2 4	2 3 5	3 4 6	4 5 7	5 6 1
5 6 1	6 7 2	7 1 3	1 2 4	2 3 5	3 4 6	4 5 7
4 5 7	5 6 1	6 7 2	7 1 3	1 2 4	2 3 5	3 4 6
3 4 6	4 5 7	5 6 1	6 7 2	7 1 3	1 2 4	2 3 5
2 3 5	3 4 6	4 5 7	5 6 1	6 7 2	7 1 3	1 2 4

Figure 4.6: A $(7 \times 7)/3$ balanced semi-Latin rectangle for 7 treatments

Example 4.4.3. Let $v = 13$. we utilize the set $\{1, 2, 6, 12\}$, which is a (13,4,1)-difference set in $(\mathbb{Z}_{13}, +)$, as given in Example 4.4.1. A direct implementation of the algorithm with $\beta = 1$ leads to the design in Figure 4.7.

Note that, in the construction, we set $a = 10$, $b = 11$, $c = 12$ and $d = 13$, with a reduction (modulo 13) in the addition.

1 2 6 c	2 3 7 d	3 4 8 1	4 5 9 2	5 6 a 3	6 7 b 4	7 8 c 5	8 9 d 6	9 a 1 7	a b 2 8	b c 3 9	c d 4 a	d 1 5 b
d 1 5 b	1 2 6 c	2 3 7 d	3 4 8 1	4 5 9 2	5 6 a 3	6 7 b 4	7 8 c 5	8 9 d 6	9 a 1 7	a b 2 8	b c 3 9	c d 4 a
c d 4 a	d 1 5 b	1 2 6 c	2 3 7 d	3 4 8 1	4 5 9 2	5 6 a 3	6 7 b 4	7 8 c 5	8 9 d 6	9 a 1 7	a b 2 8	b c 3 9
b c 3 9	c d 4 a	d 1 5 b	1 2 6 c	2 3 7 d	3 4 8 1	4 5 9 2	5 6 a 3	6 7 b 4	7 8 c 5	8 9 d 6	9 a 1 7	a b 2 8
a b 2 8	b c 3 9	c d 4 a	d 1 5 b	1 2 6 c	2 3 7 d	3 4 8 1	4 5 9 2	5 6 a 3	6 7 b 4	7 8 c 5	8 9 d 6	9 a 1 7
9 a 1 7	a b 2 8	b c 3 9	c d 4 a	d 1 5 b	1 2 6 c	2 3 7 d	3 4 8 1	4 5 9 2	5 6 a 3	6 7 b 4	7 8 c 5	8 9 d 6
8 9 d 6	9 a 1 7	a b 2 8	b c 3 9	c d 4 a	d 1 5 b	1 2 6 c	2 3 7 d	3 4 8 1	4 5 9 2	5 6 a 3	6 7 b 4	7 8 c 5
7 8 c 5	8 9 d 6	9 a 1 7	a b 2 8	b c 3 9	c d 4 a	d 1 5 b	1 2 6 c	2 3 7 d	3 4 8 1	4 5 9 2	5 6 a 3	6 7 b 4
6 7 b 4	7 8 c 5	8 9 d 6	9 a 1 7	a b 2 8	b c 3 9	c d 4 a	d 1 5 b	1 2 6 c	2 3 7 d	3 4 8 1	4 5 9 2	5 6 a 3
5 6 a 3	6 7 b 4	7 8 c 5	8 9 d 6	9 a 1 7	a b 2 8	b c 3 9	c d 4 a	d 1 5 b	1 2 6 c	2 3 7 d	3 4 8 1	4 5 9 2
4 5 9 2	5 6 a 3	6 7 b 4	7 8 c 5	8 9 d 6	9 a 1 7	a b 2 8	b c 3 9	c d 4 a	d 1 5 b	1 2 6 c	2 3 7 d	3 4 8 1
3 4 8 1	4 5 9 2	5 6 a 3	6 7 b 4	7 8 c 5	8 9 d 6	9 a 1 7	a b 2 8	b c 3 9	c d 4 a	d 1 5 b	1 2 6 c	2 3 7 d
2 3 7 d	3 4 8 1	4 5 9 2	5 6 a 3	6 7 b 4	7 8 c 5	8 9 d 6	9 a 1 7	a b 2 8	b c 3 9	c d 4 a	d 1 5 b	1 2 6 c

Figure 4.7: A $(13 \times 13)/4$ balanced semi-Latin rectangle for 13 treatments

Example 4.4.4. Let $v = 13$. To obtain, for instance, a $(13 \times 26)/3$ BSLR for 13 treatments, we recognize that there exists a $(13, 3, 1)$-difference family in $(\mathbb{Z}_{13}, +)$ (formed by the sets $\{1, 2, 5\}$ and $\{2, 4, 10\}$) as given in Example 4.4.1.

Notice that, $\beta = 2$, hence $\beta v = 26$ and 2 Latin squares are to be used in the construction. In constructing the design, we put the treatment labels a, b, c and d for 10, 11, 12 and 13, respectively, with a reduction (modulo 13) in the addition. By implementing the algorithm, the desired design is obtained and presented in Figure 4.8.

Example 4.4.5. Let $v = 7$. To obtain, for instance, a $(7 \times 28)/3$ BSLR for 7 treatments, we utilize the $(7, 3, 4)$-difference family in $(\mathbb{Z}_7, +)$ given in Example 4.4.1, where the component sets are $\{1, 3, 5\}$, $\{2, 6, 3\}$, $\{3, 2, 1\}$, and $\{4, 2, 1\}$.

Notice that, in this example, $\beta = 4$, hence $\beta v = 28$ and 4 Latin squares are required for the construction. The design produced via a direct implementation of the algorithm is shown in Figure 4.9.

Remark. Any existing difference set or difference family can be used in a similar manner to obtain the corresponding design.

4.4.4 Designs of the class $(\beta v \times v)/k$

Given the algorithmic procedure in section 4.4.3, if step 4. is modified to read "create a $(\beta v \times v)/k$ array and divide the rows into β subdivisions of equal size, separated by double horizontal lines.", this leads to another class of BSLRs with v treatments, having βv rows and v columns.

125	236	347	458	569	67a	78b	89c	9ad	ab1	bc2	cd3	d14	24a	35b	46c	57d	681	792	8a3	9b4	ac5	bd6	c17	d28	139
d14	125	236	347	458	569	67a	78b	89c	9ad	ab1	bc2	cd3	139	24a	35b	46c	57d	681	792	8a3	9b4	ac5	bd6	c17	d28
cd3	d14	125	236	347	458	569	67a	78b	89c	9ad	ab1	bc2	d28	139	24a	35b	46c	57d	681	792	8a3	9b4	ac5	bd6	c17
bc2	cd3	d14	125	236	347	458	569	67a	78b	89c	9ad	ab1	c17	d28	139	24a	35b	46c	57d	681	792	8a3	9b4	ac5	bd6
ab1	bc2	cd3	d14	125	236	347	458	569	67a	78b	89c	9ad	bd6	c17	d28	139	24a	35b	46c	57d	681	792	8a3	9b4	ac5
9ad	ab1	bc2	cd3	d14	125	236	347	458	569	67a	78b	89c	ac5	bd6	c17	d28	139	24a	35b	46c	57d	681	792	8a3	9b4
89c	9ad	ab1	bc2	cd3	d14	125	236	347	458	569	67a	78b	9b4	ac5	bd6	c17	d28	139	24a	35b	46c	57d	681	792	8a3
78b	89c	9ad	ab1	bc2	cd3	d14	125	236	347	458	569	67a	8a3	9b4	ac5	bd6	c17	d28	139	24a	35b	46c	57d	681	792
67a	78b	89c	9ad	ab1	bc2	cd3	d14	125	236	347	458	569	792	8a3	9b4	ac5	bd6	c17	d28	139	24a	35b	46c	57d	681
569	67a	78b	89c	9ad	ab1	bc2	cd3	d14	125	236	347	458	681	792	8a3	9b4	ac5	bd6	c17	d28	139	24a	35b	46c	57d
458	569	67a	78b	89c	9ad	ab1	bc2	cd3	d14	125	236	347	57d	681	792	8a3	9b4	ac5	bd6	c17	d28	139	24a	35b	46c
347	458	569	67a	78b	89c	9ad	ab1	bc2	cd3	d14	125	236	46c	57d	681	792	8a3	9b4	ac5	bd6	c17	d28	139	24a	35b
236	347	458	569	67a	78b	89c	9ad	ab1	bc2	cd3	d14	125	35b	46c	57d	681	792	8a3	9b4	ac5	bd6	c17	d28	139	24a

Figure 4.8: A $(13 \times 26)/3$ balanced semi-Latin rectangle for 13 treatments

As an illustration, if a (26×13) array is created and the 2 Latin squares used in constructing the design in Figure 4.8 are inserted, then the resulting design is a $(26 \times 13)/3$ BSLR

Remark. Transposing a corresponding $(v \times \beta v)/k$ BSLR leads to a BSLR of the same size as the one in this section.

4.4.5 More designs from the constructions

Suppose β, the number of Latin squares used in the preceding constructions is a nonprime. Then, just like in section 4.3.3, the array size can be adjusted to obtain some more designs of appropriate sizes, whose rows and columns are multiples of v.

For instance, in constructing the $(7 \times 28)/3$ balanced semi-Latin rectangle for 7 treatments in Figure 4.9, 4 Latin squares (each of order 7) were used. Since 4 is a nonprime, then the 4 Latin squares can be put in a (14×14) array to obtain a $(14 \times 14)/3$ BSLR for 7 treatments.

Remark. The number of blocks in the QBD of the new design is invariant, that is, it has precisely identical number of blocks as the basic design as well as the alternative basic design.

1 3 5	2 4 6	3 5 7	4 6 1	5 7 2	6 1 3	7 2 4	2 6 3	3 7 4	4 1 5	5 2 6	6 3 7	7 4 1	1 5 2	3 2 1	4 3 2	5 4 3	6 5 4	7 6 5	1 7 6	2 1 7	4 2 1	5 3 2	6 4 3	7 5 4	1 6 5	2 7 6	3 1 7
7 2 4	1 3 5	2 4 6	3 5 7	4 6 1	5 7 2	6 1 3	1 5 2	2 6 3	3 7 4	4 1 5	5 2 6	6 3 7	7 4 1	2 1 7	3 2 1	4 3 2	5 4 3	6 5 4	7 6 5	1 7 6	3 1 7	4 2 1	5 3 2	6 4 3	7 5 4	1 6 5	2 7 6
6 1 3	7 2 4	1 3 5	2 4 6	3 5 7	4 6 1	5 7 2	7 4 1	1 5 2	2 6 3	3 7 4	4 1 5	5 2 6	6 3 7	1 7 6	2 1 7	3 2 1	4 3 2	5 4 3	6 5 4	7 6 5	2 7 6	3 1 7	4 2 1	5 3 2	6 4 3	7 5 4	1 6 5
5 7 2	6 1 3	7 2 4	1 3 5	2 4 6	3 5 7	4 6 1	6 3 7	7 4 1	1 5 2	2 6 3	3 7 4	4 1 5	5 2 6	7 6 5	1 7 6	2 1 7	3 2 1	4 3 2	5 4 3	6 5 4	1 6 5	2 7 6	3 1 7	4 2 1	5 3 2	6 4 3	7 5 4
4 6 1	5 7 2	6 1 3	7 2 4	1 3 5	2 4 6	3 5 7	5 2 6	6 3 7	7 4 1	1 5 2	2 6 3	3 7 4	4 1 5	6 5 4	7 6 5	1 7 6	2 1 7	3 2 1	4 3 2	5 4 3	7 5 4	1 6 5	2 7 6	3 1 7	4 2 1	5 3 2	6 4 3
3 5 7	4 6 1	5 7 2	6 1 3	7 2 4	1 3 5	2 4 6	4 1 5	5 2 6	6 3 7	7 4 1	1 5 2	2 6 3	3 7 4	5 4 3	6 5 4	7 6 5	1 7 6	2 1 7	3 2 1	4 3 2	6 4 3	7 5 4	1 6 5	2 7 6	3 1 7	4 2 1	5 3 2
2 4 6	3 5 7	4 6 1	5 7 2	6 1 3	7 2 4	1 3 5	3 7 4	4 1 5	5 2 6	6 3 7	7 4 1	1 5 2	2 6 3	4 3 2	5 4 3	6 5 4	7 6 5	1 7 6	2 1 7	3 2 1	5 3 2	6 4 3	7 5 4	1 6 5	2 7 6	3 1 7	4 2 1

Figure 4.9: A $(7 \times 28)/3$ balanced semi-Latin rectangle for 7 treatments

4.5 Constructions based on complete sets of mutually orthogonal Latin squares (MOLSs)

4.5.1 Preliminaries

Definition 4.5.1. Let Λ and Δ denote two Latin squares of the same order, n, say. Then Λ and Δ are said to be mutually orthogonal if when Λ (or Δ) is superimposed on Δ (or Λ), the n^2 cells of the resulting array consists entirely of every ordered pair of symbols of Λ and Δ, each appearing in exactly one cell.

Definition 4.5.2. Let $A = \{L_i\}_{i=1}^{t}$ denote a set of t Latin squares of the same finite order, m, say. Then the L_is are said to constitute mutually orthogonal Latin squares if for all $u, w \in \{1, 2, ..., t\}$, L_u and L_w are orthogonal, where $u \neq w$.

In particular, $t \leq m - 1$. Furthermore, if m is a power of a prime, then there exists a set of $m - 1$ mutually orthogonal Latin squares.

Remark. A is said to be a complete set of mutually orthogonal Latin squares if $t = m - 1$.

A complete set of MOLSs can be used in conjunction with a square array containing treatment symbols to obtain an affine resolvable BIBD, and this design is also called balanced square lattice design: see, for example, John and Williams (1995, Chapters 1 & 4), as well as Raghavarao and Padgett (2005, Chapters 4 & 9). See also Street and

Street (1987, Chapter 8) for discussion on affine resolvability of a BIBD. Its definition is also contained in definition 2.2.8 in Chapter 2 of this work. This is then utilized in our construction of BSLRs.

Besides having each treatment appearing in exactly one block of each superblock (replicate or resolution class) of a BIBD, hence no two blocks containing any treatment in common which is basic to all resolvable designs, affine resolvable designs possess an additional property that any two blocks from distinct resolution classes contain an equal number, b_s of treatments in common. This is consistent with our notation in definition 2.2.8. Given an affine resolvable (v, k, λ)-BIBD with b^* blocks in each resolution class, then $b_s = k/b^*$, where $b^* = v/k$. Consequently, b_s reduces to k^2/v: see, for example, Street and Street (1987, Chapter 8) and Raghavarao and Padgett (2005, Chapter 4).

4.5.2 Construction procedure

Let $\mathcal{V} = \{1, 2, ..., v\}$ denote a set of treatments, where $v = g^2$, $g = p^x$, p being a prime, and $x \in \mathbb{Z}$, $x \geq 1$.

The g^2 treatments are arranged in a $g \times g$ array as given in Figure 4.10

1	2	$\cdots$	g
$g+1$	$g+2$	$\cdots$	$2g$
$2g+1$	$2g+2$	$\cdots$	$3g$
$3g+1$	$3g+2$	$\cdots$	$4g$
$\vdots$	$\vdots$	$\cdots$	$\vdots$
$(g-2)g+1$	$(g-2)g+2$	$\cdots$	$g(g-1)$
$(g-1)g+1$	$(g-1)g+2$	$\cdots$	g^2

Figure 4.10: An arrangement of the g^2 treatments in a $(g \times g)$ array

A complete set, $g-1$ MOLSs, each of order g is then used in conjunction with the g^2 treatments in the array to obtain a $(g^2, g(g+1), g+1, g, 1)$-BIBD which is affine resolvable via grouping those treatments that appear together in each row, and also those in each column of the array to form two resolution classes. The remaining $g-1$ classes are obtained by using each of the $g-1$ orthogonal Latin squares-grouping those treatments which are in correspondence with each symbol of the Latin square with respect to position. Notice that the BIBD has $g+1$ resolution classes, where each class has g blocks, each being of size g.

Now, for all $i \in \{1, 2, ..., g+1\}$, let B_{ij} denote the jth block in the ith resolution class of the BIBD, where $j = 1, 2, ..., g$. Then for each i, make a Latin square of order g using B_{i1}, B_{i2}, . . .,B_{ig} as symbols; and subsequently, insert the Latin squares into an array of appropriate size to obtain the desired design.

4.5.3 Construction for designs of the class $(g \times g(g+1))/g$

We create a $g \times g(g+1)$ array and divide its columns into $g+1$ subdivisions, each subdivision having g columns and separated by double vertical lines. The Latin squares are then inserted into the various subdivision-one Latin square in each subdivision to give the design.

An algorithmic procedure for the construction

1. For all $i \in \{1, 2, ..., g+1\}$, obtain B_{ij} as described previously, where $j = 1, 2, ..., g$.
2. For each i, make a Latin square, L_i of order g using B_{i1}, B_{i2}, . . .,B_{ig} as symbols
3. Create a $g \times g(g+1)$ array and divide its columns into $g+1$ subdivisions of equal sizes, separated by double vertical lines.
4. Insert the Latin squares made in 2. into the array, one in each subdivision.

Remark. Just like in the previous constructions, the insertion of the Latin squares into the array does not need to follow a definite order-any of the Latin squares can be inserted in any subdivision of the array.

Example 4.5.1. Let $v = 9$ such that $g = 3$. Then an implementation of the algorithmic procedure produces a $(3 \times 12)/3$ BSLR.

The 3×3 array containing the treatments is obtained to be

1	2	3
4	5	6
7	8	9

Let Λ_1 and Λ_2 constitute a complete set of orthogonal Latin squares of order 3 with the symbol set $\{A, B, C\}$, where

$\Lambda_1 =$

A	B	C
C	A	B
B	C	A

and

$$\Lambda_2 = \begin{array}{|c|c|c|} \hline A & B & C \\ \hline B & C & A \\ \hline C & A & B \\ \hline \end{array}$$

We obtain, for instance, $B_{11} = \{1,2,3\}$, $B_{12} = \{4,5,6\}$, $B_{13} = \{7,8,9\}$; $B_{21} = \{1,4,7\}$, $B_{22} = \{2,5,8\}$, $B_{23} = \{3,6,9\}$; $B_{31} = \{1,5,9\}$, $B_{32} = \{2,6,7\}$, $B_{33} = \{3,4,8\}$; and $B_{41} = \{1,6,8\}$, $B_{42} = \{2,4,9\}$, $B_{43} = \{3,5,7\}$, for the 1st, 2nd, 3rd, and 4th resolution classes, respectively.

Suppose we put the Latin squares L_1, L_2, L_3 and L_4 obtained from these in the array in a natural order, then we obtain the design shown in Figure 4.11.

1 2 3	4 5 6	7 8 9	1 4 7	2 5 8	3 6 9	1 5 9	2 6 7	3 4 8	1 6 8	2 4 9	3 5 7
7 8 9	1 2 3	4 5 6	3 6 9	1 4 7	2 5 8	3 4 8	1 5 9	2 6 7	3 5 7	1 6 8	2 4 9
4 5 6	7 8 9	1 2 3	2 5 8	3 6 9	1 4 7	2 6 7	3 4 8	1 5 9	2 4 9	3 5 7	1 6 8

Figure 4.11: A $(3 \times 12)/3$ balanced semi-Latin rectangle for 9 treatments

Example 4.5.2. Suppose $v = 16$. Then $g = 4$; and from the construction, a $(4 \times 20)/4$ BSLR is obtained.

Notice that, for this example, the treatment array is thus

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

Let Δ_1, Δ_2, and Δ_3 form a complete set of MOLSs of order 4 with the symbol set $\{A, B, C, D\}$, where

$$\Delta_1 = \begin{array}{|c|c|c|c|}\hline A & B & C & D \\ \hline B & A & D & C \\ \hline C & D & A & B \\ \hline D & C & B & A \\ \hline\end{array}$$

$$\Delta_2 = \begin{array}{|c|c|c|c|}\hline A & B & C & D \\ \hline C & D & A & B \\ \hline D & C & B & A \\ \hline B & A & D & C \\ \hline\end{array}$$

and

$$\Delta_3 = \begin{array}{|c|c|c|c|}\hline A & B & C & D \\ \hline D & C & B & A \\ \hline B & A & D & C \\ \hline C & D & A & B \\ \hline\end{array}$$

The B_{ij}s are, for instance, $B_{11} = \{1,2,3,4\}$, $B_{12} = \{5,6,7,8\}$, $B_{13} = \{9,10,11,12\}$, $B_{14} = \{13,14,15,16\}$; $B_{21} = \{1,5,9,13\}$, $B_{22} = \{2,6,10,14\}$, $B_{23} = \{3,7,11,15\}$, $B_{24} = \{4,8,12,16\}$; $B_{31} = \{1,6,11,16\}$, $B_{32} = \{2,5,12,15\}$, $B_{33} = \{3,8,9,14\}$, $B_{34} = \{4,7,10,13\}$; $B_{41} = \{1,7,12,14\}$, $B_{42} = \{2,8,11,13\}$, $B_{43} = \{3,5,10,16\}$, $B_{44} = \{4,6,9,15\}$; and $B_{51} = \{1,8,10,15\}$, $B_{52} = \{2,7,9,16\}$, $B_{53} = \{3,6,12,13\}$, $B_{54} = \{4,5,11,14\}$, for the 1st, 2nd, . . ., 5th resolution classes, respectively.

If we put the Latin squares L_1, L_2, L_3, L_4 and L_5 obtained from these in the array in a natural order, we obtain the design in Figure 4.12.

Suppose the Latin squares are inserted in a different order, say, L_3, L_1, L_2, L_5, L_4, then this produces the design in Figure 4.13.

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

Figure 4.12: A $(4\times 20)/4$ balanced semi-Latin rectangle for 16 treatments with constituent Latin squares arranged in a natural order

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

Figure 4.13: A $(4\times 20)/4$ balanced semi-Latin rectangle for 16 treatments with a different arrangement of the constituent Latin squares

4.5.4 Construction for designs of the class $(g(g+1)\times g)/g$

We adapt the construction given in section 4.5.3, but this time, the created array that accommodates the constituent Latin squares is of size $g(g+1)\times g$, and the rows of this array are divided into $g+1$ equal subdivisions of g rows each which are separated by double horizontal lines. Each constituent Latin square is then inserted into a subdivision

1	2	3	4	5	6	7	8	9	1	4	7	2	5	8	3	6	9
7	8	9	1	2	3	4	5	6	3	6	9	1	4	7	2	5	8
4	5	6	7	8	9	1	2	3	2	5	8	3	6	9	1	4	7
1	5	9	2	6	7	3	4	8	1	6	8	2	4	9	3	5	7
3	4	8	1	5	9	2	6	7	3	5	7	1	6	8	2	4	9
2	6	7	3	4	8	1	5	9	2	4	9	3	5	7	1	6	8

Figure 4.14: A $(6 \times 6)/3$ balanced semi-Latin rectangle for 9 treatments

of the array (the order being immaterial) to obtain the design.

We note that a design of the same size can be obtained via transposition of a corresponding $(g \times g(g+1))/g$ BSLR.

4.5.5 Construction for designs of the class $(ge \times gs)/g$, where $es = g + 1$

Given the construction in section 4.5.3. Suppose $g+1$, the number of constituent Latin squares is a nonprime. By putting $es = g+1$, where $e, s \in \mathbb{Z}$ and $1 < e, s < g+1$; modifying step 3 by creating an array of size $(ge \times gs)$ and dividing its rows and columns into e and s subdivisions of equal sizes, respectively, we obtain a design of corresponding size, for all e, s.

Remark. This construction provides some designs other than the $(g \times ges)/g$ that would be obtained by directly implementing the algorithmic procedure if $g+1 = es$. However, their quotient block design is the same for a design of a given size.

Furthermore, there can be as many designs depending on the number of possible values of the pair (e, s). In particular, if $g+1$ is a perfect square, then e can be identical to s. Hence, in the construction, there is a possibility of having a design with identical number of rows as columns.

Example 4.5.3. For instance, notice that, in example 4.5.1, $g+1 = 4$, hence there are 4 constituent Latin squares, Λ_1, Λ_2, Λ_3 and Λ_4, that make the design. Thus $e = s = 2$ and we obtain a $(6 \times 6)/3$ BSLR as a possibility: see Figure 4.14

Notice that the design in Figure 4.14 takes the form

Λ_1	Λ_2
Λ_3	Λ_4

However, Λ_i, $i = 1, 2, 3, 4$ can appear in any subdivision of the (6×6) array, leading to various non-isomorphic designs.

As another illustration, suppose $v = 25$ such that $g = 5$. Then $g+1 = 6$, and $(e, s) = (2, 3), (3, 2)$. Hence, a $(10 \times 15)/5$ BSLR and a $(15 \times 10)/5$ BSLR can be obtained.

Note that a direct implementation of the algorithmic procedure in section 4.5.3 would produce a $(5 \times 30)/5$ BSLR instead.

Similarly, if $g + 1 = 16$, then the possibilities are BSLRs of the sizes $(60 \times 60)/15$, $(30 \times 120)/15$, and $(120 \times 30)/15$. Again, a direct implementation of the procedure in section 4.5.3 would produce a $(15 \times 240)/15$ BSLR.

4.6 Constructions based on Complementation

4.6.1 Preliminaries

Complementation is a useful concept for obtaining another BIBD from an existing one, as the complement of a BIBD is another BIBD with the same numbers of treatments and blocks, though other parameters may be different. Suppose there exists a BIBD with the parameters (v, b, r, k, λ), then its complementary design, which is the design obtained by replacing the treatments of each block by those treatments that are missing from it is a BIBD with the parameters (v, b, r', k', λ'), where $r' = b - r$, $k' = v - k$, and $\lambda' = b - 2r + \lambda$: see for example, Raghavarao and Padgett (2005, Chapter 4) and Street and Street (1987, Chapter 2).

We designate each construction for an $(h \times p)/k$ BSLR for v treatments we have given so far, both in the current chapter and the preceding chapter a direct construction. Notice that $k' = k$, if and only, if $v = 2k$ such that $k = v/2$. Similarly, $k' > k$, if and only, if $v > 2k$ such that $k < v/2$; and $k' < k$, if and only, if $v < 2k$ such that $k > v/2$. We adopt three approaches to complementation, viz, block (cell) complementation, column complementation, and row complementation. In cell complementation, the complementation is done with respect to each cell, while column and row complementations involve complementation with respect to each column and each row, respectively.

Furthermore, using cell complementation, we concentrate on obtaining BSLRs with $k' \geq k$, or equivalently, $v \geq 2k$ such that, $k \leq v/2$, for convenience, since it may be easier to obtain a direct construction for BSLRs with small values of k. However, we employ column and row complementations to obtain designs with two rows and two columns, respectively in situations where $v = 2k$ such that $k = v/2$, which is identical to k', provided there exists a BIBD with p blocks to facilitate column complementation, and similarly, there need to exist a BIBD with h blocks to facilitate row complementation.

4.6.2 Construction by block (cell) complementation

For a given v and k, given any direct construction for an $(h \times p)/k$ BSLR, we obtain construction for an $(h \times p)/k'$ BSLR for the same number of treatments via block complementation, where $k' = v - k$.

Theorem 4.6.1. *Let Γ denote an $(h \times p)/k$ balanced semi-Latin rectangle for v treatments whose QBD is a $(v, hp, hn_r = pn_c, k, \lambda)$-BIBD, where $n_c = kh/v$, $n_r = kp/v$ and $\lambda =$*

$hn_r(k-1)/v-1$. Suppose Γ' is a design obtained from Γ by replacing the treatments in each cell of Γ with those treatments that are missing from that cell, called the complementary treatments with respect to that cell. Then Γ' is an $(h \times p)/k'$ BSLR for the same number of treatments whose QBD is a $(v, hp, h(p-n_r) = p(h-n_c), k', \lambda')$-BIBD, where $k' = v-k$ and $\lambda' = hp - \lambda - 2(hn_r - \lambda)$.

Proof. Let $V = \{1, 2, ..., v\}$ denote the set of treatments in Γ. Furthermore, let S_{ij} and S'_{ij} denote the set of treatments in the (i, j)th cell of Γ and Γ', respectively, where $i = 1, 2, ..., h$ and $j = 1, 2, ..., p$. Then for each $i \in \{1, 2, ..., h\}$ and $j \in \{1, 2, ..., p\}$, $S'_{ij} = V \backslash S_{ij}$. Notice that $|S_{ij}| = k$, for all i and j, where $k < v$, since S_{ij}s are incomplete blocks. Hence $S_{ij} \subset V$, $S'_{ij} \subset V$ and $S'_{ij} \neq \emptyset$. It follows that $k' = |S'_{ij}| = v - k$, where $k' < v$.

Let $\alpha \in V$. Then α appears n_r times, that is, in n_r cells per row and in n_c cells per column in Γ. Let n'_r and n'_c denote the respective number of cells in each row and column of Γ' that α appears. Then $n'_r = p - n_r$ and $n'_c = h - n_c$. By Corollary 4.6.2, $n'_r > 0$ and $n'_c > 0$. Then it follows that Γ' is an $(h \times p)/v - k$ SLR.

We now investigate whether the QBD of Γ' is balanced. Let $\mathcal{B} = \{B_j\}_{j=1}^{hp}$ denote the set of blocks in the QBD of Γ such that $|\mathcal{B}| = hp$ and $|B_j| = k$ for all $j = 1, 2, ..., hp$. Let $A = \{B_j : \alpha \in B_j\}$. Then $|A| = hn_r = pn_c$; $A' = \{B_j : \alpha \notin B_j\} = \mathcal{B} \backslash A$; and $|A'| = hp - hn_r = h(p - n_r)$. Also, $|A'| = hp - pn_c = p(h - n_c)$. Hence $|A'|$ is a positive integer by Corollary 4.6.2.

Now, let $\beta \in V$, where $\beta \neq \alpha$; and let $C = \{B_j : (\alpha, \beta) \in B_j\}$. Then $|C| = \lambda$; $C' = \{B_j : (\alpha, \beta) \notin B_j\} = \mathcal{B} \backslash C$; and $|C'| = hp - \lambda$. Furthermore, by letting $E = \{B_j : \alpha \in B_j, \beta \notin B_j\} = A \backslash C$, then $|E| = hn_r - \lambda$, which is identical to $pn_c - \lambda$.

Let $H = \{B_j : \beta \in B_j\}$. Then $|H| = hn_r = pn_c = |A|$. Let $L = \{B_j : \beta \in B_j, \alpha \notin B_j\} = H \backslash C$. Then $|L| = hn_r - \lambda$, which is identical to $pn_c - \lambda$.

Notice that, for α and β to appear together in the same block of the QBD of Γ', then there must be a corresponding block in the QBD of Γ that contains neither of them. Suppose Z denotes a collection of all such blocks in Γ. Then $Z = \{B_j : \alpha \notin B_j, \beta \notin B_j\} = \mathcal{B} \backslash E \cup L \cup C = C' \backslash E \cup L$. Notice that $E \cap L = \emptyset$. Hence $|Z| = |C'| - (|E| + |L|) = (hp - \lambda) - 2(hn_r - \lambda)$. We note that $|Z|$ is the number of blocks in the QBD of Γ that neither α nor β makes an appearance, that is, those blocks where α is missing and β is also missing.

Now, let W denote a collection of all corresponding blocks in Γ' formed from each member block of Z. Then $W = \{V \backslash B_j : \alpha \notin B_j, \beta \notin B_j\}$ and $|W|$ is precisely the number of blocks in the QBD of Γ' that contain both α and β. Since $|W| = |Z|$, it follows that $\lambda' = hp - \lambda - 2(hn_r - \lambda)$.

Hence the QBD of Γ' is balanced, being a $(v, hp, hn'_r = pn'_c, k', \lambda')$-BIBD, where $n'_r = p - n_r$, $n'_c = h - n_c$, $k' = v - k$ and $\lambda' = hp - \lambda - 2(hn_r - \lambda)$; making Γ' an $(h \times p)/k'$ BSLR.

■

Comment. The proof of Theorem 4.6.1 can be approached more easily by simply showing that Γ' is an $(h \times p)/k'$ SLR for v treatments, where $k' = v - k$ and then recognizing that its QBD is a BIBD, since the complement of a BIBD is another BIBD as Γ' is a design complementary to Γ.

Note that, since the cells in Γ constitute incomplete blocks, then the set of treatments in each cell of Γ is a proper subset of its entire set of treatments and is non-empty, where each cell contains the same number of treatments, since Γ is a SLR. Similarly, the set of treatments in each cell of Γ' is also a proper subset of the set of treatments in Γ and is also non-empty. Moreover, all the treatments in Γ' are also the entire treatments in Γ and each cell of Γ' contains the same number of treatments..

Now, since each cell of Γ' contains the same number of treatments, then to show that Γ' is a SLR, it is sufficient to show that there are positive integers, n'_r and n'_c such that each treatment appears n'_r times per row and n'_c times per column, where $n'_r = p - n_r$ and $n'_c = h - n_c$ (where n_r and n_c denote the respective number of times that each treatment appears per row and per column in Γ). Notice that, by Corollary 4.6.2, each of n'_r and n'_c is a positive integer

Finally, since the QBD of Γ' is a BIBD, then Γ' is an $(h \times p)/k'$ BSLR for v treatments.

Corollary 4.6.1. *$hp - \lambda - 2(hn_r - \lambda) = hp - 2hn_r + \lambda$ and is identical to $h(p - n_r)(v - k - 1)/(v - 1)$*

Corollary 4.6.2. *$n_r < p$ and $n_c < h$ since $k < v$, as the blocks are incomplete.*

Corollary 4.6.3. *By Corollaries 4.6.1 and 4.6.2, $\lambda' \geq 0$. $\lambda' = 0$ if and only if $k = v - 1$. For values of $k \leq v - 2$, $\lambda' > 0$*

Remark. By Corollary 4.6.3, if $k = v - 1$, then $k' = 1$. It follows that each cell of Γ' has exactly 1 treatment with no pair, making $\lambda' = 0$ and if $h = p$, Γ' is trivially, a Latin square of order p, which is a trivial case of the SLR.

Corollary 4.6.4. *If $p > 2n_r$ (or $h > 2n_c$) and $k < v/2$, then $\lambda' > \lambda$.*

Proof. Given $\lambda = hn_r(k - 1)/v - 1$ and $\lambda' = h(p - n_r)(v - k - 1)/v - 1$. Then $\lambda' > \lambda$ if and only if $h(p - n_r)(v - k - 1) > hn_r(k - 1)$. Notice that, if $p - n_r > n_r$ and $v - k > k$, then $\lambda' > \lambda$. The last two statements are equivalent to $p > 2n_r$ and $v > 2k$, respectively; and the last expression is equivalent to $k < v/2$. Hence the result follows. Furthermore, since $hn_r = pn_c$, by putting $n_r = pn_c/h$, then $p > 2n_r$ becomes $h > 2n_c$. ∎

Remark. Similarly, if $p < 2n_r$ (or $h < 2n_c$) and $k > v/2$, then $\lambda' < \lambda$; and if $p = 2n_r$ (or $h = 2n_c$) and $k = v/2$, then $\lambda' = \lambda$.

Moreover, since $n'_r = p - n_r$ and $n'_c = h - n_c$, then the following statements are equivalent: $p > 2n_r$ and $n'_r > n_r$ ($h > 2n_c$ and $n'_c > n_c$); $p < 2n_r$ and $n'_r < n_r$) ($h < 2n_c$ and $n'_c < n_c$); and $p = 2n_r$ and $n'_r = n_r$ ($h = 2n_c$ and $n'_c = n_c$).

Corollary 4.6.5. *$hn_r \leq (\lambda + hp)/2$.*

Proof. From the combinatorial properties of BSLRs, $\lambda < hn_r < hp$. Notice that $hp - \lambda = 2(hn_r - \lambda)$ if and only if $hn_r = (\lambda + hp)/2$, which is the average of λ and hp. Similarly, $hp - \lambda > 2(hn_r - \lambda)$ if and only if $hn_r < (\lambda + hp)/2$; and $hp - \lambda < 2(hn_r - \lambda)$ if and only if $hn_r > (\lambda + hp)/2$.

From Theorem 4.6.1, if $hp - \lambda = 2(hn_r - \lambda)$, then $\lambda' = 0$. Similarly, if $hp - \lambda > 2(hn_r - \lambda)$, then $\lambda' > 0$; and if $hp - \lambda < 2(hn_r - \lambda)$, then $\lambda' < 0$.

Since by Corollary 4.6.3, $\lambda' \geq 0$, then $hp - \lambda \geq 2(hn_r - \lambda)$. It follows that $hn_r \leq (\lambda + hp)/2$.

■

Remark. $(\lambda + hp)/2 \in \mathbb{Z}$ if and only if 2 divides $(\lambda + hp)$, that is, where both λ and hp are either even or odd; or $\lambda = 0$ and hp is even.

Notice that Corollary 4.6.5 is also evident by imposing the nonnegativity condition of λ', where λ' is as given in Theorem 4.6.1.

Corollary 4.6.6. *$k' = v - 1$, if and only, if $k = 1$ and consequently, $\lambda = 0$. Then $\lambda' = hp - 2hn_r$, and provided, $k' > 1$, or equivalently, $v > 2$, then $\lambda' > 0$ such that $p > 2n_r$. Similarly, since $hn_r = pn_c$, then it also follows that $h > 2n_c$.*

Corollary 4.6.7. *Let $k' = k$, where $k > 1$. Then $v - k = k$ such that $v = 2k$, where $v > 2$, or equivalently, $k = v/2$. Hence $\lambda > 0$, $\lambda' > 0$, where $\lambda' = (hp - hn_r)\lambda/hn_r = \lambda n'_r/n_r$. Similarly, $\lambda' = \lambda n'_c/n_c$.*

Corollary 4.6.8. *Corollary 4.6.7 stipulates that if the conditions given there are satisfied, then both λ and λ' are strictly positive; and provided $n'_r > n_r$ (or equivalently, $n'_c > n_c$), then $\lambda' > \lambda$. Similarly, if $n'_r < n_r$ (or equivalently, $n'_c < n_c$), then $\lambda' < \lambda$; and if $n'_r = n_r$ (or equivalently, $n'_c = n_c$), then $\lambda' = \lambda$.*

Remark. Since $n'_r = p - n_r$ and $n'_c = h - n_c$, then Corollary 4.6.8 is equivalent to saying that if those conditions are satisfied; if $p > 2n_r$ (or $h > 2n_c$), then $\lambda' > \lambda$. Similarly, if $p < 2n_r$ (or $h < 2n_c$), then $\lambda' < \lambda$; and if $p = 2n_r$ (or $h = 2n_c$), then $\lambda' = \lambda$.

Corollary 4.6.9. *By Theorem 4.6.1 and Corollary 4.6.6 , an $(h \times p)/v - 1$ BSLR, where $h = p$ can be obtained via cell complementation of a Latin square of order p.*

Construction procedure

Suppose Γ exists and whose direct construction is given. Let $V = \{1, 2, ..., v\}$ denote the treatment set of Γ; and let S_{ij} denote the set of treatments in the (i, j)th cell of Γ, where $i = 1, 2, ..., h$ and $j = 1, 2, ..., p$. For each (i, j)th cell in Γ, by replacing S_{ij} with $S'_{ij} = V \backslash S_{ij}$, where S'_{ij} is the set of treatments complementary to S_{ij}, that is, the set of treatments that are missing from S_{ij}, we obtain an $(h \times p)/k'$ BSLR, where $k' = v - k$, and our interest is on $k' \geq k$.

Remark. The BSLR obtained by cell complementation will hereinafter be called a complementary BSLR.

An Algorithm for the construction

1. Obtain a direct construction for an $(h \times p)/k$ BSLR for v treatments.

2. Create an $h \times p$ array.

3. For $i = 1, 2, ..., h$ and $j = 1, 2, ..., p$, put in the (i, j)th cell of the array, the set, $S'_{ij} = V \backslash S_{ij}$ of treatments, where S_{ij} is the set of treatments in the corresponding cell of the design in step 1.

Remark. An implementation of this algorithm produces a complementary BSLR for v treatments that is of size $(h \times p)/k'$, where $k' = v - k$.

An $(h \times p)/v - 1$ BSLR for which $h = p$ that is a complementary design can be obtained by implementing the algorithm with the design in step 1 being a $p \times p$ Latin square.

Example 4.6.1. We obtain, by block complementation, a $(7 \times 7)/4$ BSLR for 7 treatments shown in Figure 4.15 using the $(7 \times 7)/3$ BSLR for 7 treatments given in Figure 4.6 as the parent design.

3 5 6 7	1 4 6 7	1 2 5 7	1 2 3 6	2 3 4 7	1 3 4 5	2 4 5 6
2 4 5 6	3 5 6 7	1 4 6 7	1 2 5 7	1 2 3 6	2 3 4 7	1 3 4 5
1 3 4 5	2 4 5 6	3 5 6 7	1 4 6 7	1 2 5 7	1 2 3 6	2 3 4 7
2 3 4 7	1 3 4 5	2 4 5 6	3 5 6 7	1 4 6 7	1 2 5 7	1 2 3 6
1 2 3 6	2 3 4 7	1 3 4 5	2 4 5 6	3 5 6 7	1 4 6 7	1 2 5 7
1 2 5 7	1 2 3 6	2 3 4 7	1 3 4 5	2 4 5 6	3 5 6 7	1 4 6 7
1 4 6 7	1 2 5 7	1 2 3 6	2 3 4 7	1 3 4 5	2 4 5 6	3 5 6 7

Figure 4.15: A $(7 \times 7)/4$ complementary BSLR for 7 treatments

Notice that the parent design has $n_c = n_r = 3$ and $\lambda = 7$. It follows that its QBD is a $(7, 49, 21, 3, 7)$-BIBD. The complementary design has the parameters $n'_c = n'_r = 4$ and $\lambda' = 14$. Hence its QBD is a $(7, 49, 28, 4, 14)$-BIBD, which conforms to Theorem 4.6.1 and the Corollaries.

Notice that, for instance, $p > 2n_r$, $h > 2n_c$, and $k < v/2$. Hence $\lambda' > \lambda$: see Corollary 4.6.4.

Example 4.6.2. From the $(3 \times 15)/2$ BSLR for 6 treatments shown in Figure 3.6, in Chapter 3, we obtain a corresponding complementary design which is a $(3 \times 15)/4$ BSLR shown in Figure 4.16.

2	3	1	3	1	2	1	3	2	4	1	2	1	2	1	3	2	3	1	2	1	2	3	4	1	2	2	3	1	4
4	5	4	∞	5	∞	4	5	5	∞	3	∞	4	5	5	∞	4	∞	3	5	4	∞	5	∞	3	4	5	∞	5	∞
1	2	2	3	1	3	1	2	1	3	2	4	2	3	1	2	1	3	3	4	1	2	1	2	1	4	1	2	2	3
5	∞	4	5	4	∞	3	∞	4	5	5	∞	4	∞	4	5	5	∞	5	∞	3	5	4	∞	5	∞	3	4	5	∞
1	3	1	2	2	3	2	4	1	2	1	3	1	3	2	3	1	2	1	2	3	4	1	2	2	3	1	4	1	2
4	∞	5	∞	4	5	5	∞	3	∞	4	5	5	∞	4	∞	4	5	4	∞	5	∞	3	5	5	∞	5	∞	3	4

Figure 4.16: A $(3 \times 15)/4$ complementary BSLR for 6 treatments

Remark. The treatment set for either design is $V = \{1, 2, ..., \infty\}$, where ∞ is a special treatment symbol that was used in the construction of the parent design, that is, the $(3 \times 15)/2$ BSLR for 6 treatments.

We remind that in the parent design, $n_c = 1$, $n_r = 5$, and $\lambda = 3$, hence its QBD being a $(6, 45, 15, 2, 3)$-BIBD: see the remarks section in Example 3.5.2. Notice that, for the complementary design, $n'_c = 2$, $n'_r = 10$ and $\lambda' = 18$, with a $(6, 45, 30, 4, 18)$-BIBD as its QBD, which is consistent with the results of Theorem 4.6.1 and the Corollaries.

Example 4.6.3. By our earlier remark, a $(4\times4)/3$ BSLR for 4 treatments can be obtained via block complementation by using a Latin square of order 4 as a parent design.

Let the parent design be the Latin square in Figure 4.17. Then we obtain the complementary design shown in Figure 4.18.

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

Figure 4.17: A 4×4 Latin square used to obtain a $(4 \times 4)/3$ BSLR via block complementation

Notice that, in the Latin square, $n_r = n_c = 1$ and its QBD is trivially a $(4, 16, 4, 1, 0)$-BIBD; while for the complementary design, the parameters $n'_r = n'_c = 3$ and its QBD is a $(4, 16, 12, 3, 8)$-BIBD. These results conform to the Corollaries.

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

Figure 4.18: A $(4 \times 4)/3$ complementary BSLR for 4 treatments

4.6.3 Construction by column complementation

We consider construction for BSLRs for $v = 2k$ treatments in two rows and p columns, which are precisely BSLRs of the class $(2 \times p)/k$, where $k = v/2$. Notice that, for such designs, each treatment would need to appear exactly once in each column, that is, $n_c = 1$. Similarly, for each row, each treatment needs to appear $p/2$ times, making $n_r = p/2$. These follow from $k = v/2$, $h = 2$, and p. Furthermore, the construction works if and only if there exists a BIBD for $v = 2k$ treatments in p blocks of size $k = v/2$, since it follows that there also exists a BIBD in $2p$ blocks for the same values of v and k.

Construction procedure

Suppose there exists a BIBD for v treatments in p blocks of size $v/2$. The p blocks of the BIBD are inserted into the first row of a $(2 \times p)$ array, thereby forming row 1 of the design. Then the treatments/entries for each cell in row 2 of the design are generated via column complementation, viz obtaining the complement of those treatments of the corresponding cell in row 1, which are precisely, the treatments that are missing from the cell directly above it.

Let $V = \{1, 2, ..., v\}$ denote the set of treatments of the BIBD which corresponds to the set of treatments for the BSLR under construction. Let i and j denote the respective row and column labels for the array, where $i = 1, 2$ and $j = 1, 2, ..., p$. Furthermore, let A_{ij} denote the set of treatments in the (i, j)th cell. Then for all j, put $A_{1j} = B_j$, where B_j is the jth block of the BIBD. Put $A_{2j} = V \backslash B_j$.

An Algorithm for the construction

1. Obtain a BIBD for v treatments in p blocks of size $k = v/2$, if one exists.

2. Create a $2 \times p$ array and insert the p blocks from the BIBD obtained in step 1 to form row 1 of the design by putting $A_{1j} = B_j$, for all $j = 1, 2, ..., p$ where A_{1j} is the set of treatments in the jth cell of row 1, and B_j is the jth block of the BIBD.

3. For all $j = 1, 2, ..., p$, put $A_{2j} = V \backslash B_j$, where V is the set of treatments, and A_{2j} is the set of treatments in the jth cell of row 2, that is, cell $(2, j)$.

Comments. Another BSLR of the same size and isomorphic to that obtained via the algorithm can be obtained by cell complementation of the constructed design. This is equivalent to swapping the cells in each column.

Each row of the design is a BIBD. The construction involves adding a complementary BIBD to an existing/parent BIBD, which results in another BIBD. Hence the QBD of the constructed design is a BIBD comprising 2 BIBDs each of which contains $v = 2k$ treatments in p incomplete blocks of size $k = k' = v/2$, each treatment being replicated $n_r = n'_r = p/2$ times and concurrences $\lambda_1 = \lambda'_1 = p(k-1)/2(2k-1)$. Overall, there are $2p$ blocks, and each pair of treatments concur in $\lambda = p(k-1)/2k-1$ blocks. We note that, $\lambda_1 = \lambda'_1$, since $n_r = n'_r$: see Corollary 4.6.7. Notice that $\lambda_1 = \lambda/2$.

Example 4.6.4. We observe, from our foregoing discussion that, if $p = 10$, for instance, then the treatment concurrences, λ in the SLR under construction is given by $\lambda = 10(k-1)/(2k-1)$, which is identical to $(k-1)/(k/5 - 1/10)$. Notice that $\lambda = 5 - 5/(2k-1) \in \mathbb{Z}_+ \cup \{0\}$ if and only if $2k - 1 = 1, 5$ such that $k = 1, 3$. In particular, if $k = 1$, then $\lambda = 0$, the trivial case; and if $k = 3$, then $\lambda = 4$. Furthermore, $v = 2k = 6$, $n_c = 1$, and $n_r = 5$. Hence there exists a $(2 \times 10)/3$ BSLR for $v = 6$ treatments whose QBD is a $(6, 20, 10, 3, 4)$-BIBD. It follows that there exists a $(6, 10, 5, 3, 2)$-BIBD which can serve as a parent BIBD (a BIBD whose blocks are to form row 1 of the BSLR under construction and whose complement forms row 2) for the construction, since $\lambda_1 = \lambda/2 = 2$ is an integer.

We first give a construction for a parent BIBD.

Construction of a parent BIBD

Notice that the parent BIBD contains 6 treatments in 10 blocks of size 3 where each treatment is replicated 5 times and each pair of treatments concur 2 times. We approach it from a combinatorial perspective. Let the treatment set be $V = \{1, 2, 3, 4, 5, \infty\}$. By identifying treatments 1 to 5 with the vertices of a 5-gon and the treatment with label ∞ kept outside the polygon, associate each pair of vertices at distance 1 with the symbol ∞, forming a triple/block each time. This generates the first 5 blocks, viz, $B_1 = \{1, 2, \infty\}$, $B_2 = \{2, 3, \infty\}$, $B_3 = \{3, 4, \infty\}$, $B_4 = \{4, 5, \infty\}$, $B_5 = \{5, 1, \infty\}$. The remaining 5 blocks are generated by associating the same pairs of vertices at distance 1, this time with the unique vertex that is equidistant from each vertex in each 2-subset of vertices at distance 1. This gives $B_6 = \{1, 2, 4\}$, $B_7 = \{2, 3, 5\}$, $B_8 = \{3, 4, 1\}$, $B_9 = \{4, 5, 2\}$, $B_{10} = \{5, 1, 3\}$. Notice that each unique vertex is at distance $(v-2)/2$ from each member vertex of the

2-subsets. We now obtain, as follows, an expression for easily generating the 10 blocks of the BIBD

Let the treatments that make the vertices of the 5-gon be numbered in a cyclic order, viz, $i, i+1, i+2, i-2, i-1$, where $i \in V\backslash\{\infty\}$, reducing each component modulo 5. Notice that, for each $i \in V\backslash\{\infty\}$, the pairs $ii, i \pm 1$ are each at distance 1, and by denoting the unique vertex equidistant from both i and $i+1$ by τ_{ii+1}, then $\tau_{ii+1} = i + 1 + (v-2)/2$ or $i - (v-2)/2$, which simplifies to $(2i+v)/2$ or $(2i - v + 2)/2$, respectively with a reduction modulo 5 for each component. Similarly, if the pair $(i, i-1)$ of vertices is considered, then the unique vertex, τ_{ii-1} that is equidistant from each of these vertices is $\tau_{ii-1} = (2i + v - 2)/2$ or $(2i - v)/2$, where each component is also reduced modulo 5.

Let B_j denote the jth block of the BIBD, for all $j = 1, 2, ..., 10$. Define

$$B_j = \begin{cases} \{j, j+1, \infty\} & \text{, if } j = 1, 2, ..., 5 \\ \{j, j+1, \tau_{jj+1} & \text{, if } j = 6, 7, ..., 10 \end{cases} \tag{4.2}$$

where each component is reduced modulo 5, and $\tau_{jji+1} = (2j+v)/2$ or $(2j - v + 2)/2$, reducing each component modulo 5, as well. Notice from (4.2) that, $B_1 = \{1, 2, \infty\}$, $B_2 = \{2, 3, \infty\}$, $B_3 = \{3, 4, \infty\}$, $B_4 = \{4, 5, \infty\}$, $B_5 = \{5, 1, \infty\}$, $B_6 = \{1, 2, 4\}$, $B_7 = \{2, 3, 5\}$, $B_8 = \{3, 4, 1\}$, $B_9 = \{4, 5, 2\}$ and $B_{10} = \{5, 1, 3\}$, as before.

Remark. Notice that, with an initial block, $B_1^* \in \{B_1, B_2, ..., B_5\}$, the other 4 blocks can be generated by a cyclic development of B_1^*. Hence, $B_j^* = B_1^* + (j-1)$, for $j = 2, 3, 4, 5$, which is equivalent to $B_j^* = B_{j-1}^* + 1$, for $j = 2, 3, 4, 5$, with reduction modulo 5. Note that $B_u + \delta = \{x + \delta : x \in B_u\}$, and $\infty + \theta = \infty$.

Notice that, B_4^*, for instance, where $B_4^* \in \{B_1, B_2, ..., B_5\}$ is obtained to be $B_4^* = B_1^* + 3$. Suppose $B_1^* = B_1 = \{1, 2, \infty\}$. Then $B_4^* = \{4, 5, \infty\}$, which is the block B_4 obtained before. Furthermore, using the equivalent expression, suppose $B_3^* = B_3 = \{3, 4, \infty\}$. Then $B_4^* = B_3^* + 1$, which also results in $B_4^* = \{4, 5, \infty\}$, as before.

Similarly, by developing an initial block, $B_6^+ \in \{B_6, B_7, ..., B_{10}\}$, the remaining blocks can also be generated. Thus, the blocks can be generated, viz, $B_j^+ = B_6^+ + (j-1)$, for $j = 7, 8, 9, 10$, which is equivalent to $B_j^+ = B_{j-1}^+ + 1$. For instance, $B_8^+ = B_6^+ + 2$. Suppose $B_6^+ = B_6 = \{1, 2, 4\}$. Then $B_8^+ = \{3, 4, 1\} = B_8$. Moreover, if the equivalent expression is used, then $B_8^+ = B_7^+ + 1$. Suppose $B_7^+ = B_7 = \{2, 3, 5\}$. Then $B_8^+ = \{3, 4, 1\}$, as before.

We note that B_1^* can be any set in $\{B_1, B_2, ..., B_5\}$. Similarly, B_6^+ can also be any set in $\{B_6, B_7, ..., B_{10}\}$, not necessarily B_1 and B_6, respectively.

We now proceed to construct the $(2 \times 10)/3$ BSLR for 6 treatments by implementing the algorithm: see Figure 4.19.

Notice from the design in Figure 4.19 that, for cell (i, j), where $i = 1, 2$ and $j = 2, ..., 5, 7, ..., 10$,

1 2 ∞	2 3 ∞	3 4 ∞	4 5 ∞	5 1 ∞	1 2 4	2 3 5	3 4 1	4 5 2	5 1 3
3 5 4	4 1 5	5 2 1	1 3 2	2 4 3	3 5 ∞	4 1 ∞	5 2 ∞	1 3 ∞	2 4 ∞

Figure 4.19: A $(2 \times 10)/3$ BSLR for 6 treatments obtained by column complementation

$$A_{ij} = \begin{cases} A_{i1} + (j-1) & \text{, if } j = 2, ..., 5 \\ A_{i6} + (j-1) & \text{, if } j = 7, ..., 10 \end{cases}$$

where each component is reduced modulo 5. This shows that, given, for instance, the entries in each cell of column 1, the entries of the cells in columns 2 to 5 can be generated by a cyclic development of the cell in column 1 of the corresponding row via successive addition of 1 (reduced modulo 5). For instance, $A_{14} = A_{11}+3$, noting that $A_{ij}+\gamma = \{y+\gamma : y \in A_{ij}$ and $\infty + \eta = \infty$. Similarly, $A_{25} = A_{21} + 4$.

Similarly, given the entries in each cell of column 6, the entries of the cells in columns 7 to 10 can be generated by a cyclic development of the cell in column 6 of the corresponding row via successive addition of 1 (reduced modulo 5). For instance, $A_{17} = A_{16} + 1$; and similarly, $A_{28} = A_{26} + 2$.

Notice also that cell complementation of the design in Figure 4.19 produces the design in Figure 4.20, which is of the same size and can also be obtained by swapping the cells in each column of the original design.

3 5 4	4 1 5	5 2 1	1 3 2	2 4 3	3 5 ∞	4 1 ∞	5 2 ∞	1 3 ∞	2 4 ∞
1 2 ∞	2 3 ∞	3 4 ∞	4 5 ∞	5 1 ∞	1 2 4	2 3 5	3 4 1	4 5 2	5 1 3

Figure 4.20: A $(2 \times 10)/3$ BSLR for 6 treatments obtained by cell complementation of the design in Figure 4.19

4.6.4 Construction by row complementation

In a similar manner to obtaining designs with two rows, where we filled the first row of the design with the blocks of a BIBD, and then generated the entries of each cell in the second row by taking the complement of the entries in corresponding cell in the first row. We modify, slightly, this procedure to obtain designs for two columns by filling the ffirst column of an $h \times 2$ array with the h blocks of a BIBD; and then generating the entries of each cell in the second column by obtaining the complement of corresponding cell in the first column.

Given $V = \{1, 2, ..., v\}$, the set of treatments of the BIBD and the BSLR under construction. Let R_{ij} denote the set of treatments in the (i, j)th cell of the $h \times 2$ array, where $i = 1, 2, ..., h$ and $j = 1, 2$. Then for all i, put $R_{i1} = B_i$, where B_i is the ith block of the BIBD used for the construction. Put $R_{i2} = V \backslash B_i$.

An Algorithm for the construction

1. Obtain a BIBD for v treatments in h blocks of size $k = v/2$, if one exists.

2. Create an $h \times 2$ array and insert the h blocks from the BIBD obtained in step 1 to form column 1 of the design by putting $R_{i1} = B_i$, for all $i = 1, 2, ..., h$ where R_{i1} is the set of treatments in the ith cell of column 1, and B_i is the ith block of the BIBD.

3. For all $i = 1, 2, ..., h$, put $R_{i2} = V \backslash B_i$, where V is the set of treatments, and R_{i2} is the set of treatments in the ith cell of column 2, that is, cell $(i, 2)$.

This procedure produces an $(h \times 2)/k$ BSLR for v treatments, where $k = v/2$.

Comments. Another $(h \times 2)/k$ BSLR, where $k = v/2$ can be obtained via cell complementation of the constructed design, and this is equivalent to swapping the cells in each row. The resulting design is isomorphic to that obtained via the algorithm.

The construction works if and only if there exists a BIBD for $v = 2k$ treatments in h blocks of size $k = v/2$, since there will also exist a BIBD in $2h$ blocks for same v and k.

This design is analogous to the design with two rows in the preceding section. For instance, each column of the design in this section is a BIBD, just like each row in the previous design. In particular, the same BIBD used in the construction of the $(2 \times p)/k$ BSLR in the preceding section can also be used for this construction, if it is required that $h = p$.

4.7 Constructions for designs of larger sizes

BSLRs of larger sizes can be obtained from another BSLR of smaller size, both having same number of treatments by making copies of the smaller design and then putting them in an array of appropriate size. The design obtained has identical block size, as the original design of smaller size but there are usually, more rows and/or columns, making it larger, except in the trivial case that involves only one copy of the 'smaller design '.

Theorem 4.7.1. *Let Δ_1 denote an $(h \times p)/k$ BSLR for v treatments. Suppose Δ_2 is a design obtained from Δ_1 by making copies of Δ_1 and putting them in an $h^\dagger \times p^\dagger$ array, where $h|h^\dagger$ and $p|p^\dagger$. Then Δ_2 is an $(h^\dagger \times p^\dagger)/k$ BSLR.*

Proof. Let $\alpha\beta$ copies of Δ_1 be made and put in an array which has been partitioned into α sub-rows and β sub-columns such that there are h rows in each sub-row and p columns in each sub-column. Since $h|h^\dagger$ and $p|p^\dagger$, then it follows that $h^\dagger = \alpha h$ and $p^\dagger = \beta p$.

Clearly, the array is of size $(\alpha h \times \beta p)$. Hence Δ_2 is of size $(\alpha h \times \beta p)/k$, since Δ_1 contains k treatments in each row-column intersection.

Let n_r and n_c denote the respective number of times each treatment appears per row and per column in Δ_1. Similarly, let $n_r^\dagger$ and $n_c^\dagger$ denote corresponding parameters in Δ_2. Suppose Δ_2 is a SLR. Then there exists $n_r^\dagger \in \mathbb{Z}_+$ and $n_c^\dagger \in \mathbb{Z}_+$. Notice that $n_r^\dagger = \beta n_r$ and $n_c^\dagger = \alpha n_c$, both positive integers, making Δ_2. a SLR.

Furthermore, since Δ_1 is a BSLR, then its QBD is a BIBD, which contains hp blocks. The QBD of Δ_2 contains $\alpha\beta hp$ blocks (which is a multiple of the number of blocks in Δ_1) and these are the hp blocks in Δ_1 each with multiplicity $\alpha\beta$. It follows that Δ_2 is a BSLR.

■

Corollary 4.7.1. *If $\beta = 1$, then Δ_1 and Δ_2 have the same number of columns and Δ_2 is an $(\alpha h \times p)/k$ BSLR. Similarly, if $\alpha = 1$, then Δ_1 and Δ_2 have the same number of rows and Δ_2 is an $(h \times \beta p)/k$ BSLR. Furthermore, if $\alpha = \beta = 1$, then Δ_2 is trivially Δ_1.*

Corollary 4.7.2. *Let $p \geq h$. If $\beta > \alpha$, then Δ_2 contains more columns than rows. However, if $p > h$ and $\beta < \alpha$, then Δ_2 may (or may not) have more columns than rows. In particular, Δ_2 would have more columns that rows if the difference $\beta p - \alpha h > 0$ but fewer columns than rows if $\beta p - \alpha h < 0$. Similarly, if $h \geq p$ and $\alpha > \beta$, then there are more rows than columns in Δ_2. But if $h > p$ and $\alpha < \beta$, then Δ_2 may (or may not) have more rows than columns. In this case, Δ_2 would have more rows than columns if $\alpha h - \beta p > 0$ but would have fewer rows than columns if $\alpha h - \beta p < 0$. Moreover, if $\beta > 1$, then Δ_2 contains more columns than Δ_1; and similarly, if $\alpha > 1$, then Δ_2 contains more rows than Δ_1. Hence, if at least one of α and β is greater than 1, then Δ_2 is larger in size than Δ_1.*

Corollary 4.7.3. *Since the QBD of Δ_1 is a $(v, hp, hn_r = pn_c, k, \lambda)$-BIBD, then the QBD of Δ_2 is a $(v, h^\dagger p^\dagger, h^\dagger n_r^\dagger = p^\dagger n_c^\dagger, k, \lambda^\dagger)$-BIBD, where $h^\dagger = \alpha h$, $p^\dagger = \beta p$, $h^\dagger n_r^\dagger = \alpha\beta h n_r$, which is identical to $\alpha\beta p n_c$; and $\lambda^\dagger = \alpha\beta\lambda$.*

Corollary 4.7.4. *By the last expression in Corollary 4.7.3, Δ_1 and Δ_2 have identical concurrences if and only if the construction of Δ_2 involves only one copy of Δ_1, that is, the trivial case of Δ_2.*

Corollary 4.7.5. *$\alpha = n_c^\dagger / n_c$ and is identical to $h^\dagger / h$. Similarly, $\beta = n_r^\dagger / n_r$, which is identical to $p^\dagger / p$.*

Corollary 4.7.6. *By Corollary 4.7.5, $\alpha = 1$ if and only if $h^\dagger = h$. Similarly, $\beta = 1$ if and only if $p^\dagger = p$.*

4.7.1 Construction procedure

Given an $(h \times p)/k$ BSLR for v treatments, we name it an initial design and then make copies of this initial design; create an array of size corresponding to the size of the design whose construction is sought; and subsequently, insert these copies into the array.

Example 4.7.1. Suppose a $(4 \times 10)/3$ BSLR for 6 treatments is of interest. Then the $(2 \times 10)/3$ BSLR in Figure 4.19 can serve as the initial design. Hence, by making 2 copies of the basic design, and subsequently putting them in a (4×10) array, we obtain the design of Figure 4.21.

1 2 ∞	2 3 ∞	3 4 ∞	4 5 ∞	5 1 ∞	1 2 4	2 3 5	3 4 1	4 5 2	5 1 3
3 5 4	4 1 5	5 2 1	1 3 2	2 4 3	3 5 ∞	4 1 ∞	5 2 ∞	1 3 ∞	2 4 ∞
1 2 ∞	2 3 ∞	3 4 ∞	4 5 ∞	5 1 ∞	1 2 4	2 3 5	3 4 1	4 5 2	5 1 3
3 5 4	4 1 5	5 2 1	1 3 2	2 4 3	3 5 ∞	4 1 ∞	5 2 ∞	1 3 ∞	2 4 ∞

Figure 4.21: A $(4 \times 10)/3$ BSLR for 6 treatments obtained by inserting 2 copies of an initial design

Comments. Notice in Example 4.7.1 that, by Theorem 4.7.1 and Corollary 4.7.5, $\alpha = 2$ and $\beta = 1$, hence $\alpha > \beta$. The construction produces a design with fewer rows than columns: see Corollary 4.7.2. Note that $h < p$ and $\alpha h < \beta p$.

Furthermore, the QBD of the design produced is a $(v, \alpha\beta hp, \alpha\beta hn_r = \alpha\beta pn_c, k, \alpha\beta\lambda)$-BIBD: see Corollary 4.7.3 and Theorem 4.7.1. where $v = 6$, $h = 2$, $p = 10$, $n_r = 5$ $n_c = 1$, $k = 3$ and $\lambda = 4$. Hence the QBD of the design in Figure 4.21 is a $(6, 40, 20, 3, 8)$-BIBD.

4.7.2 Designs of the classes $(2h \times p)/k$ and $(h \times 2p)/k$

Given an initial design, an $(h \times p)/k$ BSLR for v treatments. Then by the method given in section 4.7.1, a corresponding $(2h \times p)/k$ (or $(h \times 2p)/k$) BSLR can be obtained by making 2 copies of the initial design and then inserting them into the $2h \times p$ (or $h \times 2p$) array.

However, we note that, if $k = v/2$, then a design of the same size can also be obtained by putting the $(h \times p)/k$ BSLR for v treatments with its complementary design, which is also an $(h \times p)/k$ BSLR for v treatments in an array of corresponding size. The designs obtained using the two methods are isomorphic.

Theorem 4.7.2. *Let Λ_1 denote an $(h \times p)/k$ BSLR for v treatments, where $k = v/2$. Let Λ_2 denote its complementary design, the design obtained from Λ_1 by replacing the treatments in each cell of Λ_1 with the treatments missing from that cell. Then the design resulting from putting Λ_1 and Λ_2 in an array of appropriate size is a BSLR for the same number of treatments.*

Proof. Let $V = \{1, 2, ..., v\}$ denote the set of treatments in each of Λ_1 and Λ_2. Let $\tau \in V$. For all $\tau \in V$, let n_r and n_c denote the number of times τ appears in each row and each column, respectively, of Λ_1. Since Λ_2 is complementary to Λ_1, then its treatment set is identical to that of Λ_1 and it also has the same number of rows and columns, hence same number of blocks as Λ_1. Moreover, since $k = v/2$, then each cell of Λ_2 contains $k' = v/2$ treatments, since $k' = v - k$. Hence, each cell in the overall design also contains $v/2$ treatments Let n'_r and n'_c denote the respective number of times that τ appears in each row and each column of Λ_2. Then by Theorem 4.6.1, $n'_r = p - n_r$ and $n'_c = h - n_c$, which are all positive integers.

Since there are 2 BSLRs to be used in the construction, supposing the array is of size $2h \times p$ and also supposing Λ_2 is placed underneath Λ_1 in the array, then we obtain another design that has $2h$ rows and p columns. Denote this design by Γ_{12}, and let n_r^+ and n_c^+ denote the number of times that τ appears in each row and each column, respectively of Γ_{12}. Suppose Γ_{12} is a SLR. Then n_r^+ and n_c^+ must be positive integers. Notice that $n_r^+ = n_r$, where $n_r \in \mathbb{Z}_+$ and $n_c^+ = n_c + n'_c$ whose value is $h \in \mathbb{Z}_+$. Hence Γ_{12} is a SLR.

Furthermore, by Theorem 4.6.1, Λ_2 is a BSLR. Notice that the QBD of Γ_{12} comprises the blocks of two BIBDs, hence it is also a BIBD. Thus Γ_{12} is a BSLR.

Now, supposing Λ_1 is placed underneath Λ_2 in the array, then another design, Γ_{21}, say, is obtained. Notice that the QBDs of Γ12 and Γ_{21} are identical, hence they have same parameters. For instance, in Γ_{21}, τ appears $n'_c + n_c$ times in each column, and in each row it appears n'_r times, both positive integers. Hence, Γ_{21} is also a BSLR.

Moreover, suppose Λ_2 is places beside Λ_1 in the array. Let the new design be denoted Γ_{12}^*. Let n_r^* and n_c^* denote the number of times τ appears per row and column, respectively. Then $n_r^* = n_r + n'_r$ and $n_c^* = n_c$, which are both positive integers, making Γ_{12}^* a SLR. Notice that Γ_{12}^* also contain identical blocks as Γ_{12} and Γ_{21}, hence same QBD and is thus a BSLR. Suppose Λ_1 is placed beside Λ_2 in the array. Then this produces another design, Γ_{21}^*, say, which has identical QBD as Γ_{12}^*, where τ appears $n'_r + n_r$ times in each row and n'_c in each column. Hence Γ_{21}^* is also a BSLR.

Since each of the four possible designs is a BSLR, then the result of the theorem follows. ∎

Corollary 4.7.7. *The QBD of each resulting BSLR, Γ_{12}, Γ_{21}, Γ_{12}^* and Γ_{21}^* is a $(v, 2hp, 2hn_r = 2pn_c = hp, v/2, 2\lambda)$-BIBD. The number of replications of each treatment in each of these designs is identical to the number of blocks in Λ_1 or Λ_2.*

Corollary 4.7.8. *This construction works only if both Λ_1 and Λ_2 have identical block size, $k = v/2$, since k and k' can only be equal if $v = 2k$ such that $k = v/2$ and $k' = v/2$, where $k' = v - k$.*

Corollary 4.7.9. *The order in which Λ_1 and Λ_2 are inserted into the array is immaterial.*

Example 4.7.2. Let Λ_1 denote the $(2 \times 10)/3$ BSLR for 6 treatments in Figure 4.19; and let Λ_2 denote its complementary design in Figure 4.20. Then, we obtain a $(4 \times 10)/3$ BSLR for 6 treatments by putting Λ_1 and Λ_2 in a 4×10 array. Supposing we insert Λ_2

underneath Λ_1, then the design presented in Figure 4.22 is obtained, which is another design of the same size as the design in Figure 4.21 and is also isomorphic to it.

1 2 ∞	2 3 ∞	3 4 ∞	4 5 ∞	5 1 ∞	1 2 4	2 3 5	3 4 1	4 5 2	5 1 3
3 5 4	4 1 5	5 2 1	1 3 2	2 4 3	3 5 ∞	4 1 ∞	5 2 ∞	1 3 ∞	2 4 ∞
3 5 4	4 1 5	5 2 1	1 3 2	2 4 3	3 5 ∞	4 1 ∞	5 2 ∞	1 3 ∞	2 4 ∞
1 2 ∞	2 3 ∞	3 4 ∞	4 5 ∞	5 1 ∞	1 2 4	2 3 5	3 4 1	4 5 2	5 1 3

Figure 4.22: A $(4 \times 10)/3$ BSLR for 6 treatments obtained by inserting a complementary design underneath an initial design

Comments. If Λ_1 is inserted underneath Λ_2 in the array, another design of the same size is also obtained, which is isomorphic to each of the other two designs. A $(2 \times 20)/3$ BSLR can be obtained by putting either of Λ_1 and Λ_2 beside the other in a 2×20 array. Similarly, swapping the order of inserting them into the array produces another design of the same size as the former, the two being isomorphic.

We note that another $(4 \times 10)/3$ BSLR for the same number of treatments can also be obtained by inserting 2 copies of Λ_2, the complementary design to Λ_1. The resulting design is isomorphic to each of the other three designs of its size.

By Corollary 4.7.7, the QBD of this design is a $(6, 40, 20, 3, 8)$-BIBD, which is identical to the QBD of the design in Figure 4.21.

4.7.3 Designs with $h = p$

In our previous constructions of BSLRs for those experimental situations where $h < p$ or $h > p$, in most cases, we juxtaposed certain Latin squares to obtain a basic (initial) design from where a design of larger size was obtained by making copies of the initial design and then juxtaposing them appropriately.

However, in certain experimental situations, interest may be on designs which have identical number of rows and columns, that is, $h = p$. In this circumstance, the basic design (or some other initial design) may be utilized in various ways to obtain constructions for designs of such class if it satisfies certain conditions, as discussed below. In particular, we give three approaches to achieving this. One of the procedures involves making copies of an initial design that satisfy the given condition and then juxtaposing them appropriately; while the other two procedures involve making a single Latin square whose symbols take different forms, for different procedures.

Suppose the initial design is an $(h \times p)/k$ BSLR for v treatments, where $h < p$. If h divides p, then making p/h copies of the initial design and inserting them in a $p \times p$ array, one underneath another produces a $(p \times p)/k$ BSLR. Similarly, if the initial design is such that $h > p$, then, provided p divides h, by making h/p copies of the initial design and then inserting them into an $h \times h$ array, one beside another produces an $(h \times h)/k$ BSLR.

Furthermore, suppose there exists a BIBD for v treatments in p blocks of size k. Then a $(p \times p)/k$ BSLR can also be obtained by making a $p \times p$ Latin square whose symbol set is constituted by the labels of the p blocks of the BIBD. Similarly, if there exists a BIBD for v treatments in h blocks of size k, then an $(h \times h)/k$ BSLR can be obtained by making a Latin square of order h with symbols, the labels of the h blocks of the BIBD.

Moreover, suppose $h < p$. If the initial design consists of θ sub-Latin squares, Λ_i, where $i = 1, 2, ..., \theta$, and $\theta = p/h$, the ith Latin square being of order $p/\theta = h$ with its symbol set comprising the labels of those blocks in row 1, say, of the ith subdivision of the columns of the initial BSLR. Then a $(p \times p)/k$ BSLR can also be obtained by making another Latin square, Γ, say, of order θ, whose symbols are Λ_i, where $i = 1, 2, ..., \theta$.

Similarly, if $p < h$, let the initial design consist of η sub-Latin squares, Υ_u, where $u = 1, 2, ..., \eta$, and $\eta = h/p$, the uth Latin square being of order $h/\eta = p$ with its symbol set comprising the p labels of those blocks in column 1, say, of the uth subdivision of the rows of the initial BSLR. Then an $(h \times h)/k$ BSLR can also be obtained by making another Latin square, Ξ, say, of order η, whose symbols are the labels of Υ_u, where $u = 1, 2, ..., \eta$.

Example 4.7.3. Suppose we wish to make a $(12 \times 12)/3$ BSLR for 9 treatments. We can make use of the $(3 \times 12)/3$ BSLR in Figure 4.11: see example 4.5.1.

Notice that the initial design consists of $\theta = 4$ sub-Latin squares, Λ_i, where $i = 1, 2, 3, 4$ and $h = 3$. Each Latin square is of order $h = 3$. The symbols of Λ_1, for instance, are A_{11}, A_{12} and A_{13}.

Notice also that the design in Figure 4.11 takes the form

Λ_1	Λ_2	Λ_3	Λ_4

where

$\Lambda_1 =$

A_{11}	A_{12}	A_{13}
A_{13}	A_{11}	A_{12}
A_{12}	A_{13}	A_{11}

with A_{11}, A_{12}, and A_{13} being the sets $\{1, 2, 3\}$, $\{4, 5, 6\}$, and $\{7, 8, 9\}$, respectively;

$$\Lambda_2 = \begin{array}{|c|c|c|}\hline A_{21} & A_{22} & A_{23} \\ \hline A_{23} & A_{21} & A_{22} \\ \hline A_{22} & A_{23} & A_{21} \\ \hline \end{array}$$

where A_{21}, A_{22}, and A_{23} are the sets $\{1, 4, 7\}$, $\{2, 5, 8\}$, and $\{3, 6, 9\}$, respectively;

$$\Lambda_3 = \begin{array}{|c|c|c|}\hline A_{31} & A_{32} & A_{33} \\ \hline A_{33} & A_{31} & A_{32} \\ \hline A_{32} & A_{33} & A_{31} \\ \hline \end{array}$$

with A_{31}, A_{32}, and A_{33} being the sets $\{1, 5, 9\}$, $\{2, 6, 7\}$, and $\{3, 4, 8\}$, respectively; and

$$\Lambda_4 = \begin{array}{|c|c|c|}\hline A_{41} & A_{42} & A_{43} \\ \hline A_{43} & A_{41} & A_{42} \\ \hline A_{42} & A_{43} & A_{41} \\ \hline \end{array}$$

where A_{41}, A_{42}, and A_{43} are the sets $\{1, 6, 8\}$, $\{2, 4, 9\}$, and $\{3, 5, 7\}$, respectively.

By making a Latin square of order $\theta = 4$ using Λ_1, Λ_2, Λ_3, and Λ_4 as symbols, we obtain the design shown in Figure 4.23.

Notice that the resulting design shown in Figure 4.23 takes the form

Λ_1	Λ_2	Λ_3	Λ_4
Λ_4	Λ_1	Λ_2	Λ_3
Λ_3	Λ_4	Λ_1	Λ_2
Λ_2	Λ_3	Λ_4	Λ_1

We note that each row of the $(3 \times 12)/3$ BSLR is a BIBD for 9 treatments in 12 blocks of size 3. Another $(12 \times 12)/3$ BSLR for 9 treatments can be obtained by making a Latin square of order 12 whose symbols are the block labels, A_{11}, A_{12}, . . ., A_{43}, of the BIBD. The design produced through this procedure is shown in Figure 4.24

Moreover, by making 4 copies of the $(3 \times 12)/3$ BSLR and putting them in a 12×12 array, we obtain the design shown in Figure 4.25

1 2 3	4 5 6	7 8 9	1 4 7	2 5 8	3 6 9	1 5 9	2 6 7	3 4 8	1 6 8	2 4 9	3 5 7
7 8 9	1 2 3	4 5 6	3 6 9	1 4 7	2 5 8	3 4 8	1 5 9	2 6 7	3 5 7	1 6 8	2 4 9
4 5 6	7 8 9	1 2 3	2 5 8	3 6 9	1 4 7	2 6 7	3 4 8	1 5 9	2 4 9	3 5 7	1 6 8
1 6 8	2 4 9	3 5 7	1 2 3	4 5 6	7 8 9	1 4 7	2 5 8	3 6 9	1 5 9	2 6 7	3 4 8
3 5 7	1 6 8	2 4 9	7 8 9	1 2 3	4 5 6	3 6 9	1 4 7	2 5 8	3 4 8	1 5 9	2 6 7
2 4 9	3 5 7	1 6 8	4 5 6	7 8 9	1 2 3	2 5 8	3 6 9	1 4 7	2 6 7	3 4 8	1 5 9
1 5 9	2 6 7	3 4 8	1 6 8	2 4 9	3 5 7	1 2 3	4 5 6	7 8 9	1 4 7	2 5 8	3 6 9
3 4 8	1 5 9	2 6 7	3 5 7	1 6 8	2 4 9	7 8 9	1 2 3	4 5 6	3 6 9	1 4 7	2 5 8
2 6 7	3 4 8	1 5 9	2 4 9	3 5 7	1 6 8	4 5 6	7 8 9	1 2 3	2 5 8	3 6 9	1 4 7
1 4 7	2 5 8	3 6 9	1 5 9	2 6 7	3 4 8	1 6 8	2 4 9	3 5 7	1 2 3	4 5 6	7 8 9
3 6 9	1 4 7	2 5 8	3 4 8	1 5 9	2 6 7	3 5 7	1 6 8	2 4 9	7 8 9	1 2 3	4 5 6
2 5 8	3 6 9	1 4 7	2 6 7	3 4 8	1 5 9	2 4 9	3 5 7	1 6 8	4 5 6	7 8 9	1 2 3

Figure 4.23: A $(12 \times 12)/3$ BSLR for 9 treatments obtained by making a Latin square with symbols another Latin square

1 2 3	4 5 6	7 8 9	1 4 7	2 5 8	3 6 9	1 5 9	2 6 7	3 4 8	1 6 8	2 4 9	3 5 7
3 5 7	1 2 3	4 5 6	7 8 9	1 4 7	2 5 8	3 6 9	1 5 9	2 6 7	3 4 8	1 6 8	2 4 9
2 4 9	3 5 7	1 2 3	4 5 6	7 8 9	1 4 7	2 5 8	3 6 9	1 5 9	2 6 7	3 4 8	1 6 8
1 6 8	2 4 9	3 5 7	1 2 3	4 5 6	7 8 9	1 4 7	2 5 8	3 6 9	1 5 9	2 6 7	3 4 8
3 4 8	1 6 8	2 4 9	3 5 7	1 2 3	4 5 6	7 8 9	1 4 7	2 5 8	3 6 9	1 5 9	2 6 7
2 6 7	3 4 8	1 6 8	2 4 9	3 5 7	1 2 3	4 5 6	7 8 9	1 4 7	2 5 8	3 6 9	1 5 9
1 5 9	2 6 7	3 4 8	1 6 8	2 4 9	3 5 7	1 2 3	4 5 6	7 8 9	1 4 7	2 5 8	3 6 9
3 6 9	1 5 9	2 6 7	3 4 8	1 6 8	2 4 9	3 5 7	1 2 3	4 5 6	7 8 9	1 4 7	2 5 8
2 5 8	3 6 9	1 5 9	2 6 7	3 4 8	1 6 8	2 4 9	3 5 7	1 2 3	4 5 6	7 8 9	1 4 7
1 4 7	2 5 8	3 6 9	1 5 9	2 6 7	3 4 8	1 6 8	2 4 9	3 5 7	1 2 3	4 5 6	7 8 9
7 8 9	1 4 7	2 5 8	3 6 9	1 5 9	2 6 7	3 4 8	1 6 8	2 4 9	3 5 7	1 2 3	4 5 6
4 5 6	7 8 9	1 4 7	2 5 8	3 6 9	1 5 9	2 6 7	3 4 8	1 6 8	2 4 9	3 5 7	1 2 3

Figure 4.24: A $(12 \times 12)/3$ BSLR for 9 treatments obtained by making a Latin square with symbols the blocks of a BIBD

4.8 Obtaining a lot more designs from the constructions

We observe that, in general, for those basic designs which involve arrangements of two or more Latin squares, a large number of designs other than the ones already discussed

1 2 3	4 5 6	7 8 9	1 4 7	2 5 8	3 6 9	1 5 9	2 6 7	3 4 8	1 6 8	2 4 9	3 5 7
7 8 9	1 2 3	4 5 6	3 6 9	1 4 7	2 5 8	3 4 8	1 5 9	2 6 7	3 5 7	1 6 8	2 4 9
4 5 6	7 8 9	1 2 3	2 5 8	3 6 9	1 4 7	2 6 7	3 4 8	1 5 9	2 4 9	3 5 7	1 6 8
1 2 3	4 5 6	7 8 9	1 4 7	2 5 8	3 6 9	1 5 9	2 6 7	3 4 8	1 6 8	2 4 9	3 5 7
7 8 9	1 2 3	4 5 6	3 6 9	1 4 7	2 5 8	3 4 8	1 5 9	2 6 7	3 5 7	1 6 8	2 4 9
4 5 6	7 8 9	1 2 3	2 5 8	3 6 9	1 4 7	2 6 7	3 4 8	1 5 9	2 4 9	3 5 7	1 6 8
1 2 3	4 5 6	7 8 9	1 4 7	2 5 8	3 6 9	1 5 9	2 6 7	3 4 8	1 6 8	2 4 9	3 5 7
7 8 9	1 2 3	4 5 6	3 6 9	1 4 7	2 5 8	3 4 8	1 5 9	2 6 7	3 5 7	1 6 8	2 4 9
4 5 6	7 8 9	1 2 3	2 5 8	3 6 9	1 4 7	2 6 7	3 4 8	1 5 9	2 4 9	3 5 7	1 6 8
1 2 3	4 5 6	7 8 9	1 4 7	2 5 8	3 6 9	1 5 9	2 6 7	3 4 8	1 6 8	2 4 9	3 5 7
7 8 9	1 2 3	4 5 6	3 6 9	1 4 7	2 5 8	3 4 8	1 5 9	2 6 7	3 5 7	1 6 8	2 4 9
4 5 6	7 8 9	1 2 3	2 5 8	3 6 9	1 4 7	2 6 7	3 4 8	1 5 9	2 4 9	3 5 7	1 6 8

Figure 4.25: A $(12 \times 12)/3$ BSLR for 9 treatments obtained by making 4 copies of a $(3 \times 12)/3$ BSLR for 9 treatments

can be obtained. This involves permuting the rows and columns of each Latin square. A permutation of the rows of each Latin square leads to non-isomorphic designs, since each row of the Latin square is only a part of the entire row. However, permuting the columns of each Latin square leads to isomorphic designs, since each column of the Latin square is also a column in the design.

Moreover, for any basic design whose construction involves arrangement of two or more Latin squares, a random ordering of the Latin squares within its array leads to designs that are isomorphic.

Chapter 5

Non-balanced Semi-Latin Rectangles with Block Size Two

5.1 Introduction

This chapter is concerned with semi-Latin rectangles whose quotient block designs (QBDs) are basically not BIBDs and whose row-column intersections, each contains exactly 2 treatments. We give constructions for semi-Latin rectangles (SLRs) of this class whose QBDs are regular-graph designs (RGDs). We consider SLRs of this kind for those sizes that a balanced semi-Latin rectangle (BSLR) fails to exist (if they exist). This is so, since RGDs are, particularly, for large number of blocks, known to contain a design with optimality properties regarding the commonly used criteria, the A-, D- and E-criteria (if any exist). SLRs whose QBDs are RGDs are known as regular-graph semi-Latin rectangles (RGSLRs): see Bailey and Monod (2001). Different RGSLRs may have different values of any given optimality criterion. Following John and Mitchell (1977), we are assuming that the optimal designs would be found in the RGDs.

RGDs are close to balanced, in the sense that treatment concurrences differ by at most 1 in absolute terms. However, they are not the same as nearly balanced designs, which were defined by Cheng and Wu (1981). We note that nearly balanced designs are not equireplicate and so cannot occur as a QBD of a SLR. We consider, in particular, SLRs whose QBDs are BIBD-extended RGDs which would give designs with good statistical properties: see Cakiroglu (2018) for discussions on BIBD-extended RGDs. We consider cases where the number of treatments, v is even and also when it is odd. When v is even, we extend the constructions given in Bailey and Monod (2001) to obtain larger designs. When v is odd, we give construction for a basic design which is then extended to obtain constructions for larger designs.

Theorem 5.1.1. *Let Δ_i, where $i = 1, 2$ denote 2 semi-Latin rectangles with the treatment set $V = \{1, 2, ..., v\}$. Suppose Δ_1 is an $(h \times p)/k$ RGSLR and Δ_2 is an $(h \times p')/k$ BSLR. Then the design obtained by putting Δ_1 and Δ_2, side by side, in an $h \times p''$ array is an*

$(h \times p'')/k$ RGSLR for v treatments, where $p'' = p + p'$.

Proof. Let Ξ denote the resulting design on putting Δ_1 and Δ_2 in the array; and let n_c and n_r denote the number of times each treatment appears in each column and each row of Ξ. Since Δ_i, where $i = 1, 2$ are semi-Latin rectangles, then for all Δ_i, there exist $n_c^{(i)}$ and $n_r^{(i)}$ both positive integers such that each $\alpha \in V$ appears $n_c^{(i)}$ and $n_r^{(i)}$ times, per column and row, respectively, in Δ_i.

Notice that since the array contains h rows and p'' columns, where $p'' = p + p'$, then $p' = p'' - p$. Furthermore, since Δ_1 and Δ_2 are put side by side in the array, it follows that each column in Δ_1 is a column in Ξ, and each column in Δ_2 is also a column in Ξ. Hence, the combination of columns in Δ_1 and Δ_2 make Ξ. Notice also that for all $i = 1, 2$, $n_c^{(i)} = kh/v$, which is a positive integer since $v|kh$, as Δ_1 and Δ_2 are SLRs and this is identical to n_c. Hence n_c is a positive integer. Similarly, each row in Ξ is a combination of corresponding rows in Δ_1 and Δ_2, and the combination of all corresponding rows in Δ_1 and Δ_2 make Ξ. Hence, $n_r = n_r^{(1)} + n_r^{(2)}$, which is a sum of two positive integers, hence a positive integer. Thus Ξ is a SLR.

Now, since Δ_1 is a RGSLR, then there are 2 distinct treatment concurrence counts. Let λ_1 and λ_2 denote these concurrence numbers, then $|\lambda_2 - \lambda_1| = 1$. Furthermore, since Δ_2 is a BSLR, denote by λ, the unique treatment concurrence number. For all $\alpha, \beta \in V$, α and β appear together in either λ_1 or λ_2 blocks in Δ_1 and in λ blocks in Δ_2. Since the QBD of Ξ comprises the set of blocks from both Δ_1 and Δ_2, then α and β concur in either $\lambda_1 + \lambda$ or $\lambda_2 + \lambda$ blocks in Ξ.

Let $\lambda_1^\dagger = \lambda_1 + \lambda$ and $\lambda_2^\dagger = \lambda_2 + \lambda$. Notice that $|\lambda_2^\dagger - \lambda_1^\dagger| = |\lambda_2 - \lambda_1| = 1$. Since each of Δ_1 and Δ_2 has the same set of treatments, V, where $|V| = v$, it follows that Ξ is an $(h \times p'')/k$ RGSLR for v treatments, where $p'' = p + p'$. ∎

Comment. Each column of Δ_i, for all $i = 1, 2$ constitute a column in Ξ. Furthermore, each row in Ξ is a combination of corresponding rows from Δ_1 and Δ_2.

5.2 Construction when v is even

Definition 5.2.1. Let $S_i \subset \mathbb{Z}_{2m}$, where $i = 1, \ldots, m$ and $|S_i| = 2$, for all i. We regard $\mathbb{Z}_{2m}$ to be the set $\{1, \ldots, 2m\}$. Let $\{S_i\}_{i=1}^m$ constitute a partition of $\mathbb{Z}_{2m}$. Define $S_i = \{x_i, y_i\}$, where x_i and y_i are such that

$$\pm(y_i - x_i) = \begin{cases} \pm i & \text{if } i < m, \\ m \text{ twice} & \text{if } i = m. \end{cases}$$

Then $\{S_i\}_{i=1}^m$ is called a starter for the cyclic group formed by $\mathbb{Z}_{2m}$ under addition in Bailey and Monod (2001).

1	6	4	5	5	1	4	6	2	3
2	5	2	6	3	6	5	3	1	4
3	4	3	1	4	2	1	2	5	6

Figure 5.1: A BTD(6)

Notice that $-i = 2m - i$ such that $-m = m$. Hence $\bigcup_{i=1}^{m} \{\pm(y_i - x_i)\}$ contain m twice (which come from S_m) and every other element of $\mathbb{Z}_{2m} \backslash \{2m\}$ exactly once (where each comes from a unique S_i, for $i < m$).

As an illustration, let $m = 4$. Then the sets $\{6, 7\}, \{1, 3\}, \{2, 5\}$ and $\{4, 8\}$ constitute a starter in $\mathbb{Z}_8$. Notice that the differences between the elements of these sets are $\pm 1, \pm 2, \pm 3$ and 4(twice), respectively, where $4 = -4$ in $\mathbb{Z}_8$.

Definition 5.2.2. Let $V = \{1, \ldots, 2m\}$ denote a set of teams available for a league tournament which is to consist of $2m - 1$ rounds, where each round is to be played on m grounds. Let the league schedule form an $m \times (2m - 1)$ array whose cells are constituted by the $\binom{2m}{2} = m(2m-1)$ distinct pairs of teams from V such that each pair of teams plays once, overall and each team plays once in each round and at most twice on each ground. Then the league schedule is said to constitute a balanced tournament design (BTD) for the $2m$ teams, denoted BTD($2m$): see Anderson (1997, Chapter 10).

We give an example of a balanced tournament design for 6 teams (BTD(6)): see Figure 5.1. Notice that there are 5 rounds (where each round corresponds to a column) and 3 grounds (where each ground corresponds to a row), that is, $m = 3$.

Definition 5.2.3. Let $\Omega = \{\Lambda_i\}_{i=1}^{m-1}$ denote a complete set of $m - 1$ mutually orthogonal Latin squares (MOLSs) of order m, where m is a prime or prime power. For all $i = 1, 2, ..., m - 1$, let S_i denote the set of symbols in Λ_i, where $S_i \cap S_{i'} = \emptyset$, for all $i \neq i'$. Let $\Gamma = \{\Lambda^{(1)}, \Lambda^{(2)}, . . ., \Lambda^{(k)}\}$ be a subset of Ω, where the cardinality of Γ, $|\Gamma| = k$ and $k \in \{2, 3, ..., m - 1\}$. If $\Lambda^{(1)}, \Lambda^{(2)}, . . ., \Lambda^{(k)}$ are superimposed and the superimposition is regarded as having $v = mk$ symbols (treatments), rather than k treatment factors with m levels each, then the resulting design is said to be an $(m \times m)/k$ Trojan square.

We give an example of a $(4 \times 4)/2$ Trojan square in Figure 5.2. Notice that the symbol sets, S_1 and S_2 of the two superimposed orthogonal Latin squares that make the Trojan square are $S_1 = \{1, 3, 5, 7\}$ and $S_2 = \{2, 4, 6, 8\}$, which are disjoint.

Definition 5.2.4. Let X be a non-empty set, and f a function such that $f : X \rightarrow$

1 2	3 4	5 6	7 8
3 6	1 8	7 2	5 4
5 8	7 6	1 4	3 2
7 4	5 2	3 8	1 6

Figure 5.2: A $(4 \times 4)/2$ Trojan square

X. Suppose f is a bijection, that is, one-to-one and also onto, then f is said to be a permutation of X.

Remark. In general, suppose X and Y are 2 non-empty sets, and g a function such that $g : X \to Y$. Then g is said to be one-to-one if for all $x_1, x_2 \in X$, $g(x_1) = g(x_2)$ means that $x_1 = x_2$. Similarly, g is said to be onto if for all $y \in Y$, there exists $x \in X$ such that $g(x) = y$.

Bailey and Monod (2001) give constructions for RGSLRs of the size $h = n$, $p = 2n$ and $k = 2$ for $v = 2n$, where $2 \leq n \leq 10$, which is clearly, an $(n \times 2n)/2$ RGSLR for $2n$ treatments. These authors employ two methods in their construction, *viz*, the use of starter and also balanced tournament design for $2n$ teams (BTD($2n$)), obtained via exchange of a pair of rows in some columns of a cyclic tournament schedule for $2n$ teams (a balanced resolvable incomplete-block design for $2n$ treatments and block size 2) and then adding an extra column to the BTD($2n$), where the entries in each cell of the added column correspond to the pair of treatments (teams) required to make each treatment appear twice in the corresponding row. They show that a starter exists for the group $\mathbb{Z}_{2n}$ if and only if $n \equiv 0$ (or 1) mod 4. They also show that if a SLR can be derived from a cyclic tournament schedule via exchange of a pair of rows in some columns, then $n \not\equiv 2$ mod 3. Furthermore, they give that their constructions work for all values of $n \not\equiv 2$ (or 11) mod 12.

Our construction extends theirs, for some values of $h = m$ (where m is identical to n in the construction given by Bailey and Monod (2001) indicating the number of rows) to give constructions for RGSLRs with $v = 2m$ treatments and $k = 2$ treatments per block where the number of columns is $p = 2m + m(2m - 1) = m(2m + 1)$ via putting a RGSLR (consisting of $2m$ treatments in m rows and $2m$ columns) obtained via the construction by Bailey and Monod (2001) and a BSLR (consisting of m rows and $m(2m-1)$ columns) in an $m \times m(2m+1)$ array, when both designs exist. We also obtain constructions for RGSLRs with $h = m$ rows and $p = 2m + m\theta = m(\theta + 2)$ columns, where $\theta = 1, 2, 4$. In particular, if $\theta = 1$, we put a RGSLR for $2m$ treatments (having m rows and $2m$ columns) and a Trojan square with $2m$ treatments in an $m \times 3m$ array. The Trojan square is obtained by superimposing 2 orthogonal Latin squares of order m, where one consists of odd number symbols and the other consists of even number symbols, when the construction involves the use of starter; but the symbol set of the 2 orthogonal Latin squares that make the Trojan is adjusted when we use an alternative construction that involves a BTD($2m$). Similarly, if $\theta = 2$, we put an $(m \times 2m)/2$ RGSLR for $2m$ treatments and another RGSLR of the same size and same set of treatments (obtained by a permutation of the treatments within each cell of the parent RGSLR) in an $m \times 4m$ array. Furthermore, if $\theta = 4$, we make 2 new RGSLRs, each consisting of the same set of treatments and also being of the same size as the parent RGSLR by applying a different permutation each time to the treatments within each cell of the parent RGSLR and then putting the parent RGSLR with the 2 new RGSLRs in an $m \times 6m$ array.

Moreover, more designs are derived from the basic constructions by creating an array of appropriate size and then rearranging the component designs within the array.

We note that the constructions given by Bailey and Monod (2001) produce designs whose QBDs consist of an RGD with an extension, which is a BIBD; and whose treatment concurrence counts are 1 and 2. Hence, the QBD of their designs are BIBD-extended RGDs. It follows that each construction presented here also produce designs whose QBDs are BIBD-extended.

5.3 Construction for designs of the class $(m \times m(2m + 1))/2$, where $v = 2m$

We extend the construction of $(m \times 2m)/2$ RGSLR for $2m$ treatments given in Bailey and Monod (2001) by putting the RGSLR and a BSLR on the same set of treatments (obtained via the method of Chapter 3) in an $m \times m(2m + 1)$ array, when both designs exist. We note that m in this construction is identical to n in the construction given in Bailey and Monod (2001).

5.3.1 Construction procedure

1. Obtain an $(m \times 2m)/2$ RGSLR for $v = 2m$ treatments.

2. Obtain an $(m \times m(2m - 1))/2$ BSLR on the same set of $2m$ treatments.

3. Create an $m \times m(2m + 1)$ array and put the BSLR and the RGSLR.

Remark. The order which the constituent designs in steps 1 and 2 are put in the array is immaterial; it can be in any order. A different arrangement of these designs within an array of the same size as in step 3 produces another RGSLR of the same size as the former. The construction produces an $(m \times m(2m + 1))/2$ BSLR-extended RGSLR for $2m$

2	5	2	6	3	6	5	3	1	4	4	1
1	6	4	5	5	1	4	6	2	3	3	2
3	4	3	1	4	2	1	2	5	6	5	6

Figure 5.3: Δ_1: A $(3 \times 6)/2$ RGSLR for 6 treatments

treatments as a basic design; that is, the resulting design has a RGSLR part and a BSLR part as its extension such that its QBD is a BIBD-extended RGD with the treatment concurrence counts being $\lambda+1$ and $\lambda+2$, where λ is the constant concurrence counts from the BSLR.

If the array in step 3 is transposed, then we obtain another class of designs which are $(m(2m+1) \times m)/2$ BSLR-extended RGSLRs on the same set of $2m$ treatments.

A larger BSLR-extended RGSLR of size $(m \times m(2+l(2m-1)))/2$ (where $l > 1$) which contains $m(1-2m+l(2m-1))$ or equivalently, $m(l-1)(2m-1)$ more columns than the basic design can be obtained from the basic design by making $(l-1)$ extra copies of the BSLR obtained in step 2 and then putting these extra copies of the BSLR with the basic design in an $m \times m(2+l(2m-1))$ array. The resulting design has treatment concurrence counts being $1+l\lambda$ or $2+l\lambda$. Moreover, by also changing the array size to $m(2+l(2m-1)) \times m$, the resulting designs are of the class $(m(2+l(2m-1)) \times m)/2$.

Example 5.3.1. Let $v = 6$. Then we can obtain a $(3 \times 21)/2$ BSLR-extended RGSLR by putting a $(3 \times 6)/2$ RGSLR and a $(3 \times 15)/2$ BSLR in a 3×21 array.

Notice that $m = 3$. Hence l needs to be 1 to achieve the required number of columns, 21, which is identical to $m(2+l(2m-1))$.

Let Δ_1 and Δ_2 denote the $(3 \times 6)/2$ RGSLR and $(3 \times 15)/2$ BSLR, respectively. We adapt Δ_1 from Bailey and Monod (2001) and Δ_2 from Chapter 3 of this thesis. We obtain Δ_1 and Δ_2 to be as shown in Figures 5.3 and 5.4, respectively. If Ξ_1 denote the BSLR-extended RGSLR obtained by putting Δ_1 and Δ_2 in a 3×21 array, then we obtain Ξ_1 to be the design in Figure 5.5.

Remark. Notice that the treatment concurrence counts in the design, Ξ_1 shown in Figure 5.5 are 4 and 5, which are the sum of concurrence counts from Δ_1 and Δ_2. In particular, the treatment concurrence counts in Δ_2 is 3.

Suppose a $(3 \times 51)/2$ BSLR-extended RGSLR for the same number of treatments, 6 is required, then this can be obtained by making 2 extra copies of Δ_2 and putting them with Ξ_1 in a 3×51 array. This is so, since $l = 3$ as $m(2+l(2m-1)) = 51$. Moreover, the resulting design has treatment concurrence counts to be 10 and 11.

1 6	2 5	3 4	2 6	3 1	4 5	3 6	4 2	5 1	4 6	5 3	1 2	5 6	1 4	2 3
3 4	1 6	2 5	4 5	2 6	3 1	5 1	3 6	4 2	1 2	4 6	5 3	2 3	5 6	1 4
2 5	3 4	1 6	3 1	4 5	2 6	4 2	5 1	3 6	5 3	1 2	4 6	1 4	2 3	5 6

Figure 5.4: Δ_2: A $(3 \times 15)/2$ BSLR for 6 treatments

2 5	2 6	3 6	5 3	1 4	4 1	1 6	2 5	3 4	2 6	3 1	4 5	3 6	4 2	5 1	4 6	5 3	1 2	5 6	1 4	2 3
1 6	4 5	5 1	4 6	2 3	3 2	3 4	1 6	2 5	4 5	2 6	3 1	5 1	3 6	4 2	1 2	4 6	5 3	2 3	5 6	1 4
3 4	3 1	4 2	1 2	5 6	5 6	2 5	3 4	1 6	3 1	4 5	2 6	4 2	5 1	3 6	5 3	1 2	4 6	1 4	2 3	5 6

Figure 5.5: Ξ_1: A $(3 \times 21)/2$ BSLR-extended RGSLR for 6 treatments

Example 5.3.2. Let $v = 8$. Then we obtain a $(4 \times 36)/2$ BSLR-extended RGSLR by putting a $(4 \times 8)/2$ RGSLR and a $(4 \times 28)/2$ BSLR in an array of size 4×36.

Notice in this example that $m = 4$ and $l = 1$. Let Λ_1 and Λ_2 denote the $(4 \times 8)/2$ RGSLR and $(4 \times 28)/2$ BSLR. Denote by Ξ_2 the $(4 \times 36)/2$ BSLR-extended RGSLR under construction. Then we obtain Λ_1, Λ_2 and Ξ_2 to be the designs shown in Figures 5.6, 5.7 and 5.8, respectively.

1 2	2 3	3 4	4 5	5 6	6 7	7 8	8 1
4 6	5 7	6 8	7 1	8 2	1 3	2 4	3 5
5 8	6 1	7 2	8 3	1 4	2 5	3 6	4 7
3 7	4 8	5 1	6 2	7 3	8 4	1 5	2 6

Figure 5.6: Λ_1: A $(4 \times 8)/2$ RGSLR for 8 treatments

1 8	2 7	3 6	4 5	2 8	3 1	4 7	5 6	3 8	4 2	5 1	6 7	4 8	5 3	6 2	7 1	5 8	6 4	7 3	1 2	6 8	7 5	1 4	2 3	7 8	1 6	2 5	3 4
4 5	1 8	2 7	3 6	5 6	2 8	3 1	4 7	6 7	3 8	4 2	5 1	7 1	4 8	5 3	6 2	1 2	5 8	6 4	7 3	2 3	6 8	7 5	1 4	3 4	7 8	1 6	2 5
3 6	4 5	1 8	2 7	4 7	5 6	2 8	3 1	5 1	6 7	3 8	4 2	6 2	7 1	4 8	5 3	7 3	1 2	5 8	6 4	1 4	2 3	6 8	7 5	2 5	3 4	7 8	1 6
2 7	3 6	4 5	1 8	3 1	4 7	5 6	2 8	4 2	5 1	6 7	3 8	5 3	6 2	7 1	4 8	6 4	7 3	1 2	5 8	7 5	1 4	2 3	6 8	1 6	2 5	3 4	7 8

Figure 5.7: Λ_2: A $(4 \times 28)/2$ BSLR for 8 treatments

1 2	2 3	3 4	4 5	5 6	6 7	7 8	8 1	1 8	2 7	3 6	4 5	2 8	3 1	4 7	5 6	3 8	4 2	5 1	6 7	4 8	5 3	6 2	7 1	5 8	6 4	7 3	1 2	6 8	7 5	1 4	2 3	7 8	1 6	2 5	3 4
4 6	5 7	6 8	7 1	8 2	1 3	2 4	3 5	4 5	1 8	2 7	3 6	5 6	2 8	3 1	4 7	6 7	3 8	4 2	5 1	7 1	4 8	5 3	6 2	1 2	5 8	6 4	7 3	2 3	6 8	7 5	1 4	3 4	7 8	1 6	2 5
5 8	6 1	7 2	8 3	1 4	2 5	3 6	4 7	3 6	4 5	1 8	2 7	4 7	5 6	2 8	3 1	5 1	6 7	3 8	4 2	6 2	7 1	4 8	5 3	7 3	1 2	5 8	6 4	1 4	2 3	6 8	7 5	2 5	3 4	7 8	1 6
3 7	4 8	5 1	6 2	7 3	8 4	1 5	2 6	2 7	3 6	4 5	1 8	3 1	4 7	5 6	2 8	4 2	5 1	6 7	3 8	5 3	6 2	7 1	4 8	6 4	7 3	1 2	5 8	7 5	1 4	2 3	6 8	1 6	2 5	3 4	7 8

Figure 5.8: Ξ_2: A $(4 \times 36)/2$ BSLR-extended RGSLR for 8 treatments

Remark. Notice that the treatment concurrence counts in Ξ_2 are 5 and 6.

5.4 Construction for designs of the class $(m \times 3m)/2$, where $v = 2m$

The designs considered in this section are a special case of RGSLRs of the class $(m \times m(\theta + 2))/2$, where $v = 2m$ and $\theta = 1$. Our construction for this class of designs involves putting an $(m \times 2m)/2$ RGSLR for $2m$ treatments and an $(m \times m)/2$ Trojan square (obtained by superimposing 2 orthogonal Latin squares of order m) in an $m \times 3m$ array.

We utilize the constructions in Bailey and Monod (2001) that use starter (and BTD($2n$) in the alternative) in conjunction with a Trojan square. Hence there need to exist a starter in $\mathbb{Z}_{2m}$ and there also need to exist at least a pair of MOLSs of order m, which guarantees the existence of a Trojan square. Furthermore, for the alternative construction, apart from a Trojan square existing, the condition for deriving a SLR from a cyclic tournament schedule via exchange of a pair of blocks in some columns must be satisfied: see Bailey and

Monod (2001). For the method involving starter, the symbol set of one of the 2 orthogonal Latin squares that constitute the Trojan square comprises odd number symbols while the symbols of the other Latin square are precisely, even number symbols; but the symbol sets for the 2 Latin squares are modified to suit the construction when we use the alternative approach. We note that for every finite order, m, where $m \neq 2$ or 6, there exists a Latin square with at least one orthogonal mate: see Bose et al. (1960).

5.4.1 Construction via starter

A starter exists for the group $\mathbb{Z}_{2m}$ if and only if $m \equiv 0$ (or 1) mod 4: see Bailey and Monod (2001).

Suppose a starter exists in $\mathbb{Z}_{2m}$. Suppose further that there exists at least a pair of MOLSs of order m. The construction involves obtaining the m starter sets and putting the entries of these sets in the m cells in the initial column of an $m \times 3m$ array (one set to a cell) thereby forming the initial blocks for the m rows, which are then developed, cyclically, via successive addition of 1 modulo $2m$ to the entries in the cells to generate the first $2m$ columns. The remaining m columns are obtained by adjoining a Trojan square obtained by superimposing two MOLSs whose symbol sets are constituted by the odd number treatments and even number treatments, respectively.

We proceed to give an algorithmic procedure for constructing the design.

5.4.2 Algorithmic procedure for constructing the design via starter

1. Label the treatments 1, 2, ..., $2m$.

2. Partition the treatment set into m pairs, that is, m 2-subsets, viz $\{x_1, y_1\}$, $\{x_2, y_2\}$, . . ., $\{x_m, y_m\}$ such that the differences (reduced modulo $2m$) between these pairs of treatments are ± 1, ± 2, . . ., $\pm m$, respectively, thereby forming a starter in the cyclic group $\mathbb{Z}_{2m}$.

3. Create an $m \times 3m$ array and label its rows $i = 1, 2, ..., m$ and its columns $j = 1, 2, ..., 2m, 2m + 1, ..., 3m$.

4. For all $i = 1, 2, ..., m$, insert in the cell in position $(i, 1)$ of the array (that is, the cell in row i and column 1), the 2-subset, $\{x_i, y_i\}$ obtained in step 2.

5. For all $i = 1, 2, ..., m$, develop the block in position $(i, 1)$, which contains $\{x_i, y_i\}$, cyclically, via successive addition of 1 (mod $2m$), thereby generating the block in position (i, j), for all $j = 2, 3, ..., 2m$.

6. Make a Trojan square via superimposition of two orthogonal Latin squares of order m, where in one of the Latin squares the symbols are the odd number treatments, while the symbols of the other Latin square are the even number treatments.

7. Insert the Trojan square obtained in 6. to fill columns $2m + 1$ to $3m$ of the array created in 3.

Remark. The pairs of treatments in the m 2-subsets that constitute the starter, where the differences between the treatments in these sets reduced modulo $2m$ are ± 1, ± 2, . . ., $\pm m$, satisfy the property that any pair with the differences $\pm\delta$, where $\delta \in \{1, 2, ..., m\}$ correspond to a pair of vertices of a regular $2m$-gon which are at distance δ, when the set of treatments are identified with the vertices of the $2m$-gon.

If the array size in step 3 is modified to be $3m \times m$ and the roles of rows and columns are exchanged such that the Trojan square now appears between rows $2m + 1$ and $3m$. Then the resulting design is a $(3m \times m)/2$ RGSLR.

Furthermore, if the Trojan square is inserted first into the array before the other component design in either of the situations described above, it produces another design of the same size as the former. For each situation, the design obtained by swapping the order of the two constituent designs is isomorphic to the original design.

The construction described above will always produce an $(m \times 3m)/2$ RGSLR for $2m$ treatments if a starter exists and there are at least 2 MOLSs, each being of order m, provided m is an even number.

Theorem 5.4.1. *Let Γ denote an $(m \times 2m)/2$ RGSLR for $2m$ treatments obtained via a starter for $\mathbb{Z}_{2m}$ by a cyclic development of the blocks in the initial column containing the m 2-subsets of the starter. Let Δ be an $(m \times m)/2$ Trojan square on the same set of treatments as Γ, where one of the constituent Latin squares of Δ consists of odd number symbols and the other Latin square consists of even number symbols. If Γ and Δ are put in an $m \times 3m$ array, then the resulting design is an $(m \times 3m)/2$ RGSLR for $2m$ treatments, if and only if m is even.*

Proof. Let $V = \{1, 2, ..., m-1, m, m+1, ..., 2m-1, 2m\}$ denote the symbol set of Γ and Δ.

Since Δ is a SLS, then each treatment appears exactly once in each row and in each column. Similarly, in each column of Γ, each treatment appears once but appears twice in each row. Now, each row of the design resulting from putting Γ and Δ in the $m \times 3m$ array is constituted by the cells from corresponding rows of Γ and Δ, hence each treatment appears in each row of the resulting design 3 times (a positive integer number of times) which is the sum of the number of times it appears in each row of Γ and Δ combined. Moreover, the columns in the overall design is constituted by the overall columns in both Γ and Δ; each column in Γ is a column in the overall design, and each column in Δ is also a column in the overall design. Since each treatment appears the same number of times (exactly once) in each column of Γ and Δ, then it also appears exactly once (a positive integer number of times) in each column of the overall design. Hence each treatment appears a constant number of times in each row and similarly, a constant number of times in each column, which makes the resulting design a SLR.

Let L_1 and L_2 denote the two orthogonal Latin squares, each of order m that make Δ with respective symbol sets V_1 and V_2. Partition V into V_1 and V_2 such that $|V_1| = |V_2| = m$. Suppose $V_1 = \{1, 3, ..., 2m-1\}$, the set of odd number symbols, such that $V_2 = \{2, 4, ..., 2m\}$, the set of even number symbols. Since the L_1 and L_2 are orthogonal and V_1 and V_2 are disjoint, then Δ consists of the m^2 distinct pairs of symbols from V_1 and V_2, each appearing exactly once.

Notice that the m^2 pairs of symbols in Δ are the result of the cross product of the sets, V_1 and V_2. Denote this cross product by $V_1 \times V_2$. For all $(i, i') \in V_1 \times V_2$, let $d(i, i')$ denote the distance between i and i', where

$$d(i, i') = \begin{cases} |i' - i| & \text{if } |i' - i| \leq m, \\ 2m - |i' - i| & \text{if } |i' - i| > m. \end{cases}$$

Let $D = \{d(i, i')\}$ denote the set of distances between each pair, (i, i') of symbols. Then $D = \{1, 3, ..., d_{\max}\}$, which consists of only the odd number symbols, since each pair of symbols in Δ involves an odd and even number symbols. Furthermore, $d_{\max} = m-1$, if m is even. Notice that if m is even, then $m-1$ is odd such that for all $i \in V_1$, $i \pm m - 1 \in V_2$, hence the pair $(i, i') = (i, i \pm m - 1)$ constitutes a cell in Δ. Similarly, if V_1 is the set of even number symbols and V_2 consists of odd number symbols, then for all $i \in V_1$, $i \pm m - 1 \in V_2$, hence the pair $(i, i') = (i, i \pm m - 1)$ also constitutes a cell in Δ. Moreover, let $|D| = \omega$. Then $\omega = m/2$

Since the number of cells/pairs of symbols in Δ is m^2; if m is even, then $\omega \mid m^2$ such that Δ consists of $m^2/\omega = 2m$ pairs of distinct symbols for each unique distance, $d^* \in D$. To show that this is true, we proceed as follows:

For all $d^* \in D$, there exists $i \in V_1$ and $i \pm d^* \in V_2$ such that the pairs $(i, i \pm d^*) = (i, i + d^*), (i, i - d^*)$ in Δ constitute treatment symbols that are at a distance, d^*, where the addition and subtraction are performed modulo $2m$. Notice that what distinguishes the two pairs are their second entries, $(i + d^*)$ and $(i - d^*)$.

Since m being even implies $d_{\max} = m - 1$, then $\exists\ d^* = m - 1 \in D$ such that for each i the second entries are $i + m - 1$ and $i - m + 1$, which are distinct since $-1 = 2m - 1 \neq 1$, as $m \neq 1$. Hence, there are two distinct pairs for each i, and thus $2m$ distinct pairs overall, for all the i's. Similarly, if $d^* = m - 1 - s$, where $s = 2, 4, ..., m-2$, then for each i, the second entries become $i + m - 1 - s$ and $i - m + 1 + s$, which are also distinct, since $-1 - s = 2m - (1 + s) \neq 1 + s$. Thus, there are also two distinct pairs for each i and consequently, $2m$ distinct pairs, overall for all i's.

Hence, if m is even, then each distance, $d^* \in D$ has the same number of pairs, $2m$ of treatment symbols associated with it. The overall design, in this case, has no pair in the part constituted by Δ for which $d^* = m$, but there are m such distinct pairs in row m, say, in the part constituted by Γ, where each pair concurs twice there. This is so since one of the started 2-subsets used to generate Γ contains a pair of treatments whose differences (modulo $2m$) is m twice while the treatment pairs that make the remaining $m-1$ 2-subsets are such that their differences are $\pm u$, for all $u = 1, 2, ..., m-1$. Hence, the sum of concurrences for each pair at distance m in the overall design is 2. Similarly, for all $d^* = y \in D$, where $y \leq m-1$, each pair concurs once in the part constituted by Δ, and also once in row y, say, in the part constituted by Γ, where $y = 1, 3, ..., m-1$, making the sum of concurrences for each pair of symbols at an odd distance ($\leq m-1$) in the design to be 2, also. However, each pair for which the distance is an even number, w, say, where w is less than m does not concur at all in the part constituted by Δ, but concurs only once in row w, say, in the part constituted by Γ, where $w = 2, 4, ..., m-2$, hence overall, concurs once. Thus, the number of concurrences for any pair of treatment symbols in the overall design is either 1 or 2, making it a RGSLR.

■

Comment. If m is odd, the construction would not give a RGSLR, which is seen as follows:

In this case, $d_{\max} = m$. Notice that if m is odd, then for all $i \in V_1$, $i \pm m \in V_2$, for situations whether V_1 consists of odd number symbols (which implies V_2 comprises even number symbols) or V_1 comprises even number symbols (which implies V_2 comprises odd number symbols). Hence the pair $(i, i') = (i, i \pm m)$ constitutes a cell in Δ. By letting $|D| = \eta$. Then $\eta = (m+1)/2$. We note that $\frac{(m+1)}{2} = \frac{(m-1)}{2} + 1$.

Bearing in mind that there are m^2 cells/pairs of symbols in Δ; if m is odd, then $\eta \nmid m^2$ and there are $2m$ pairs of symbols for each $d^* < d_{\max}$, and m pairs for $d^* = d_{\max}$. This can be seen as follows:

For all $d^* \in D$, there exists $i \in V_1$ and $i \pm d^* \in V_2$ such that the pairs $(i, i \pm d^*) = (i, i + d^*), (i, i - d^*)$ in Δ are treatment symbols that are at a distance, d^*, where the addition and subtraction are performed modulo $2m$. As before, what distinguishes the two pairs are their second entries, $(i + d^*)$ and $(i - d^*)$.

If m is odd and consequently, $d_{\max} = m$, then $\exists\, d^* = m \in D$ such that for each i the second entries are thus $i + m$ and $i - m$, which are identical, since $m = -m \pmod{2m}$. Now, suppose $d^* = m - l$, where $l = 2, 4, ..., m-1$. Then for each i, the second entries become $i + m - l$ and $i - m + l$, which are distinct, since $-l = 2m - l \neq l$, for all l. Hence, in Δ; for odd m, $\exists\, d^* = m \in D$, with only one pair for each $i \in V_1$ (since its second entries in V_2 are the same), and consequently, m pairs overall, for all the i's. Similarly, for each $d^* < m$, there are two distinct pairs for each i (since it has distinct second entries), and overall, $2m$ pairs for all the i's.

Hence, from the foregoing discussions, it is clear that, if m is odd, then there are m distinct pairs of treatment symbols for which $d^* = m$ (where there is only one pair for each i) in the part of the overall design constituted by Δ; and each of these pairs also concurs twice in row m, say, in the part constituted by Γ, which makes the sum of concurrences for each pair of symbols in the overall design with this property to be 3, while it is 2 for every other pair with $d^* \in D \backslash \{m\}$, and 1 for those pairs with $d^{*'} \notin D$, which are those with

1 2	2 3	3 4	4 5	5 6	6 7	7 8	8 1	1 2	3 4	5 6	7 8
5 7	6 8	7 1	8 2	1 3	2 4	3 5	4 6	3 6	1 8	7 2	5 4
3 6	4 7	5 8	6 1	7 2	8 3	1 4	2 5	5 8	7 6	1 4	3 2
4 8	5 1	6 2	7 3	8 4	1 5	2 6	3 7	7 4	5 2	3 8	1 6

Figure 5.9: A $(4 \times 12)/2$ RGSLR for 8 treatments

even distances, x, which appear only in row x, say, in the part of the design constituted by Γ, where $x = 2, 4, ..., m-1$. This means that, in this circumstance, there are three distinct concurrences, 1, 2 and 3 in the overall design. Thus, the design obtained via the above procedure will not be a RGSLR.

Corollary 5.4.1. *The $(m \times 3m)/2$ RGSLR for $2m$ treatments resulting from the construction is such that $m \equiv 0 \bmod 4$.*

Remark. The $3m^2$ blocks in the design consists of $m(m+1)$ blocks from an RGD and $m(2m-1)$ blocks from a BIBD. Hence its QBD is a BIBD-extended RGD. Notice that $2m-1 > m+1$ for all $m > 2$. Hence the BIBD part contributes more blocks to the design than the RGD.

Example 5.4.1. Let $v = 8$. Then $m = 4$. We obtain a $(4\times12)/2$ RGSLR for 8 treatments whose QBD is BIBD-extended as shown in Figure 5.9.

Notice from Figure 5.9 that the set of treatments, $V = \{1, 2, ..., 8\}$. Notice also that the following 2-subsets of V form a starter in $\mathbb{Z}_8$: $\{1, 2\}$, $\{5, 7\}$, $\{3, 6\}$ and $\{4, 8\}$, where the differences between the pairs in each set are ±1, ±2, ±3 and 4, respectively. We note that $4 = -4$ in $\mathbb{Z}_8$.

Comments. By virtue of the construction, the design could be subdivided into two sections, viz, a rectangle and square, as demarcated by the double vertical lines, where the rectangular section is a $(4 \times 8)/2$ RGSLR for 8 treatments and the square section is a $(4 \times 4)/2$ Trojan square obtained by superimposing a Latin square whose treatment set is $V_1 = \{1, 3, 5, 7\}$, the set of odd number treatments on another Latin square which is orthogonal to it and whose treatment set is $V_2 = \{2, 4, 6, 8\}$, the set of even number treatments.

The rectangle contains pairs of treatments at all the distances, $d = 1, 2, 3,$ and 4; and those pairs at an equal distance appear in a single row. For instance, pairs at distance d, appear in row d. For all $d < 4$, each pair concurs once; and for $d = 4$, it concurs twice.

Furthermore, the square contains those pairs of treatments at odd distances, $d = 1$ and 3, only (where each of these pairs appears exactly once); thus, no block contains treatments for even d. As a consequence, the treatment concurrences are 0 and 1 (which is true for any Trojan square), with 0 corresponding to the concurrence counts for pairs of treatments at even distances (the missing pairs), and 1, otherwise. Two rows (rows 1 and 4 in this case), each contains entirely pairs of treatments for a unique d, that is $d = 1$ and 3 for all pairs in rows 1 and 4, respectively; but for he remaining two rows, that is, the 2 middle rows–rows 2 and 3 each contains pairs for each odd value of d, the same number of times, 2.

Hence, overall, in the design, every pair of treatments either concurs once or twice. Those that concur once are those for which d is an even number less than 4, that is, where $d = 2$; while those pairs for which $d = 4$ or an odd number concur twice.

The design has its QBD consisting of 48 blocks, which comprises an RGD and a BIBD. The RGD component consists of 20 blocks (4 distinct blocks in the last row of the rectangle and the 16 blocks from the Trojan square) while the BIBD component consists of 28 blocks (all the remaining blocks in the rectangle).Thus the design is a RGSLR whose QBD is a BIBD-extended RGD with treatment concurrence counts being 1 and 2.

5.4.3 Some Important Notes

If m is odd, then the number of pairs of symbols for each $d^* < d_{\max} = m$ is $2m = \frac{(m^2-m)}{(m+1)/2-1} = \frac{m(m-1)}{(m-1)/2}$. Furthermore, $|V_1 \times V_2| = m^2 = 1(m) + 2m(m-1)/2$ gives the total number of cells/pairs of symbols in Δ. This is true since $m \in D$, and there are m pairs for which $d^* = m$ given by $(i, i \pm m)$, $\forall i \in V_1$ and $i \pm m \in V_2$, where $(i, i \pm m)$ is simply, the unique pair $(i, i+m)$ or $(i, i-m)$, since $(i, i+m) = (i, i-m)$ in $\mathbb{Z}_{2m}$. Notice that $i \pm m \in V_2$, which comprises even number symbols if V_1 consists of odd number symbols (since odd$\pm$ odd = even). Similarly, if V_1 consists of even number symbols, then $i \pm m \in V_2$, which consists of odd number symbols (since even$\pm$ odd = odd). But for even m, $d^* \in D$ is less than m, for each pair, with $d_{\max} = m - 1$ (since in this case, supposing there exists $d^* = m$, then for all $i \in V_1$, $i + m \notin V_2$, whether V_1 consists of odd or even number symbols, as odd $\pm$ even = odd (or even $\pm$ even = even), which is a contradiction in the sense that each pair consists of elements from the same set, V_1. So we cannot have a pair at distance m in the Trojan square.

Furthermore, if m is odd, every partition of V into V_1 and V_2 of sizes m will always lead to the constructed design containing at least one pair with concurrence 3. To show this, we proceed as follows:

Let $n_o(V_l)$ and $n_e(V_l)$ denote the respective number of odd and even number symbols in V_l, where $l = 1, 2$ for any partition of V into V_1 and V_2, with $|V_1|$=$|V_2|$= m, and $n_o(V_l)$,

$n_e(V_l) \in [0, m]$, $\forall\ l = 1, 2$ such that

$$\sum_{l=1}^{2} n_o(V_l) = \sum_{l=1}^{2} n_e(V_l) = m.$$

Similarly, $n_o(V_t) + n_e(V_t) = |V_t| = m$, $n_o(V_l) = n_e(V_{l'})$, and

$$n_e(V_l) = n_o(V_{l'}) \tag{5.1}$$

where $l, l' \in \{1, 2\}$, $l \neq l'$ and $t = l, l'$.

Notice that, if m is odd, every partition of V into V_1 and V_2 leads to $n_o(V_1) \neq n_e(V_1)$. Similarly, $n_o(V_2) \neq n_e(V_2)$. Consequently, by (5.1), $n_o(V_1) \neq n_o(V_2)$.

Theorem 5.4.2. *Let $V = \{1, 2, ..., 2m\}$ be partitioned into V_1 and V_2 of the same size, m. Let Δ^+ denote a Trojan square of side m obtained by superimposing a Latin square, L_1 on its orthogonal mate L_2, and whose symbol sets correspond to V_1 and V_2, respectively. Furthermore, let $D^+ = \{d(i, i') : i \in V_1, i' \in V_2\}$, where $d(i, i')$ is the distance between the vertices corresponding to the treatment symbols i and i' in V. If m is odd, then for any such partition, the number of pairs of treatment symbols in Δ^+ for which $d(i, i') = m$ is $n^* = m - 2b$, where $b = |H|$, $H = \{p \pm m : p \in V_1, p \pm m \in V_1,$ where p is odd and $p \pm m$ is even$\}$.*

Proof. Note that m is odd. Let $A = \{x : x \in V_1,\ x \text{ is odd}\}$, $B = \{y : y \in V_2,\ y \text{ is even}\}$, $C = \{x + m : x \in A\}$, $A' = \{z : z \in V_1,\ z \text{ is even}\}$, and $B' = \{u : u \in V_2,\ u \text{ is odd}\}$. Partition C into D and D', where $D = \{x + m : x \in A, x + m \in B\}$, and $D' = \{x + m : x \in A, x + m \in A'\}$. Furthermore, let $E = \{z + m : z \in A'\}$. In a similar manner, partition E into F and F', where $F = \{z + m : z \in A', z + m \in B'\}$, and $F' = \{z + m : z \in A', z + m \in A\}$.

Hence, $|A| = n_o(V_1) = q$, say, and $|B| = n_e(V_2) = q$, where $q \in [0, m]$. Notice that A' is identical to $V_1 \backslash A$ so that $|A'| = n_e(V_1) = m - |A| = m - q$. Similarly, B' is identical to $V_2 \backslash B$ so that $|B'| = n_o(V_2) = m - |B| = m - q$. Since D^+ concerns those pairs of symbols that make the cells in Δ^+, then for all $x \in A$ and $y \in B$, there is a cell in Δ^+ containing the pair, (x, y) such that $d(x, y) = d^+ \in D^+$, where $d(x, y)$ is odd, and $y = x \pm d^+$. Also, for all $z \in A'$ and $u \in B'$, there is a cell in Δ^+ containing the pair, (z, u) such that $d(z, u) \in D^+$, where $d(z, u)$ is odd, and $u = z \pm d^+$. Thus, the (x, y) and (z, u) pairs in Δ^+ have the property that $d(x, y), d(z, u)$ is odd. Since m is odd, then $d^+ = m$ is associated with those (x, y) and/or (z, u) pairs in Δ^+ for which $y = x + m = x - m$, and $u = z + m = z - m$, thus making y and u unique pairs for x and z, respectively.

We note that, in general, $q \in [0, m]$. Let $|D| = a$ and $|D'| = b$, where $a \in [0, q]$, $b \in [0, s]$ and $s = \min\{q, m - q\}$. Notice that, $a + b = |C| = |A| = q$.

Let $|F| = c$ and $|F'| = d$, where $c \in [0, m - q]$, and $d \in [0, t]$, where $t = \min\{m - q, q\} = \min\{q, m - q\} = s$. Notice that each of F' and D' involves a correspondence between elements of the same pair of sets, A and A', hence $d = b$.

Now, $c + d = |E| = |A'| = m - q$. Notice that, if $m - q = 0$, then $c = d = 0$. Similarly, if $m - q = m$, then $c = m$ and $d = 0$. For the case that $q \in (0, m)$, if $c = 0$, then $d\ (= m - q) \neq 0$ (since $m - q \in (0, m)$, otherwise a contradiction) and vice versa. Moreover, if $d = q$, then $c\ (= m - 2q) \neq 0$, since m is odd.

Hence, overall, with A as the originating set, there are a corresponding treatment symbols from B (which are precisely, the elements of D) and the remaining $q - a = b$ of them from A' (which are precisely, the elements of D'), thus only the a symbols form pairs at distance m with the elements of A in Δ^+. Similarly, overall, with A' as the originating set, there are c corresponding symbols from B' (which are precisely, the elements of F) and the remaining $m - q - c = d$ of them from A (which are precisely, the elements of F'), hence only the c symbols form pairs at distance m with the elements of A' in Δ^+. Thus, for odd m, and $q \in [0, m]$, $n^* = a + c$.

Since $q = a + b$, and $m - q = c + d$, then $q + (m - q) = (a + b) + (c + d) = m$. Thus, $a + c = m - (b + d)$. Recall that $b = d$. Consequently $a + c = m - 2b$. Hence $n^* = m - 2b$.

■

Corollary 5.4.2. $n^* \in [1, m]$, *and its value is always odd.*

Proof. Since m is odd, and $2b$ is always even, then $n^* = m - 2b \neq 0$, and is odd.

We remind that $b \in [0, s]$, where $s = \min\{q, m - q\}$. The maximum value of n^* is attained if and only if $2b$ attains its minimum value, that is, if b attains its minimum value. Thus n^* is maximum if and only if $b = 0$. Hence the upper bound for $n^* = m$.

Furthermore, since n^* is non-negative, and also, as given in the introductory part of this proof non-zero, for odd m. Then, n^* is strictly positive (that is $n^* > 0$) for odd m. Consequently, the lower bound for n^* is 1, since $n^* \in \mathbb{Z}$.

Hence, since for odd m the upper bound for $n^* = m$, and its lower bound 1, then $n^* \in [1, m]$, and its value is always odd.

■

Corollary 5.4.3. *If m is even, then $n^* = m - (b^\dagger + d^\dagger)$, where $b^\dagger = |S|$, $S = \{g \pm m : g \in V_1, g \pm m \in V_1,$ where both g and $g \pm m$ are odd$\}$; and $d^\dagger = |T|$, $T = \{j \pm m : j \in V_1, j \pm m \in V_1,$ where both j and $j \pm m$ are even$\}$.*

In this case $n^* \in [0, m]$, and it has an even value if it is non-zero.

In conclusion, since by Corollary 5.4.2, if m is odd, n^* is at least 1. Then, Δ^+ contains n^* pairs of symbols (which is at least one and at most m) for which $d(i, i') = m$, where each of these pairs already concurs twice in row m, say, in the part of the design constituted by Γ, thus making the sum of concurrences for each of the n^* pairs to be 3. Hence, it follows that every partition of V into V_1 and V_2 of sizes m will always lead to the constructed design containing at least one pair with concurrence 3.

5.4.4 An alternative construction for $(m \times 3m)/2$ RGSLRs for $v = 2m$ treatments

If m is even, an alternative construction for an $(m \times 2m)/2$ RGSLR for $2m$ treatments which involves the use of a balanced tournament design (BTD($2m$)) via exchange of a pair of blocks in some columns: see Bailey and Monod (2001) can also be utilized in combination with a Trojan square to obtain an $(m \times 3m)/2$ RGSLR for $2m$ treatments in a manner similar to the previous method that involves the use of starter. But this time, the set of symbols V_1 and V_2 of the 2 Latin squares that make the Trojan square are such that each of them consists of those entries in $m/2$ cells (combined) that appear in either of the last two columns of the $(m \times 2m)/2$ RGSLR.

The aforementioned construction for the $(m \times 2m)/2$ RGSLR is based on the cyclic group $\mathbb{Z}_{2m-1}$. Moreover, if a SLR can be derived from a cyclic tournament schedule via exchange of a pair of blocks in some columns, then $m \not\equiv 2 \pmod 3$: see Bailey and Monod (2001).

Put $u = 2m - 1$ and regard $\mathbb{Z}_u$, the set of integers modulo u as $\{1, \ldots, u\}$. Denote the treatment set by $V = \{1, \ldots, u\} \cup \{\infty\}$. Create an array of size $m \times 3m$ and label its rows $i = 1, \ldots, m-1, \infty$ and the columns $j = 1, \ldots, u, \infty, u+2, \ldots, 3m$. Let S_{ij} denote the set of entries in the cell in position (i, j).

We give an algorithmic procedure for constructing the design below

5.4.5 An algorithmic procedure for the alternative construction

1. Create an $m \times 3m$ array and label its rows $i = 1, \ldots, m-1, \infty$ and the columns $j = 1, \ldots, u, \infty, u+2, \ldots, 3m$, where $u = 2m - 1$..

2. For $i = 1, \ldots, m-1$ and $j = 1, \ldots, u$, put $S_{ij} = \{j+i, j-i\}$; and put $S_{\infty j} = \{j, \infty\}$.

3. For $j = 1, \ldots, u-1$ and $i = i^* \in \{2j, -2j\} \cap \{1, \ldots, m-1\}$, exchange S_{i^*j} with $S_{\infty j}$, where i^* is the unique entry in the intersection region of the sets.

4. For $i = 1, \ldots, m-1$, put $S_{i\infty} = \{3i/2, -3i/2\}$; and put $S_{\infty\infty} = \{u, \infty\}$.

5. Make a Trojan square by superimposing 2 Latin squares, each being of order m (from a set of MOLSs) and whose symbol sets, V_1 and V_2, respectively, are the overall entries in any $m/2$ cells (chosen such that V_1 and V_2 are disjoint) in column j^+, where $j^+ = u$ or ∞.

6. Put the Trojan square obtained in step 5 into the remaining section of the array, spanning columns $u+2$ to $3m$.

Remark. The overall design comprises both an $(m \times 2m)/2$ RGSLR and an $(m \times m)/2$ Trojan square, hence can be subdivided into a rectangle and square, where the rectangle corresponds to the $(m \times 2m)/2$ RGSLR and the square corresponds to the Trojan square.

The $(m \times 2m)/2$ RGSLR is formed by the cells in columns 1 to ∞, where the column label ∞ is equivalent to $u+1$, but the symbol ∞ is used for purposes of the construction. Similarly, the cells in columns $u+2$ to $3m$ constitute the Trojan square.

Each pair of treatments that appear as a block in the rectangular section concurs exactly once there except the pairs in columns u and ∞ where each of them concurs twice in the rectangle (in these 2 columns, precisely). The pattern of forming V_1 and V_2 ensures that the entries in any cell in either of columns u and ∞, which constitute the pairs that concur higher in the rectangle do not appear as a block in the Trojan-the square section, since the cells in the Trojan square are constituted by pairs of symbols from the cross product between V_1 and V_2. Hence no pair in columns u and ∞ appears as a block in the square section. Moreover, each pair in the square section also appears exactly once in the rectangle, but there are some pairs in the rectangle which are missing from the square.

Notice that, in the overall design, the rectangular section constitutes $2m^2$ blocks while the square section constitutes m^2 blocks. Furthermore, apart from the $2m$ blocks in columns u and ∞, there are $2m(m-1)$ other blocks in the rectangle. Notice also that $2m(m-1) > m^2$ for all $m > 2$, since $2(m-1) - m > 0$ for all $m > 2$. Notice also that $2m(m-1) - m^2 = m(m-2)$. Hence there are $m(m-2)$ pairs of treatments that appear as a block in the rectangle and not in the square, hence they are precisely the pairs with concurrences 0 in the Trojan and 1 in the overall design. Notice also that there is a total of $m + m^2 = m(m+1)$ pairs with concurrences 2 in the overall design, where m of them are the distinct pairs in columns u and $u+1$ while m^2 of them are from the square. Moreover, $m(m-2) < m(m+1)$ if $m > 0$ and that $m(m-2) = 0$ if $m = 2$ but has a value that is a positive integer for all $m > 2$. Hence when a Trojan square exists, the overall design has $m(m-2)$ (fewer) treatment pairs with concurrences 1 and $m(m+1)$ (more) pairs with concurrences 2.

Thus, in summary, the treatment concurrences are 2 for each pair in columns u and ∞. Similarly, the concurrences are 2 for those pairs in the rectangle that also appear in the square; but it is 1 for those pairs in the rectangle that are missing from the square. Hence the treatment concurrences in the overall design are 1 and 2, hence the construction produces a RGSLR.

The QBD of the constructed design is a BIBD-extended RGD comprising an RGD part formed by the $m(m+1)$ blocks in columns ∞ to $3m$ and a BIBD part formed by the $m(2m-1)$ blocks in columns 1 to u, as the extension. Hence there are more blocks in the BIBD component than the RGD since $m(2m-1) > m(m+1)$ for all $m > 2$.

Example 5.4.2. Let $v = 8$. Then the corresponding $(4 \times 12)/2$ RGSLR whose QBD is BIBD-extended obtained via the algorithmic procedure in section 5.4.5 is as shown in Figure 5.10.

By replacing the treatment symbol ∞ by 8, the design in Figure 5.10 transforms to the design shown in Figure 5.11.

2 7	3 1	3 ∞	4 ∞	6 4	7 5	1 6	5 2	1 3	2 4	5 7	6 ∞
1 ∞	4 7	5 1	6 2	7 3	6 ∞	2 5	3 4	2 7	1 ∞	6 3	5 4
4 5	2 ∞	6 7	7 1	5 ∞	2 3	3 4	1 6	5 ∞	6 7	1 4	2 3
3 6	5 6	4 2	5 3	1 2	1 4	7 ∞	7 ∞	6 4	5 3	2 ∞	1 7

Figure 5.10: An alternative $(4 \times 12)/2$ RGSLR for 8 treatments

2 7	3 1	3 8	4 8	6 4	7 5	1 6	5 2	1 3	2 4	5 7	6 8
1 8	4 7	5 1	6 2	7 3	6 8	2 5	3 4	2 7	1 8	6 3	5 4
4 5	2 8	6 7	7 1	5 8	2 3	3 4	1 6	5 8	6 7	1 4	2 3
3 6	5 6	4 2	5 3	1 2	1 4	7 8	7 8	6 4	5 3	2 8	1 7

Figure 5.11: An alternative $(4 \times 12)/2$ RGSLR for 8 treatments with the treatment symbol ∞ replaced by 8

Comment. Notice that the designs shown in Figures 5.9 and 5.11 are non-isomorphic, that is, neither of them can be obtained from the other by a permutation of its rows, a permutation of its columns, a permutation of its treatments or a combination of more than one of these. The symbol sets of the 2 Latin squares that make the Trojan square component in Figure 5.10 are $V_1 = \{1, 2, 5, 6\}$ and $V_2 = \{3, 4, 7, \infty\}$ obtained by pooling the treatments from 2 distinct cells in either column 7 or column ∞. Note that treatments from any other different pairs of cells can be combined to give V_1 and V_2, provided $V_1 \cap V_2 = \emptyset$.

5.5 Construction for designs of the class $(m \times 4m)/2$, where $v = 2m$

We give construction for $(m \times 4m)/2$ RGSLRs for $2m$ treatments which is another special case of $(m \times m(\theta + 2))/2$ RGSLRs with $2m$ treatments, where $\theta = 2$. The construction is equivalent to putting two $(m \times 2m)/2$ RGSLRs for $2m$ treatments in an $m \times 4m$ array, where the distinct treatment concurrence counts, λ_1 and λ_2 for both constituent SLRs are such that $\lambda_1, \lambda_2 \in \{\lambda, \lambda+1\}$, for $\lambda = 1$ but in one of the SLRs, those treatment pairs with a higher treatment concurrence counts, $\lambda + 1$ in the other SLR has a lower concurrence count, λ in it whilst some treatment pairs which concur a fewer number of times, λ in the other now concur a higher number of times, $\lambda + 1$ in it. We obtain a parent $(m \times 2m)/2$ RGSLR via starter or BTD if the condition for using each approach is satisfied: see Bailey and Monod (2001), and then obtain the other from the parent design .

Let Λ_1 and Λ_2 denote the two constituent RGSLRs that make the design, where Λ_1 is the parent design. If Λ_1 is obtained via starter, then Λ_2 is obtained by first identifying a set of m blocks in row m of Λ_1 containing distinct pairs of treatments (where the differences modulo $2m$ between these pairs of treatments in each of the blocks is m) which are precisely, those pairs of treatments that concur a higher number of times, 2 in Λ_1 and also form a parallel class. Denote this parallel class P_1. Similarly, if Λ_1 is obtained via BTD, then P_1 is constituted by the pairs of symbols in the m cells in either of columns u and ∞, which are the pairs that concur higher (also twice) in this case, where $u = 2m - 1$ and ∞ is used for purposes of construction to label column $u + 1$. Another parallel class, P_2, say, is obtained from P_1 by applying a permutation, α, say, that relabels the treatments in each set of P_1 such that P_2 contains no set in common with P_1. Λ_2 is then obtained by applying α to relabel the treatments within each block of Λ_1, thereby generating the corresponding block in Λ_2. If Λ_1 and Λ_2 are put (in any order) in an $m \times 4m$ array, the resulting design is an $(m \times 4m)/2$ RGSLR for $2m$ treatments.

We give algorithmic procedures for construction based on the 2 methods in sections 5.5.1 and 5.5.2, respectively.

5.5.1 An algorithmic procedure for the construction via starter

1. Label the treatments 1, 2, ..., $2m$.

2. Partition the treatment set into m pairs, that is, m 2-subsets, viz $\{x_1, y_1\}$, $\{x_2, y_2\}$, . . ., $\{x_m, y_m\}$ such that the differences (reduced modulo $2m$) between these pairs of treatments are ± 1, ± 2, . . ., $\pm m$, respectively, thereby forming a starter in the cyclic group $\mathbb{Z}_{2m}$.

3. Create an $m \times 4m$ array and label its rows $i = 1, 2, ..., m$ and its columns $j = 1, 2, ..., 2m, 2m + 1, ..., 4m$.

4. For all $i = 1, 2, ..., m$, insert in the cell in position $(i, 1)$ of the array (that is, the cell in row i and column 1) the 2-subset, $\{x_i, y_i\}$ obtained in 2.

5. For all $i = 1, 2, ..., m$, develop the block in position $(i, 1)$, which contains $\{x_i, y_i\}$, cyclically, via successive addition of 1 (mod $2m$), thereby generating the block in position (i, j), for all $j = 2, 3, ..., 2m$.

6. Denote by P_1, the parallel class formed by m blocks containing distinct pairs of treatments in row m (between columns 1 and $2m$ of the array) where the treatments in each of these blocks concur twice in this row, that is, each block in row m between columns 1 and $2m$ has multiplicity 2; then, find a permutation, α that relabels the treatments in each set in P_1 to obtain another parallel class, P_2, where P_2 contains no pair in common with P_1 and $P_1 \cup P_2$ gives the edges of a connected design, that is, a single polygon on $2m$ vertices.

7. Apply α to every treatment in the first design Λ_1, which occupies columns 1 to $2m$ of the array to obtain Λ_2 occupying columns $2m + 1$ to $4m$.

5.5.2 An algorithmic procedure for the construction via BTD

1. Label the treatments $1, \ldots, u, \infty$, where $u = 2m - 1$.

2. Create an $m \times 4m$ array and label its rows $i = 1, \ldots, m-1, \infty$ and the columns $j = 1, \ldots, u, \infty, u+2, \ldots, 4m$.

3. For $i = 1, \ldots, m-1$ and $j = 1, \ldots, u$, put $S_{ij} = \{j+i, j-i\}$; and put $S_{\infty j} = \{j, \infty\}$.

4. For $j = 1, \ldots, u-1$ and $i = i^* \in \{2j, -2j\} \cap \{1, \ldots, m-1\}$, exchange S_{i^*j} with $S_{\infty j}$, where i^* is the unique entry in the intersection region of the sets.

5. For $i = 1, \ldots, m-1$, put $S_{i\infty} = \{3i/2, -3i/2\}$; and put $S_{\infty\infty} = \{u, \infty\}$.

6. Denote by P_1, the parallel class formed by the m blocks in either of columns u and ∞ where the treatments in each of these blocks concur once in each of these columns, that is, each block in either column u or column ∞ has multiplicity 2 (while the other treatments in the rest of the blocks concur only once); then, find a permutation, γ, say, that relabels the treatments in each set in P_1 to obtain another parallel class, P_2, where P_2 contains no pair in common with P_1 and $P_1 \cup P_2$ gives the edges of a connected design, that is, a single polygon on $2m$ vertices.

7. Apply γ to every treatment in the first design Λ_1, which occupies columns 1 to ∞ of the array to obtain Λ_2 occupying columns $u + 2$ to $4m$.

5.5.3 Basis for imposing the restriction that $P_1 \cup P_2$ should give the edges of a polygon on $2m$ vertices

If we have 6 treatments and 6 edges, one possibility is a hexagon, another possibility is 2 triangles. If we have 6 blocks of size 2, the hexagon is best because the other one is not connected. If we consider a BIBD-extended design, suppose we have all pairs from 6 treatments in that BIBD. Then for our BIBD-extended design, having a BIBD and adding an optimal design to it should be better than adding a non-optimal design. A good strategy is that what is added should be good in the smallest case.

For instance, consider the two designs for 6 treatments in 6 blocks of size 2–the hexagon and the 2 triangles. For the design that constitutes the hexagon, its scaled information matrix has the eigenvalues 0.0000, 0.2500 (2), 0.7500 (2) and 1.0000, where the values in the brackets are the corresponding multiplicities; giving the c.e.f.s of the design to be 0.2500 (2), 0.7500 (2) and 1.0000 such that the A-, D- and E-efficiency measures are 0.4286, 0.5119 and 0.2500, respectively. Similarly, for the design with the 2 triangles, the eigevnalues of its scaled information matrix are 0.0000 (2) and 0.7500 (4) with the corresponding multiplicities in brackets, hence the c.e.f.s are 0.0000, 0.7500 (4) and the A-, D- and E-efficiency measures are all zero, showing that the design with the hexagon is better than the one with 2 triangles with respect to the A-, D- and E-optimality criteria.

Now, consider the BIBD-extended designs consisting of all the $\binom{6}{2} = 15$ blocks from the BIBD and the 6 blocks from each of the hexagon and the 2 triangles added to it. The scaled information matrix of the BIBD-extended design has eigenvalues 0.0000, 0.5000 (2), 0.6429 (2) and 0.7143, which gives the c.e.f.s as 0.5000 (2), 0.6429 (2) and 0.7143, for the case that the hexagon is added to the BIBD. Hence the A-, D- and E-efficiency measures are 0.5875, 0.5938 and 0.5000, respectively. Similarly, for the BIBD-extended design where the 2 triangles are added to the BIBD, its scaled information matrix has the eigenvalues 0.0000, 0.4286 and 0.6429 (4), giving the c.e.f.s to be 0.4286 and 0.6429 (4), and the A-, D- and E-efficiency measures to be 0.5845, 0.5928 and 0.4286, respectively.

Hence, the BIBD-extended design is better on all the 3 optimality criteria (A-, D- and E-) when the hexagon is added to the BIBD than when the 2 triangles are added.

Comments. Let the constructed design be Γ. Then, by the algorithm, Γ takes the form

$$\Gamma = \begin{array}{|c||c|}\hline \Lambda_1 & \Lambda_2 \\ \hline \end{array}$$

Another design of the same size can be obtained by swapping the order of inserting Λ_1 and Λ_2 in the array, where the design resulting from the swapping is isomorphic to the original design, Γ.

Furthermore, if the array size is adjusted to $4m \times m$, and the roles of rows and columns are exchanged, then the resulting design is a $(4m \times m)/2$ RGSLR.

Again, by adjusting the size of the array to $2m \times 2m$, and then putting Λ_1 and Λ_2, we obtain another design, Γ^*, say, which is a $(2m \times 2m)/2$ RGSLR, and can take the form

$$\Gamma^* = \begin{array}{|c|}\hline \Lambda_1 \\ \hline\hline \Lambda_2 \\ \hline \end{array}$$

if Λ_1 is inserted in the array to cover rows 1 to m and then Λ_2 which covers rows $m+1$ to $2m$. However, if Λ_1 and Λ_2 are put in reversed order in the array, it produces another design of the same size, the two designs being isomorphic.

5 6	6 7	7 8	8 9	9 10	10 1	1 2	2 3	3 4	4 5	6 1	1 3	3 7	7 8	8 9	9 10	10 2	2 4	4 5	5 6
1 9	2 10	3 1	4 2	5 3	6 4	7 5	8 6	9 7	10 8	10 8	2 9	4 10	5 2	6 4	1 5	3 6	7 1	8 3	9 7
3 10	4 1	5 2	6 3	7 4	8 5	9 6	10 7	1 8	2 9	4 9	5 10	6 2	1 4	3 5	7 6	8 1	9 3	10 7	2 8
4 8	5 9	6 10	7 1	8 2	9 3	10 4	1 5	2 6	3 7	5 7	6 8	1 9	3 10	7 2	8 4	9 5	10 6	2 1	4 3
2 7	3 8	4 9	5 10	6 1	7 2	8 3	9 4	10 5	1 6	2 3	4 7	5 8	6 9	1 10	3 2	7 4	8 5	9 6	10 1

Figure 5.12: Γ: A $(5 \times 20)/2$ RGSLR for 10 treatments

The quotient block design (QBD) of Λ_g, where $g = 1, 2$ is a BIBD-extended RGD. It comprises an RGD with m blocks (whose blocks are P_g), and a BIBD which has $b = m(2m-1)$ blocks (having index, $\lambda = 1$), as extension. Hence, the overall design, Γ or Γ^* has its QBD comprising an RGD with $2m$ blocks (whose blocks are $P_1 \cup P_2$) whose treatment concurrence counts are 0 and 1, and a BIBD with $b = 2m(2m-1)$, $\lambda = 2$, as extension. In particular, for Λ_g, $g = 1, 2$, the treatment concurrences are 2 for each pair in P_g and 1, otherwise. Since $P_g \cap P_{g'} = \emptyset$, then each pair in P_g, $g = 1, 2$ appears exactly once in $\Lambda_{g'}$, $g \neq g'$. Hence, in Γ or Γ^*, the treatment concurrences are 3 for each pair in $P_1 \cup P_2$, and 2, otherwise. Thus, Γ or Γ^* is a RGSLR whose QBD is a BIBD-extended RGD.

Moreover, since the constructed design is a RGSLR whose QBD is a BIBD-extended RGD, then by virtue of the BIBD component, it is a connected design.

We illustrate the construction with the following examples

Example 5.5.1. Let $v = 10$. Then $m = 5$. The sets $\{5, 6\}$, $\{1, 9\}$, $\{3, 10\}$, $\{4, 8\}$ and $\{2, 7\}$, for instance, form a starter in $\mathbb{Z}_{10}$, and we obtain a $(5 \times 20)/2$ RGSLR, Γ as shown in Figure 5.12 using the algorithmic procedure given in section 5.5.1.

Notice that Λ_1 and Λ_2 are the designs shown in Figures 5.13 and 5.14, respectively, where $P_1 = \{\{2, 7\}, \{3, 8\}, \{4, 9\}, \{5, 10\}, \{6, 1\}\}$. Similarly, $P_2 = \{\{2, 3\}, \{4, 7\}, \{5, 8\}, \{6, 9\}, \{1, 10\}\}$.

We note that P_2 was obtained by imposing a permutation, α on the treatments in each set of P_1, where

$$\alpha = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\ 10 & 2 & 4 & 5 & 6 & 1 & 3 & 7 & 8 & 9 \end{pmatrix}$$

Notice that P_2 contains no pair in common with P_1, that is $P_1 \cap P_2 = \emptyset$. Also $P_1 \cup P_2$ form the edges of a 10-gon (or decagon). The QBD of Λ_1 comprises P_1, which is an RGD with 5 blocks and a BIBD with $b = 45$, $\lambda = 1$, as extension.

Similarly, the QBD of Λ_2 comprises P_2, which is also an RGD with 5 blocks, as before and a BIBD (with the same value of b and λ as Λ_1), as extension.

5 6	6 7	7 8	8 9	9 10	10 1	1 2	2 3	3 4	4 5
1 9	2 10	3 1	4 2	5 3	6 4	7 5	8 6	9 7	10 8
3 10	4 1	5 2	6 3	7 4	8 5	9 6	10 7	1 8	2 9
4 8	5 9	6 10	7 1	8 2	9 3	10 4	1 5	2 6	3 7
2 7	3 8	4 9	5 10	6 1	7 2	8 3	9 4	10 5	1 6

Figure 5.13: Λ_1: A $(5 \times 10)/2$ RGSLR for 10 treatments obtained via starter

6 1	1 3	3 7	7 8	8 9	9 10	10 2	2 4	4 5	5 6
10 8	2 9	4 10	5 2	6 4	1 5	3 6	7 1	8 3	9 7
4 9	5 10	6 2	1 4	3 5	7 6	8 1	9 3	10 7	2 8
5 7	6 8	1 9	3 10	7 2	8 4	9 5	10 6	2 1	4 3
2 3	4 7	5 8	6 9	1 10	3 2	7 4	8 5	9 6	10 1

Figure 5.14: Λ_2: A $(5 \times 10)/2$ RGSLR for 10 treatments obtained from Λ_1 via the permutation, α

The design, Γ, shown in Figure 5.12 is a RGSLR whose QBD is a BIBD-extended RGD, comprising an RGD with 10 blocks, whose blocks are $P_1 \cup P_2$ and a BIBD with $b = 90$, $\lambda = 2$, as extension. The blocks of the RGD form the edges of a 10-gon (or decagon). The design is connected since it contains a BIBD.

Example 5.5.2. Let $v = 18$. Then $m = 9$, and the sets $\{16, 17\}$, $\{3, 5\}$, $\{10, 13\}$, $\{4, 8\}$, $\{2, 15\}$, $\{6, 12\}$, $\{7, 14\}$, $\{1, 11\}$ and $\{9, 18\}$, for instance, form a starter in $\mathbb{Z}_{18}$, and we obtain a $(9 \times 36)/2$ RGSLR for 18 treatments as shown in Figure 5.15 via the algorithmic procedure in section 5.5.1, where Λ_1 and Λ_2 are the designs shown in Figures 5.16 and 5.17, respectively.

Notice that $P_1 = \{\{9, 18\}, \{10, 1\}, \{11, 2\}, \{12, 3\}, \{13, 4\}, \{14, 5\}, \{15, 6\}, \{16, 7\}, \{17, 8\}\}$. Similarly, $P_2 = \{\{9, 10\}, \{11, 18\}, \{12, 1\}, \{13, 2\}, \{14, 3\}, \{15, 4\}, \{16, 5\}, \{17, 6\}, \{8, 7\}\}$. P_2 was obtained by imposing a permutation β on the treatments in each set of P_1, where

$$\beta = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 & 17 & 18 \\ 18 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 9 & 11 & 12 & 13 & 14 & 15 & 16 & 17 & 8 & 10 \end{pmatrix}$$

Notice also that, P_2 contains no pair in common with P_1, and $P_1 \cup P_2$ form the edges of an 18-gon (or octadecagon). The QBD of Λ_1 comprises P_1, which is an RGD with 9

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

Figure 5.15: Γ: A $(9 \times 36)/2$ RGSLR for 18 treatments

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

Figure 5.16: Λ_1: A $(9 \times 18)/2$ RGSLR for 18 treatments obtained via starter

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

Figure 5.17: Λ_2: A $(9 \times 18)/2$ RGSLR for 18 treatments obtained from Λ_1 by imposing the permutation, β

2 5	2 ∞	3 ∞	5 3	1 4	4 1	5 ∞	5 3	1 3	∞ 1	4 2	2 4
1 ∞	4 5	5 1	4 ∞	2 3	3 2	4 3	2 ∞	∞ 4	2 3	5 1	1 5
3 4	3 1	4 2	1 2	5 ∞	5 ∞	1 2	1 4	2 5	4 5	∞ 3	∞ 3

Figure 5.18: A $(3 \times 12)/2$ RGSLR for 6 treatments

blocks, and a BIBD with $b = 153$, $\lambda = 1$, as extension.

Similarly, the QBD of Λ_2 comprises P_2, which is also an RGD with 9 blocks, like P_1, and a BIBD (with the same value of b and λ as in Λ_1), which is the extension.

The design, Γ, shown in figure 5.15 is a RGSLR with QBD a BIBD-extended RGD, comprising an RGD with 18 blocks, whose blocks are $P_1 \cup P_2$ and a BIBD with $b = 306$, $\lambda = 2$, as extension, where the blocks of the RGD form the edges of an18-gon (or octadecagon). It is a connected design by virtue of the BIBD component,

Example 5.5.3. Let $v = 6$. Then $m = 3$. We obtain a $(3 \times 12)/2$ RGSLR shown in Figure 5.18 via the procedure in section 5.5.2.

Notice from Figure 5.18 that $P_1 = \{\{1, 4\}, \{2, 3\}, \{5, \infty\}\}$, and $P_2 = \{\{4, 2\}, \{5, 1\}, \{\infty, 3\}\}$.

Furthermore, the permutation

$$\gamma = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & \infty \\ \\ 4 & 5 & 1 & 2 & \infty & 3 \end{pmatrix}$$

was imposed on every treatment in P_1 to obtain P_2 and also on every treatment in Λ_1 to obtain Λ_2.

5.6 Construction for designs of the class $(m \times 6m)/2$, where $v = 2m$

An $(m \times 6m)/2$ RGSLR for $2m$ treatments is another special case of $(m \times m(\theta + 2))/2$ RGSLRs for $2m$ treatments, where $\theta = 4$. The construction given here extends the construction for $(m \times 4m)/2$ RGSLRs given in section 5.5.1. In this case, the design under construction requires an extra $2m$ columns more than an $(m \times 4m)/2$ SLR (but on the same set of treatments with cardinality $2m$), and similarly, $4m$ columns more than an $(m \times 2m)/2$ SLR.

We start by obtaining a parent $(m \times 2m)/2$ RGSLR for $2m$ treatments, where this parent design occupies columns 1 to $2m$ of an $m \times 6m$ array. We modify the two procedures in Section 5.5, viz, two permutations are sought. A parallel class is identified in row m (or either of columns u and ∞, where $u = 2m - 1$) of the parent design, as described in section 5.5 depending on the method used. One of the permutations is first applied to the treatments within each set in the parallel class to obtain another parallel class. The other permutation is then imposed, also on the treatments within each set in the same parallel class from the parent design to obtain a third parallel class.

Let α_1 and α_2 denote the permutations; and let P_1, P_2 and P_3 denote the three parallel classes, respectively. Suppose α_1 is applied to P_1. Then α_1 relabels the treatments within each cell in the parent design to obtain treatments to be contained in a corresponding cell between columns $2m + 1$ and $4m$ of an $m \times 6m$ array. Similarly, α_2 permutes the treatments within each cell in the parent design to generate entries for a corresponding cell between columns $4m + 1$ and $6m$ of the same array.

The choices of α_1 and α_2 are such that, P_1, P_2 and P_3 are pairwise disjoint, that is, P_t and $P_{t'}$ contain no pair of treatment symbols in common, for all $t, t' = 1, 2, 3$, where $t \neq t'$; and the 'union' of each pair of these parallel classes gives the edges of a connected design, that is, a single polygon on $2m$ vertices. Moreover, we choose P_3 such that each pair of symbols in it are equidistant from each other on the $2m$-gon formed by $P_1 \cup P_2$, where the distance between the symbols of each pair on the $2m$-gon is maximal, m, giving the diameter of the $2m$-gon.

5.6.1 An algorithmic procedure for constructing the designs via starter

1. Label the treatments 1, 2, ..., $2m$.

2. Partition the treatment set into m pairs, that is, m 2-subsets, viz $\{x_1, y_1\}$, $\{x_2, y_2\}$, . . ., $\{x_m, y_m\}$ such that the differences (reduced modulo $2m$) between these pairs of treatments are ± 1, ± 2, . . ., $\pm m$, respectively, thereby forming a starter in the cyclic group $\mathbb{Z}_{2m}$.

3. Create an $m \times 6m$ array and label its rows $i = 1, 2, ..., m$ and its columns $j = 1, 2, ..., 6m$.

4. For all $i = 1, 2, ..., m$, insert in the cell in position $(i, 1)$ of the array (that is, the cell in row i and column 1) the 2-subset, $\{x_i, y_i\}$ obtained in step 2.

5. For all $i = 1, 2, ..., m$, develop the block in position $(i, 1)$, which contains $\{x_i, y_i\}$, cyclically, via successive addition of 1 (mod $2m$), thereby generating the block in position (i, j), for all $j = 2, 3, ..., 2m$.

6. Denote by P_1, the parallel class formed by m distinct pairs of treatment symbols in row m between columns 1 and $2m$ of the array, where each pair concurs twice in this row; and find a permutation, α_1, say, of the treatments that relabels the treatments in each pair in P_1 to obtain another parallel class, P_2, where P_2 contains no pair in common with P_1. Find another permutation, α_2 of the treatments and impose it on P_1 to relabel the treatments within its pairs, thereby obtaining another parallel class, P_3, where P_3 contains no pair in common with $P_1 \cup P_2$; and for all $t \neq t'$, where $t = 1, 2, 3$. $P_t \cup P_{t'}$ gives the edges of a connected design-a single polygon on $2m$ vertices.

7. Apply α_1 to every treatment in the first design Υ_1, which occupies columns 1 to $2m$ of the array to obtain Υ_2 in columns $2m + 1$ to $4m$. Similarly, apply α_2 to Υ_1 to obtain Υ_3 to fill columns $4m + 1$ to $6m$.

5.6.2 An algorithmic procedure for the construction via BTD

1. Label the treatments $1, \ldots, u, \infty$, where $u = 2m - 1$.

2. Create an $m \times 6m$ array and label its rows $i = 1, \ldots, m - 1, \infty$ and the columns $j = 1, \ldots, u, \infty, u + 2, \ldots, 4m, ..., 6m$.

3. For $i = 1, \ldots, m-1$ and $j = 1, \ldots, u$, put $S_{ij} = \{j+i, j-i\}$; and put $S_{\infty j} = \{j, \infty\}$.

4. For $j = 1, \ldots, u-1$ and $i = i^* \in \{2j, -2j\} \cap \{1, \ldots, m-1\}$, exchange $S_{i^* j}$ with $S_{\infty j}$, where i^* is the unique entry in the intersection region of the sets.

5. For $i = 1, \ldots, m - 1$, put $S_{i\infty} = \{3i/2, -3i/2\}$; and put $S_{\infty\infty} = \{u, \infty\}$.

6. Denote by P_1, the parallel class formed by the m blocks in either of columns u and ∞ where the treatments in each of these blocks concur once in each of these columns , that is, each block in either column u or column ∞ has multiplicity 2 (while the other treatments in the rest of the blocks concur only once); then, find a permutation, γ_1, say, that relabels the treatments in each set in P_1 to obtain another parallel class, P_2, where P_2 contains no pair in common with P_1. Find another permutation, γ_2 and apply it to each treatment in P_1 to obtain another parallel class P_3, where P_3 contains no pair in common with $P_1 \cup P_2$; and for all $w = 1, 2, 3$, where $w \neq w'$, $P_w \cup P_{w'}$ gives the edges of a connected design–a single polygon on $2m$ vertices.

7. Apply γ_1 to every treatment in the first design Υ_1, which occupies columns 1 to ∞ of the array to obtain Υ_2 occupying columns $u + 2$ to $4m$. Similarly, apply γ_2 to Υ_1 to obtain Υ_3 to fill columns $4m + 1$ to $6m$.

Comments. Let Ω denote the constructed design. Then, by the algorithm, Ω takes the form

$$\Omega = \begin{array}{|c||c||c|}\hline \Upsilon_1 & \Upsilon_2 & \Upsilon_3 \\ \hline\end{array}$$

where Υ_l, $l = 1, 2, 3$ can appear in any order in the array. Thus another design of the same size can be obtained by interchanging the positions where Υ_l, $l = 1, 2, 3$ appears in an array of the same size. This means that, in addition to Ω, there are $n_p - 1$ other designs of the same size that can be obtained by randomly ordering the constituent designs within the array, where $n_p =^3 P_3 = 3!$. Each resulting design is isomorphic to Ω.

Furthermore, if the array size is adjusted to $6m \times m$, and the roles of rows and columns in the entire array are exchanged, then the resulting design is a $(6m \times m)/2$ RGSLR. Other designs of the same size can be obtained if the positions of the Υ_is are interchanged within an array of the same size.

Again, by adjusting the size of the array to $3m \times 2m$, and then putting Υ_1, Υ_2 and Υ_3, we obtain another design, Ω^*, say, which is a $(3m \times 2m)/2$ RGSLR for $2m$ treatments, and takes the form

$$\Omega^* = \begin{array}{|c|}\hline \Upsilon_1 \\ \hline\hline \Upsilon_2 \\ \hline\hline \Upsilon_3 \\ \hline\end{array}$$

if they are put in the array in a natural order such that Υ_1 appears between rows 1 and m, Υ_2 appears between rows $m + 1$ and $2m$ and then Υ_3 appears between rows $2m + 1$ to

$3m$. Any change in ordering within an array of the same size produces another design of the same size which is isomorphic to Ω^*.

Moreover, if the array size is adjusted to $2m \times 3m$, and the roles of rows and columns are exchanged within Υ_l, for all $l = 1, 2, 3$, then the resulting design is a $(2m \times 3m)/2$ RGSLR. Also different orderings/arrangements within arrays of the same size produce designs of the same size that are isomorphic.

The QBD of Υ_l, $l = 1, 2, 3$ is a BIBD-extended RGD. It comprises an RGD with m blocks (whose blocks are P_l), and a BIBD which has $b = m(2m-1)$ blocks (having index, $\lambda = 1$), as extension. Hence, the overall design (whether Ω, Ω^*, or any of the other possibilities) has its QBD comprising an RGD with $3m$ blocks (whose blocks are $P_1 \cup P_2 \cup P_3$) which has concurrence counts 0 and 1, and a BIBD with $b = 3m(2m-1)$, $\lambda = 3$, as extension. Thus, the overall design is a RGSLR whose QBD is a BIBD-extended RGD whose treatment concurrence counts are 3 and 4.

In particular, for Υ_l, where $l = 1, 2, 3$, the treatment concurrence counts are 2 for each pair in P_l and 1, otherwise. Since for all $l \neq l'$, $P_l \cap P_{l'} = \emptyset$ (the P_ls are pairwise disjoint), then each pair in P_l, where l= $1, 2, 3$ appears exactly once in $\Upsilon_{l'}$ for all $l \neq l'$. Hence in Ω, Ω^*, or any of the other possibilities, the treatment concurrence counts are 4 for each pair in $P_1 \cup P_2 \cup P_3$, and 3, otherwise. That is, if $\lambda_{uu'}$ denote the treatment concurrence counts between the pair (u, u') in the overall design, then

$$\lambda_{uu'} = \begin{cases} 4, & \text{if } (u, u') \in \bigcup_{l=1}^{3} P_l, \\ 3, & \text{otherwise.} \end{cases}$$

Moreover, by virtue of the BIBD component, it is a connected design.

We illustrate the construction with the following example

Example 5.6.1. Let $v = 10$. Then $m = 5$ and the sets $\{5, 6\}$, $\{1, 9\}$, $\{3, 10\}$, $\{4, 8\}$ and $\{2, 7\}$, as was given in example 5.5.1, form a starter in $\mathbb{Z}_{10}$. By the procedure in section 5.6.1, we obtain Ω, a $(5 \times 30)/2$ RGSLR for 10 treatments whose QBD is a BIBD-extended RGD: see Figure 5.19.

Notice that $P_1 = \{\{2, 7\}, \{3, 8\}, \{4, 9\}, \{5, 10\}, \{6, 1\}\}$. Similarly, $P_2 = \{\{2, 3\}, \{4, 7\}, \{5, 8\}, \{6, 9\}, \{1, 10\}\}$ and $P_3 = \{\{2, 1\}, \{7, 10\}, \{4, 5\}, \{9, 8\}, \{6, 3\}\}$.

P_2 and P_3 can be obtained from P_1 by imposing the permutations, α_1 and α_2, respectively, on every treatment in P_1, where

$$\alpha_1 = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\ 10 & 2 & 4 & 5 & 6 & 1 & 3 & 7 & 8 & 9 \end{pmatrix}$$

5	6	7	8	9	10	1	2	3	4	6	1	3	7	8	9	10	2	4	5	9	6	1	10	5	8	3	2	7	4
6	7	8	9	10	1	2	3	4	5	1	3	7	8	9	10	2	4	5	6	6	1	10	5	8	3	2	7	4	9
1	2	3	4	5	6	7	8	9	10	10	2	4	5	6	1	3	7	8	9	3	2	7	4	9	6	1	10	5	8
9	10	1	2	3	4	5	6	7	8	8	9	10	2	4	5	6	1	3	7	5	8	3	2	7	4	9	6	1	10
3	4	5	6	7	8	9	10	1	2	4	5	6	1	3	7	8	9	10	2	7	4	9	6	1	10	5	8	3	2
10	1	2	3	4	5	6	7	8	9	9	10	2	4	5	6	1	3	7	8	8	3	2	7	4	9	6	1	10	5
4	5	6	7	8	9	10	1	2	3	5	6	1	3	7	8	9	10	2	4	4	9	6	1	10	5	8	3	2	7
8	9	10	1	2	3	4	5	6	7	7	8	9	10	2	4	5	6	1	3	10	5	8	3	2	7	4	9	6	1
2	3	4	5	6	7	8	9	10	1	2	4	5	6	1	3	7	8	9	10	2	7	4	9	6	1	10	5	8	3
7	8	9	10	1	2	3	4	5	6	3	7	8	9	10	2	4	5	6	1	1	10	5	8	3	2	7	4	9	6

Figure 5.19: Ω: A $(5 \times 30)/2$ RGSLR for 10 treatments

and

$$\alpha_2 = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\ 3 & 2 & 7 & 4 & 9 & 6 & 1 & 10 & 5 & 8 \end{pmatrix}$$

Notice that $\forall\, l \neq l'$, $P_l \cup P_{l'}$ form the edges of a $2m$-gon. Notice also that $(P_1 \cup P_2) \cap P_3 = (P_1 \cap P_3) \cup (P_2 \cap P_3) = \emptyset$. Hence $P_1 \cap P_3 = P_2 \cap P_3 = \emptyset$ and $P_1 \cap P_2 \cap P_3 = \emptyset$. Thus, $P_1 \cap P_2 = P_1 \cap P_3 = P_2 \cap P_3 = \emptyset$, that is, P_1, P_2 and P_3 are pairwise disjoint since $P_l \neq \emptyset$ for any $l = 1, 2, 3$. Moreover, $P_1 \cup P_2$, $P_1 \cup P_3$, and $P_2 \cup P_3$, each form the edges of a 10-gon

Notice also that, each constituent design, Υ_1, Υ_2 and Υ_3 in Figure 5.19, comprises an RGD, P_l, which has 5 blocks (whose treatment concurrences are 0 and 1) and a BIBD with 45 blocks (whose treatment concurrences are 1), which is the extension.

The constructed design, Ω, is thus, a RGSLR whose QBD is BIBD-extended, comprising an RGD with 15 blocks, whose blocks are $P_1 \cup P_2 \cup P_3$ (having concurrences 0 and 1) and a BIBD with 135 blocks, $\lambda = 3$, as extension. Overall, in the design, each pair of treatments in $P_1 \cup P_2 \cup P_3$ concurs 4 times, while each of those pairs that are not in $P_1 \cup P_2 \cup P_3$ concurs 3 times.

Notice that Ω extends the $(5 \times 20)/2$ RGSLR for 10 treatments (Γ) shown in Figure 5.12 by adding an extra 10 columns to it.

Example 5.6.2. Let $v = 6$. Then $m = 3$. An implementation of the algorithm in section 5.6.2 produces the $(3 \times 18)/2$ RGSLR shown in Figure 5.20.

Notice from Figure 5.20 that $P_1 = \{\{1, 4\}, \{2, 3\}, \{5, \infty\}$, $P_2 = \{\{4, 2\}, \{5, 1\}, \{\infty, 3\}\}$ and $P_3 = \{\{1, 3\}, \{2, 5\}, \{4, \infty\}\}$. Furthermore, the permutation γ_1, where

$$\gamma_1 = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & \infty \\ 4 & 5 & 1 & 2 & \infty & 3 \end{pmatrix}$$

2 5	2 ∞	3 ∞	5 3	1 4	4 1	5 ∞	5 3	1 3	∞ 1	4 2	2 4	2 4	2 ∞	5 ∞	4 5	1 3	3 1
1 ∞	4 5	5 1	4 ∞	2 3	3 2	4 3	2 ∞	∞ 4	2 3	5 1	1 5	1 ∞	3 4	4 1	3 ∞	2 5	5 2
3 4	3 1	4 2	1 2	5 ∞	5 ∞	1 2	1 4	2 5	4 5	∞ 3	∞ 3	5 3	5 1	3 2	1 2	4 ∞	4 ∞

Figure 5.20: A $(3 \times 18)/2$ RGSLR for 6 treatments

imposed on every treatment in P_1 gives P_2. Similarly, P_3 can be obtained by applying the permutation γ_2 to every treatment in P_1, where

$$\gamma_2 = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & \infty \\ 1 & 2 & 5 & 3 & 4 & \infty \end{pmatrix}$$

5.7 Construction when v is odd

We give constructions for $(m \times \eta m)/2$ RGSLRs where the number of treatments, $v = m$ is odd. As in the previous constructions, we consider designs whose QBDs are BIBD-extended. We give constructions for designs with $\eta = 1, 2, \ldots, t$, where $t \leq \delta$, $\delta = (m-1)/2$. In particular, if m is an odd prime, then $t = \delta$. But if m is not an odd prime, then $t < \delta$. When $\eta = 1$, we obtain the parent/basic design, which is then utilized to give constructions for larger designs with higher values of η. We also give, in addition, a general construction for these. Moreover, if $\eta = \delta$, the construction gives a BSLR.

In the designs under construction, each treatment appears $n_r = 2\eta$ times in each row and $n_c = 2$ times in each column, hence overall, appears $2\eta m$ times. Moreover, for any treatment of the design, the sum of concurrences with other treatments is $2\eta m$.

Furthermore, in a similar manner to when v was even, we obtain RGSLRs whose QBDs are also BIBD-extended by adding a BSLR to a RGSLR, where both contain the same treatments and are conformable with respect to size.

5.7.1 Construction for $(m \times m)/2$ RGSLRs

The designs considered in this section are a special case of the $(m \times \eta m)/2$ RGSLRs given in the preceding section where $\eta = 1$. They are the basic design from which the larger designs can be obtained. It follows from the discussion in the preceding section that each treatment of these designs appears 2 times in each row and in each column, hence appears $2m$ times in the design. Similarly, for each treatment, the sum of concurrences with the other treatments is $2m$, which is identical to its replication number but doubles the number of treatments.

5.7.2 Some preliminaries

Let the treatment set be denoted $V = \{1, \ldots, m\}$. The m treatments can be put in m sets of size 2 such that each treatment appears in 2 sets and each time with a different treatment, that is, it appears 2 times overall and the multiplicity of each set is 1, hence no repeats of the sets. To find an appropriate allocation (pattern) of concurrences for each treatment, $\alpha \in V$ with its pair within its set and also with distinct pairs from other sets (where any pair from another set when paired with α is not a pair contained in any of the sets) such that these concurrences sum to $2m$, we proceed as follows. Firstly, we find the number of distinct pairs that α requires from external sets, that is, other sets excluding the set where it is contained.

Denote the member sets for the m sets by A_i, where $i = 1, \ldots, m$. Since A_i, for all i contains distinct treatments and each treatment appears 2 times among the sets with no repeats of A_i, then for $\alpha, \beta \in A_i$, there exist $A_{i'}$ and $A_{i''}$, where $i \neq i' \neq i''$ such that $\alpha \in A_{i'}$ and $\beta \in A_{i''}$. It follows that $A_i \cap A_{i'} = \{\alpha\}$, $A_i \cap A_{i''} = \{\beta\}$ and $A_i \cap (A_{i'} \cup A_{i''}) = (A_i \cap A_{i'}) \cup (A_i \cap A_{i''}) = A_i$.

For a fixed α, between $A_{i'}$ and $A_{i''}$, α has exactly 1 distinct pair such that it does not form a set which is already among the A_is; this pair is from $A_{i''}$ and is precisely the element in the singleton $A_{i''} \backslash \{\beta\}$. Denote this treatment by ϵ.

Now, by excluding A_i, $A_{i'}$ and $A_{i''}$, there are $m - 3$ sets left which contain, overall, $2(m-3)$ treatments from which α can select distinct pairs from, since these $2(m-3)$ treatments are not all distinct. Notice that for each of the $m-3$ sets, there are also 2 other sets which contain its elements (treatments)-1 treatment contained in one set and the remaining treatment in the other set, since each treatment appears 2 times overall among the sets and with a different treatment each time. Moreover, ϵ is among the remaining $2(m-3)$ treatments, that is, it also appears in one of the $(m-3)$ sets. Hence needs to be excluded, that is, not to be counted twice. Similarly, the single element, ρ, say, in the singleton $A_{i'} \backslash \{\alpha\}$ also appears a second time, among the $2(m-3)$ elements, which also needs to be excluded from there when counting since $\{\rho\} \cup \{\alpha\} = A_{i'}$ ($\{\alpha\} \subset A'_i$), which is among the A_is.

Finally, after excluding the 2 treatments, ϵ and ρ from the $2(m-3)$ elements, each of the remaining $2(m-3) - 2$ (or $2(m-4)$) treatments appears 2 times and needs to be counted exactly once to make a distinct pair with α. Thus there are $2(m-4)/2 = m-4$ distinct pairs with α from among the $2(m-3)$ elements.

Overall, considering all the pairs, there is a total of $1 + (m-4) = m-3$ distinct treatments from among the sets where α does not appear (where 1 in the summand accounts for the treatment ϵ) that can concur with α, that is, when any of them is paired with α it does not appear as a set among the A_is.

Notice that, there are 2 distinct treatments, β and ρ which are in the sets A_i and $A_{i'}$, respectively, where α makes an appearance, hence can also concur with α differently. The value 2 denoting the number of distinct treatments can also be seen by noting that for all $\alpha \in V$, there are $m-1$ distinct treatments which can concur with α. Moreover, $m-3 < m-1$. By letting x denote the number of treatments that complement the concurrences of the $m-3$ treatments from external sets where α does not make an appearance. Then $m-3+x = m-1$, which implies that $x = 2$. Now, let λ_1 denote the treatment concurrence counts between α and each of β and ρ. Similarly, let λ_2 denote the treatment concurrence counts between α and each of the $m-3$ treatments from external sets, that is, those sets where α does not make an appearance. Then, for the concurrence relationship, it follows that

$$2\lambda_1 + (m-3)\lambda_2 = 2m \tag{5.2}$$

where $m \geq 3$.

The only solution set that satisfies (5.2) is $(\lambda_1, \lambda_2) = (3, 2)$, which gives a RGD. It is revealed in (5.2) that, to have a RGD, the only appropriate pattern of concurrences for all $\alpha \in V$ is to have α concur 3 times with each of the 2 treatments, β and ρ that appear in the same set with it, that is every pair in the same set needs to appear 3 times as a block. Similarly, α needs to concur 2 times with each of the remaining $m-3$ distinct treatments which do not appear in the same set with it and also when paired with α does not make one of the m sets. In particular, α concurs 2 times with each element of the set $V^* = V \backslash \{\beta, \rho : \lambda_1 = 3\} \cup \{\alpha\}$, for all $\alpha \in V$, where $|V^*| = m-3$.

The aforementioned m sets, $A_1, A_2, \ldots, A_m$ take the form $\{1, 1+c\}$, $\{2, 2+c\}$, $\ldots$, $\{m-1, m-1+c\}$, $\{m, m+c\}$, if a cyclic construction is used, where $c \in \{1, 2, \ldots, \delta\}$, $\delta = (m-1)/2$. In particular, if $c = \delta$, then the m sets are identical to $\{1, (m+1)/2\}$, $\{2, (m+3)/2\}$, $\ldots$, $\{m-1, 3(m-1)/2\}$, $\{m, (3m-1)/2\}$, respectively with reduction modulo m.

Comment. The concurrence relationship between the treatments so that we get a RGD can be viewed in a much more simple way by expressing the sum of concurrences, $2m$ as a sum of two components, viz,

$$2m = (m-1)2 + 2 \tag{5.3}$$

Notice that the right hand side of (5.3) is identical to $(m-3)2 + 2(3)$ or $2(3) + (m-3)2$, showing that the only way to have a RGD is for each treatment to have two concurrences of 3 and all the rest ($m-3$ of them) equal to 2. Hence the QBD will be a BIBD if $m = 3$, since each treatment will have precisely 2 (all) concurrences 3.

Furthermore, notice that if $m = 3$, then $\delta = 1$, where $\delta = (m-1)/2$. Since m is an odd prime, as will be seen later in section 5.8.4, a $(3 \times 3)/2$ RGSLR for 3 treatments which is a special case of an $(m \times \eta m)/2$ RGSLR for m treatments, where $m = 3$ and $\eta = 1$ (η having the value δ) is a BSLR.

5.7.3 Construction procedure

Notice that the RGSLR under construction requires m^2 blocks. We use the cyclic group $\mathbb{Z}_m$ and regard $\mathbb{Z}_m$ as $\{1, 2, \ldots, m\}$. Notice that each set among the A_is generates all the other sets, that is, m blocks by its cyclic development via successive addition of 1 reduced modulo m, hence for each of these m pairs to appear 3 times in the design, as required, we need 3 sets from the A_is where each of them can generate m blocks. Thus with these 3 sets, $3m$ blocks, that is, 3 rows (or columns) can be generated in the design. Furthermore, for each of the $m-3$ other pairs which is identical to the number of rows (or columns) left; notice that $m-3 = 2(\delta-1)$. Hence with each of the $\delta-1$ other possible values of c another set of m blocks (1 row or column) can be generated, and each block also generates the rest via the same procedure and overall $m(\delta-1)$ blocks which is equivalent to $\delta-1$ rows (or columns). By making 2 copies of the $m(\delta-1)$ blocks, which is equivalent to 2 copies of the $\delta-1$ distinct sets of m blocks we obtain a total of $2m(\delta-1)$ blocks which can be used to form the remaining $2(\delta-1) = m-3$ columns. Notice that $2m(\delta-1) + 3m = m^2$, the total number of blocks, as required. Moreover, each block from the 3 sets of the A_is appears 3 times in the design while each block obtained from each of the other possible values of c appears 2 times.

Now to arrange the m^2 blocks into an $m \times m$ array to make an $(m \times m)/2$ RGSLR under construction, we modify the starter sets we used in the previous sections for some constructions involving designs with even value of v. This time, each treatment appears in 2 sets and the pairs of treatments in the m starter sets are such that, overall, the set of differences between the treatments in these sets (with a reduction modulo m) consists of $\pm1, \pm2, \ldots, \pm(\delta-1), \pm\delta$ with the multiplicity 3 for one of them (that is, one of these differences is identified with 3 sets in the starter) while the multiplicity is 2 for each of the other $\delta-1$ values (that is, the other $\delta-1$ differences are identified with 2 sets each in the starter). The set of differences is thus a multiset which consists of all the non-zero elements of $\mathbb{Z}_m$. We note that, any one of these differences can have the higher multiplicity. However, each time the starter sets are reformulated such that more sets are identified with another value of the differences, it produces another design of the same size.

However, for purposes of describing the construction, we choose one of them, $\pm\delta$, to have the higher multiplicity, 3, that is, 3 sets in the starter are identified with the differences $\pm\delta$ while the elements of the set $\{\pm1, \pm2, \ldots, \pm(\delta-1)\}$ has multiplicity 2 each, that is, each of these other differences is identified with 2 sets in the starter.

Definition 5.7.1. Let there be m non-empty sets (m being odd) consisting of 2-subsets of $\mathbb{Z}_m$ with each element appearing a constant number of times, overall, each time with a different element. We regard $\mathbb{Z}_m$ to be the set $\{1, \ldots, m\}$. Let the differences between the elements of these sets, modulo m, constitute a multiset consisting of all the non-zero elements of $\mathbb{Z}_m$. Denote the multiset of differences by $A = \{\pm\nu_i\}_{i=1}^{m}$. Let there exist $\pm\theta \in A$ whose multiplicity is λ^*, say, while the multiplicity of every other element of A is

$\lambda^* - 1$, where $\lambda^* \in \mathbb{Z}$, $\lambda^* > 1$. Then the m sets constitute a starter for the cyclic group $\mathbb{Z}_m$.

Let S_{ul} denote the lth set from the starter sets for which the differences, modulo m, between its elements are $\pm u$, where $u = 1, 2, \ldots, \delta$ and

$$l = \begin{cases} 1,2,3 & \text{if } u = \delta, \\ 1,2 & \text{if } u < \delta. \end{cases}$$

We proceed to give a table showing starters in $\mathbb{Z}_m$ for small odd values of m: see Table 5.1. An algorithmic procedure for constructing the design is also given in section 5.7.4.

Table 5.1: Starters in $\mathbb{Z}_m$ for some small odd values of m

m	starter								
5	$\{1,2\}$	$\{1,3\}$	$\{2,4\}$	$\{4,5\}$	$\{3,5\}$				
7	$\{1,2\}$	$\{3,5\}$	$\{2,5\}$	$\{3,6\}$	$\{4,6\}$	$\{1,7\}$	$\{4,7\}$		
9	$\{1,2\}$	$\{4,6\}$	$\{3,6\}$	$\{3,7\}$	$\{8,4\}$	$\{2,8\}$	$\{5,7\}$	$\{1,9\}$	$\{5,9\}$

5.7.4 An algorithmic procedure for the construction

1. Label the treatments 1, 2, ..., m.

2. Form m sets each of size 2 with the m treatments such that each treatment appears 2 times among the sets, each time with a different treatment and the differences (reduced modulo m) between the pairs of treatments in these sets are ± 1, ± 2, . . ., $\pm\delta$, the non-zero elements of $\mathbb{Z}_m$, whose multiplicities are 3 for $\pm\delta$ and 2 for others, and these form a starter for the cyclic group $\mathbb{Z}_m$.

3. Create an $m \times m$ array and label its rows $i = 1, 2, ..., m$ and its columns $j = 1, 2, ..., m$.

4. For all $u = 1, 2, \ldots, \delta$, and

$$l = \begin{cases} 1,2,3 & \text{if } u = \delta, \\ 1,2 & \text{if } u < \delta. \end{cases}$$

   put S_{ul} in the cell in position $(1, j)$ of the array, that is, the cell in row 1 and column j, where S_{ul} denotes the lth set in the starter for which the differences (reduced modulo m) between its elements is $\pm u$, and

$$j = \begin{cases} \delta, \delta+1, m & \text{if } u = \delta, \\ u, m-u & \text{if } u < \delta. \end{cases}$$

5. For all $j = 1, 2, ..., m$, develop the block in position $(1, j)$, cyclically, via successive addition of 1 $\pmod m$, thereby generating the m blocks in each column.

Comments. Each treatment appears 2 times in each row and in each column, hence appears $2m$ times overall which is identical to the sum of concurrences with each treatment. In particular, if $u = \delta$, then each pair of treatments with the difference $\pm\delta$ concurs 3 times in the design, once in each of columns δ, $\delta+1$ and m which are the columns generated by the starter set for which the difference between its constituent treatments is $\pm\delta$. Similarly, if $u < \delta$, then for any $u \in \{1, 2, \ldots, (\delta-1)\}$, each pair of treatments with the differences $\pm u$ concurs 2 times in the design, once in columns u and $m - u$. Notice that $\delta + 1 = m - \delta$.

The number of sets in the starter associated with each unique difference which corresponds to the multiplicity of that difference can be seen by writing $m = 2\delta + 1$ as $2(\delta - 1) + 3(1)$, where $\delta \geq 1$, showing that among the m sets consisting of pairs of treatments whose differences consist of elements of the set $\{\pm1, \pm2, \ldots, \pm(\delta - 1), \pm\delta\}$, 3 of the sets are identified with one difference, $\pm\delta$ in this case (that is, the difference, $\pm\delta$ has multiplicity 3) while 2 sets are identified with each of the other $\delta - 1$ differences (that is, every other difference that is less than δ in absolute terms has multiplicity 2) .

Moreover, if the starter sets are put either in a different order in the cells in row 1 of the array or they are put in the cells in column 1, say, instead of row 1, then the design obtained by this different arrangement of the sets is isomorphic to the former design. As noted earlier, if a different element from the set of differences (reduced modulo m) is allocated a higher multiplicity, then another set of pairs of treatments now appear more often as a block in the design, thereby producing a different design. In this circumstance, the 3rd set identified with the difference of higher multiplicity in the starter can be put in the last cell of row 1, which means a slight modification of the algorithmic procedure.

The QBD of the constructed design consists of a RGD and BIBD. Hence it is a BIBD extended design. The RGD component consists of blocks from a single column, $j \in \{\delta, \delta+1, m\}$, that is a column where each pair of treatments in it concurs a higher number of times, 3 in the overall design and its treatment concurrence counts are 0 and 1. Hence the RGD component contributes m blocks to the overall design. Similarly, the BIBD part consists of $m(m-1)$ blocks formed by the rest of the columns, $m - 1$ of them. The treatment concurrence counts in the BIBD part of the design is 2. Hence, in the overall design, each pair of treatments concur in either 2 or 3 blocks.

Example 5.7.1. Let $v = 5$. Then the sets $\{1, 2\}$, $\{4, 5\}$, $\{1, 3\}$, $\{2, 4\}$ and $\{3, 5\}$ with the differences $\pm1, \pm1, \pm2, \pm2, \text{and} \pm 2$, respectively, constitute the starter.

1	2	1	3	2	4	4	5	3	5
2	3	2	4	3	5	5	1	4	1
3	4	3	5	4	1	1	2	5	2
4	5	4	1	5	2	2	3	1	3
5	1	5	2	1	3	3	4	2	4

Figure 5.21: A $(5 \times 5)/2$ RGSLR for 5 treatments

1	2	3	5	1	4	2	3	4	5
2	3	4	1	2	5	3	4	5	1
3	4	5	2	3	1	4	5	1	2
4	5	1	3	4	2	5	1	2	3
5	1	2	4	5	3	1	2	3	4

Figure 5.22: Another $(5 \times 5)/2$ RGSLR for 5 treatments

Notice that $S_{11} = \{1, 2\}$, $S_{12} = \{4, 5\}$, $S_{21} = \{1, 3\}$, $S_{22} = \{2, 4\}$ and $S_{23} = \{3, 5\}$. Notice also that the difference ± 2 is of higher multiplicity, 3.

By the algorithmic procedure, we obtain the $(5 \times 5)/2$ RGSLR for 5 treatments as shown in Figure 5.21.

Remark. Supposing we decide to allocate the higher multiplicity, 3 to the differences ± 1 instead of ± 2, that is, we want a design for which any pair of treatments with the differences ± 1 concurs a higher number of times, 3. Then the starter set can be reformulated to suit this.

The sets, $\{1, 2\}$, $\{2, 3\}$, $\{4, 5\}$, $\{1, 4\}$ and $\{3, 5\}$ with the differences $\pm 1, \pm 1, \pm 1, \pm 2$, and $\pm$ 2, respectively, for instance, form another starter that can be used to obtain such design, which is another $(5 \times 5)/2$ RGSLR for 5 treatments, and is as shown in Figure 5.22

Example 5.7.2. Let $v = 7$. Then we obtain a $(7 \times 7)/2$ RGSLR for 7 treatments as shown in Figure 5.23 using the given procedure, where the starter comprises the sets $\{1, 2\}$, $\{1, 7\}$, $\{3, 5\}$, $\{4, 6\}$ $\{2, 5\}$, $\{3, 6\}$ and $\{4, 7\}$.

1	2	3	5	2	5	3	6	4	6	1	7	4	7
2	3	4	6	3	6	4	7	5	7	2	1	5	1
3	4	5	7	4	7	5	1	6	1	3	2	6	2
4	5	6	1	5	1	6	2	7	2	4	3	7	3
5	6	7	2	6	2	7	3	1	3	5	4	1	4
6	7	1	3	7	3	1	4	2	4	6	5	2	5
7	1	2	4	1	4	2	5	3	5	7	6	3	6

Figure 5.23: A $(7 \times 7)/2$ RGSLR for 7 treatments

Notice that the differences (reduced modulo 7) between the treatments in the starter sets are $\pm 1, \pm 1, \pm 2, \pm 2, \pm 3 \pm 3$ and $\pm$ 3, respectively, with the respective multiplicities being 2, 2, and 3 for the differences ± 1, ± 2 and ± 3.

Remark. The construction can be generalized for higher values of η as given in section 5.7.5.

5.7.5 Generalization of the construction for $(m \times \eta m)/2$ RGSLRs when $v = m$ is odd

We give a general construction for the aforementioned class of designs, where $\eta = 1, 2, \ldots, t$, $t \leq \delta$ and $\delta = (m-1)/2$. In particular, if m is an odd prime, then $t = \delta$. But if m is not an odd prime, then $t < \delta$. When $\eta = 1$, the parent/basic design is obtained which is utilized to obtain larger designs with higher values of η. Moreover, if $\eta = \delta$, then the construction gives a BSLR. As before, we use the cyclic group $\mathbb{Z}_m$ and regard $\mathbb{Z}_m$ as $\{1, 2, \ldots, m\}$.

Let the treatment set be denoted by $V = \{1, 2, \ldots, m\}$, For all $\sigma \in V$, to have a RGD, a sensible choice for its concurrence relationship with the other $m - 1$ treatments is governed by (5.4).

$$2\eta\lambda_1 + (m - 1 - 2\eta)\lambda_2 = 2m\eta \tag{5.4}$$

where $m \geq 2\eta + 1$.

From (5.4), λ_1 and λ_2 need to be $2\eta + 1$ and 2η, respectively. Hence, $\sigma \in V$ needs to concur with 2η treatments in $2\eta + 1$ blocks and with the other $m - 1 - 2\eta$ treatments in 2η blocks.

We give an algorithmic procedure for the generalized construction in the next section.

Remark. Notice that (5.4) can also be seen from $2\eta m = (m-1)2\eta + 2\eta = (m-1-2\eta)2\eta + 2\eta(2\eta+1)$ which is identical to $2\eta(2\eta+1)+(m-1-2\eta)2\eta$. Hence, to have a RGD, each treatment needs to have 2η concurrences of $2\eta+1$ and all the rest ($m-1-2\eta$ of them) equal to 2η. Thus if $m = 2\eta+1$, then each treatment will have precisely all concurrences being $2\eta+1$ and the QBD will be a BIBD making the constructed design a BSLR. Notice also that if $m = 2\eta+1$, then $\delta = \eta$, which is consistent with the condition above for obtaining a BSLR.

5.7.6 An algorithmic procedure for the generalized construction

1. Label the treatments 1, 2, ..., m.

2. Form m sets each of size 2 with the m treatments where each treatment appears 2 times among the sets and the differences (reduced modulo m) between the pairs of treatments in these sets are ± 1, ± 2, . . ., $\pm\delta$, the non-zero elements of $\mathbb{Z}_m$, whose multiplicities are 3 for $\pm\delta$ and 2 for others, and these form a starter for the cyclic group $\mathbb{Z}_m$.

3. Create an $m \times \eta m$ array and label its rows $i = 1, 2, ..., m$ and its columns $j = 1, 2, ..., \eta m$.

4. For all $u = 1, 2, \ldots, \delta$, and

$$l = \begin{cases} 1,2,3 & \text{if } u = \delta, \\ 1,2 & \text{if } u < \delta. \end{cases}$$

   put S_{ul} in the cell in position $(1, j)$ of the array, that is, the cell in row 1 and column j, where S_{ul} denotes the lth set in the starter for which the difference (reduced modulo m) between its elements is $\pm u$, where

$$j = \begin{cases} \delta, \delta+1, m & \text{if } u = \delta, \\ u, m-u & \text{if } u < \delta. \end{cases}$$

5. For $j = 1, 2, ..., m$, develop the block in position $(1, j)$, cyclically, via successive addition of $1 \pmod m$, thereby generating the m blocks in each column. Stop here if a design with $\eta = 1$ is required.

6. If $\eta > 1$, then denote the design obtained in step 5 by Λ_1. Find $\eta - 1$ distinct generators of $\mathbb{Z}_m$ and denote them by $\alpha_1, \alpha_2, \ldots, \alpha_{\eta-1}$ corresponding to the order they are to be used in permuting the treatments, viz, α_1 is the first to be used and $\alpha_{\eta-1}$, the last, where $\alpha_s \neq \pm\alpha_0, \pm\alpha_1, \ldots, \pm\alpha_{s-1}$ for all $s = 1, 2, \ldots, \eta-1$ and $\alpha_0 = 1$.

1 2	1 3	2 4	4 5	3 5	2 4	2 1	4 3	3 5	1 5
2 3	2 4	3 5	5 1	4 1	4 1	4 3	1 5	5 2	3 2
3 4	3 5	4 1	1 2	5 2	1 3	1 5	3 2	2 4	5 4
4 5	4 1	5 2	2 3	1 3	3 5	3 2	5 4	4 1	2 1
5 1	5 2	1 3	3 4	2 4	5 2	5 4	2 1	1 3	4 3

Figure 5.24: A $(5 \times 10)/2$ RGSLR for 5 treatments

7. For all $s = 1, 2, \ldots, \eta - 1$, apply α_s to Λ_1 via multiplication by the treatments within its blocks to obtain Λ_q, where $q = 1 + s$ and Λ_q is the qth constituent design, $q = 2, \ldots, \eta$.

Comments. The overall design consists of η constituent designs, $\Lambda_1, \Lambda_2, \ldots, \Lambda_\eta$. Each constituent design is an $(m \times m)/2$ RGSLR for m treatments and has a QBD consisting of a RGD (having m blocks) and a BIBD (having $m(m-1)$ blocks). The overall design consists of $m\eta$ blocks in the RGD component of its QBD. Similarly, the BIBD component contributes $m\eta(m-1)$ blocks to the overall design. Hence the QBD of the overall design is BIBD-extended, where the BIBD component is the extension.

Moreover, the restrictions on choosing the permutations $\alpha_1, \alpha_2, \ldots, \alpha_{\eta-1}$ is to ensure that the permutations do not result in blocks with the same pairs concurring a higher number of times in more than 1 constituent design such that with the various constituent designs, the overall design remains a RGD.

Since the design contains ηm columns, then if all the starter sets were to be obtained, it would require ηm sets, where each set generates a column and the multiplicities of the differences is as follows. Notice that $\eta m = \eta[2(\delta-\eta)+(2\eta+1)] = 2\eta(\delta-\eta)+(2\eta+1)\eta$. Hence the starter sets for the overall design contain treatment pairs with δ distinct differences where η of these differences are of multiplicity $2\eta + 1$ each, while the multiplicity of the other $\delta - \eta$ of them is 2η each.

Example 5.7.3. Let $v = 5$ and $\eta = 2$. Then by implementing the algorithm, using the generator, $\alpha_1 = 2$, we obtain a $(5 \times 10)/2$ RGSLR as shown in Figure 5.24.

Remark. Notice that each pair with the differences ± 1 concurs 2 times in Λ_1 and 3 times in Λ_2. Similarly, each pair with the differences ± 2 concurs 3 times in Λ_1. and 2 times in Λ_2.

Furthermore, the treatment concurrence counts in the RGD part in the overall design is 1 for each pair of treatments (which comprises the concurrences from the RGDs in both

1 2	3 5	2 5	3 6	4 6	1 7	4 7	2 4	6 3	4 3	6 5	1 5	2 7	1 7
2 3	4 6	3 6	4 7	5 7	2 1	5 1	4 6	1 5	6 5	1 7	3 7	4 2	3 2
3 4	5 7	4 7	5 1	6 1	3 2	6 2	6 1	3 7	1 7	3 2	5 2	6 4	5 4
4 5	6 1	5 1	6 2	7 2	4 3	7 3	1 3	5 2	3 2	5 4	7 4	1 6	7 6
5 6	7 2	6 2	7 3	1 3	5 4	1 4	3 5	7 4	5 4	7 6	2 6	3 1	2 1
6 7	1 3	7 3	1 4	2 4	6 5	2 5	5 7	2 6	7 6	2 1	4 1	5 3	4 3
7 1	2 4	1 4	2 5	3 5	7 6	3 6	7 2	4 1	2 1	4 3	6 3	7 5	6 5

Figure 5.25: A $(7 \times 14)/2$ RGSLR for 7 treatments

Λ_1 and Λ_2 which are 0 and 1 in each–noting that there is no repetition of blocks in the 2 RGDs, whose overall blocks consist of all the possible pairs of treatments exactly once, each), hence the RGD part is a BIBD. Similarly, for the BIBD part, the concurrence counts is 4 (2 from each of Λ_1 and Λ_2). Hence, the treatment concurrence counts is 5 for each pair of treatments in the overall design.

Notice that the RGD part in Λ_1 is constituted by the pairs of treatments with the differences ± 2 which can be seen to be the pairs in any of columns 2, 3 and 5. Similarly, pairs with the differences ± 1 constitute the RGD part in Λ_2: see the pairs in any of the 2nd, 3rd and 5th columns that Λ_2 occupies or equivalently, any of columns 7, 8 and 10 of the overall design. Hence the cells in columns 5 and 10, for instance, constitute the RGD component of the overall design (which is a BIBD), and the cells from the rest of the columns constitute another BIBD component. Hence, the QBD of the overall design is a BIBD.

The QBD of the design is the same as that of the $(5 \times 10)/2$ BSLR shown in Figure 3.2 in Chapter 3 of this thesis.

Example 5.7.4. Let $v = 7$ and $\eta = 2$. Then the design obtained via the algorithmic procedure is as shown in Figure 5.25 if the generator $\alpha_1 = 2$ is used to permute the treatments in Λ_1 to obtain Λ_2. The design obtained is a $(7 \times 14)/2$ RGSLR for 7 treatments.

Example 5.7.5. Let $v = 7$ and $\eta = 3$. Then the construction gives a $(7 \times 21)/2$ RGSLR as shown in Figure 5.26, if the generators $\alpha_1 = 2$ and $\alpha_2 = 3$ are the permutations applied to Λ_1 to obtain Λ_2 and Λ_3, respectively.

Remark. Notice that the QBD of the $(7 \times 21)/2$ RGSLR for 7 treatments shown in Figure 5.26 is BIBD-extended; consisting of a RGD and a BIBD as the extension.

1 2	3 5	2 5	3 6	4 6	1 7	4 7	2 4	6 3	4 3	6 5	1 5	2 7	1 7	3 6	2 1	6 1	2 4	5 4	3 7	5 7
2 3	4 6	3 6	4 7	5 7	2 1	5 1	4 6	1 5	6 5	1 7	3 7	4 2	3 2	6 2	5 4	2 4	5 7	1 7	6 3	1 3
3 4	5 7	4 7	5 1	6 1	3 2	6 2	6 1	3 7	1 7	3 2	5 2	6 4	5 4	2 5	1 7	5 7	1 3	4 3	2 6	4 6
4 5	6 1	5 1	6 2	7 2	4 3	7 3	1 3	5 2	3 2	5 4	7 4	1 6	7 6	5 1	4 3	1 3	4 6	7 6	5 2	7 2
5 6	7 2	6 2	7 3	1 3	5 4	1 4	3 5	7 4	5 4	7 6	2 6	3 1	2 1	1 4	7 6	4 6	7 2	3 2	1 5	3 5
6 7	1 3	7 3	1 4	2 4	6 5	2 5	5 7	2 6	7 6	2 1	4 1	5 3	4 3	4 7	3 2	7 2	3 5	6 5	4 1	6 1
7 1	2 4	1 4	2 5	3 5	7 6	3 6	7 2	4 1	2 1	4 3	6 3	7 5	6 5	7 3	6 5	3 5	6 1	2 1	7 4	2 4

Figure 5.26: A $(7 \times 21)/2$ RGSLR for 7 treatments

Notice also that this design has the same QBD as the $(7 \times 21)/2$ BSLR in Figure 3.3 in Chapter 3 of this thesis.

Comment. Now, we have found a concept called undirected terrace to be a useful and efficient technique that we can utilize to obtain the starter sets for the case where $v = m$ is odd which enhances the construction.

5.8 Another approach to obtaining the general construction via undirected terrace

We employ the concept known as undirected terrace to obtain the starter sets for the generalized construction of $(m \times \eta m)/2$ RGSLRs when $v = m$ is odd. This provides a more convenient and efficient means for obtaining these sets for odd values of m. As in the previous section, we use the cyclic additive group, $\mathbb{Z}_m$ for the integers modulo m and regard $\mathbb{Z}_m$ to be the set $\{1, \ldots, m\}$.

5.8.1 Undirected terrace and associated starter sets

Let $m \in \mathbb{Z}_+$, where m is odd. Then the sequence $1,\ m,\ 2,\ m-1,\ 3,\ m-2, \ldots, (m+1)/2$ forms an undirected terrace for $\mathbb{Z}_m$: see, for example, Bailey (1984) and Durier et al. (1997) for discussions on this concept.

Some small examples include 1, 5, 2, 4, 3; 1, 7, 2, 6, 3, 5, 4; and 1, 9, 2, 8, 3, 7, 4, 6, 5 which are undirected terraces for $\mathbb{Z}_5$, $\mathbb{Z}_7$ and $\mathbb{Z}_9$, respectively.

By writing down the plus/minus differences between each adjacent (successive) pair, modulo m, and considering the row as a circle that joins up the two ends, we have

$\pm 1, \pm 2, \ldots, \pm(m-1)/2, \pm(m-1)/2, \pm(m-3)/2, \ldots, \pm 1, \pm(m-1)/2$. For instance, in the example for $\mathbb{Z}_7$, we get $\pm 1, \pm 2, \pm 3, \pm 3, \pm 2, \pm 1, \pm 3$. Hence the pairs of elements which give these differences, which are the sets $\{1,7\}$, $\{7,2\}$, $\{2,6\}$, $\{6,3\}$, $\{3,5\}$, $\{5,4\}$, and $\{4,1\}$, respectively constitute a starter. Similarly, the corresponding starter sets based on the undirected terraces for $\mathbb{Z}_5$ and $\mathbb{Z}_9$ given above are $\{1,5\}$, $\{5,2\}$, $\{2,4\}$, $\{4,3\}$, $\{3,1\}$ with the differences being $\pm 1, \pm 2, \pm 2, \pm 1, \pm 2$ and $\{1,9\}$, $\{9,2\}$, $\{2,8\}$, $\{8,3\}$, $\{3,7\}$, $\{7,4\}$, $\{4,6\}$, $\{6,5\}$, $\{5,1\}$ with the differences $\pm 1, \pm 2, \pm 3, \pm 4, \pm 4$, $\pm 3, \pm 2$, $\pm 1, \pm 4$, respectively.

5.8.2 Procedure

We obtain the basic design in a similar manner as before but using the starter sets obtained from the undirected terrace and then utilize the basic design to obtain larger designs. As before, we denote the basic design Λ_1. In certain situations, to obtain a larger design, we use only one generator of $\mathbb{Z}_m$ (an element of $\mathbb{Z}_m$ that is coprime to m), α, say, where $\alpha \neq 1, m-1$. We multiply (successively), each treatment in Λ_l by α to obtain Λ_{l+1} (where Λ_l is the lth constituent design) for all $l = 1, \ldots, \eta - 1$, that is, we apply α to Λ_1 to obtain Λ_2, and similarly to Λ_2 to obtain Λ_3, and so on until the overall design is generated. In other cases we use more than one generator.

If m is an odd prime, then we obtain designs for values of $\eta = 1, 2, \ldots, t$, where $t = \delta$, and $\delta = (m-1)/2$ using a single generator, α. Moreover, to obtain a design with $\eta > \delta$, we use more than one generator. Let Λ_1, . . ., Λ_η denote the constituent designs for the construction, where Λ_1 is the basic design obtained when $\eta = 1$.

Let $(y-1)\delta + 1 \leq \eta \leq y\delta$, where $y = \lceil \eta/\delta \rceil$, $\eta \geq 1$. Denote y generators of $\mathbb{Z}_m$ by $\alpha_1, \ldots, \alpha_y$, where $\alpha_t \neq 1, m-1$ for any $t \in \{1, \ldots, y\}$. If $y = 1$, then it implies that $1 \leq \eta \leq \delta$, hence we use one generator, $\alpha_1 = \alpha$, say (any generator that satisfies the aforementioned condition can be used). Thus, we multiply each treatment in Λ_1 by α to obtain Λ_2. Similarly, we multiply each treatment in Λ_2 by α to obtain Λ_3; and so on until each treatment in $\Lambda_{\eta-1}$ is multiplied by α to obtain Λ_η. This generates the overall design under construction.

Similarly, if $y = 2$, then it follows that $\delta+1 \leq \eta \leq 2\delta$ and we use two generators, α_1 and α_2, where α_1 is used in the same manner (successively) as α to obtain Λ_2, . . ., Λ_δ; while α_2 is then applied to every treatment in Λ_1 to obtain $\Lambda_{\delta+1}$, then α_2 is applied to every treatment in $\Lambda_{\delta+1}$ to obtain $\Lambda_{\delta+2}$, and this continues until Λ_η is obtained by multiplying every treatment in $\Lambda_{\eta-1}$ by α_2. The successive procedure continues depending on the value of y until the generator, α_y has been applied to every treatment in Λ_1 to obtain $\Lambda_{(y-1)\delta+1}$, then α_y on every treatment in $\Lambda_{(y-1)\delta+1}$ to obtain $\Lambda_{(y-1)\delta+2}$, and so on until α_y is used to multiply every treatment in $\Lambda_{\eta-1}$ to obtain Λ_η, hence the overall design generated.

On the other hand, if m is not an odd prime, then we obtain designs for values of $\eta = 1, 2, \ldots, t$, where $t < \delta$ and whose value is given by $t = \frac{1}{2}\varphi(m)$, where $\varphi(m) = m \prod_{p|m} (1 - 1/p)$, where the product is taken over the distinct prime numbers that divide m, and $\varphi(m)$ is called Euler's totient function or Euler's phi function, which gives the number of integers in the range 1 to m that are coprime to m. In this situation, to obtain designs for values of $\eta > 1$, we use one generator of $\mathbb{Z}_m$, α, say, where $\alpha \neq 1, m-1$ and as before, any generator that satisfies this condition can be used. Thus α is applied in a successive manner as before by multiplying every treatment in Λ_1 by α to obtain Λ_2; then α is also imposed in a similar manner on the treatments in Λ_2 to obtain Λ_3; and so on until it is multiplied by every treatment in $\Lambda_{\eta-1}$ to obtain Λ_η, thus generating the overall design.

We give in the next section an algorithmic procedure for the construction.

5.8.3 An algorithmic procedure for the Generalized construction via undirected terrace

1. Label the treatments 1, 2, ..., m.

2. Form a sequence of the elements of $\mathbb{Z}_m$ that constitutes an undirected terrace for $\mathbb{Z}_m$, viz, $1,\ m,\ 2,\ m{-}1,\ 3,\ m{-}2,\ldots (m{+}1)/2$. Consider the row/sequence as a circle that joins up the two ends and then combine each adjacent (successive) pair of elements such that the differences between these pairs, modulo m, are $\pm1, \pm2, \ldots, \pm(m-3)/2, \pm(m-1)/2, \pm(m-1)/2, \pm(m-3)/2, \ldots, \pm1, \pm(m-1)/2$, thus forming m starter 2-subsets for $\mathbb{Z}_m$. Label the starter sets S_1, S_2, . . ., S_m in the order these differences are listed.

3. Create an $m \times \eta m$ array and label its rows $i = 1, 2, ..., m$ and its columns $j = 1, 2, ..., \eta m$.

4. For $i = 1$ and $j = 1, \ldots, m$, put in the cell in position $(1, j)$ of the array, the jth set, S_j from the starter.

5. For $j = 1, 2, ..., m$, develop the block in position $(1, j)$, cyclically, via successive addition of 1 modulo m, thereby generating the m blocks in each column. Stop here if a design with $\eta = 1$ is required.

6. If $\eta > 1$, then denote the design obtained in step 5 by Λ_1 and let $\Lambda_2, \ldots, \Lambda_\eta$ denote the other constituent designs that make the overall design under construction. Let $y = \lceil \eta/\delta \rceil$ and $(y-1)\delta + 1 \leq \eta \leq y\delta$, where $\delta = (m-1)/2$. Let $\alpha_1, \ldots, \alpha_y$ be y generators of $\mathbb{Z}_m$, where $\alpha_t \neq 1, m-1$ for any $t \in \{1, \ldots, y\}$. If $y = 1$, which implies that $\eta \leq \delta$, then multiply each treatment in Λ_l by $\alpha_1 = \alpha$, say, to obtain Λ_{l+1} for all $l = 1, \ldots, \eta - 1$, thus generating the overall design.

7. If $y > 1$, which implies $\eta > \delta$ (involving designs with an odd prime value of m in our construction), then multiply each treatment in Λ_l by α_1 to obtain Λ_{l+1} for all $l = 1, \ldots, \delta - 1$. Similarly, multiply each treatment in Λ_1 by α_2 to obtain $\Lambda_{\delta+1}$ and then multiply each treatment in $\Lambda_{\delta+l}$ by α_2 to obtain $\Lambda_{\delta+l+1}$ for all $l = 1, \ldots, \delta - 1$.

1	7	2	6	3	5	4	2	7	4	5	6	3	1	4	7	1	3	5	6	2	3	7	6	4	2	1	5
7	2	6	3	5	4	1	7	4	5	6	3	1	2	7	1	3	5	6	2	4	7	6	4	2	1	5	3
2	1	3	7	4	6	5	4	2	6	7	1	5	3	1	4	5	7	2	3	6	6	3	2	7	5	4	1
1	3	7	4	6	5	2	2	6	7	1	5	3	4	4	5	7	2	3	6	1	3	2	7	5	4	1	6
3	2	4	1	5	7	6	6	4	1	2	3	7	5	5	1	2	4	6	7	3	2	6	5	3	1	7	4
2	4	1	5	7	6	3	4	1	2	3	7	5	6	1	2	4	6	7	3	5	6	5	3	1	7	4	2
4	3	5	2	6	1	7	1	6	3	4	5	2	7	2	5	6	1	3	4	7	5	2	1	6	4	3	7
3	5	2	6	1	7	4	6	3	4	5	2	7	1	5	6	1	3	4	7	2	2	1	6	4	3	7	5
5	4	6	3	7	2	1	3	1	5	6	7	4	2	6	2	3	5	7	1	4	1	5	4	2	7	6	3
4	6	3	7	2	1	5	1	5	6	7	4	2	3	2	3	5	7	1	4	6	5	4	2	7	6	3	1
6	5	7	4	1	3	2	5	3	7	1	2	6	4	3	6	7	2	4	5	1	4	1	7	5	3	2	6
5	7	4	1	3	2	6	3	7	1	2	6	4	5	6	7	2	4	5	1	3	1	7	5	3	2	6	4
7	6	1	5	2	4	3	7	5	2	3	4	1	6	7	3	4	6	1	2	5	7	4	3	1	6	5	2
6	1	5	2	4	3	7	5	2	3	4	1	6	7	3	4	6	1	2	5	7	4	3	1	6	5	2	7

Figure 5.27: A $(7 \times 28)/2$ RGSLR for 7 treatments

Continue this successive procedure until α_y is used to multiply every treatment in Λ_1 to obtain $\Lambda_{(y-1)\delta+1}$ and subsequently each treatment in $\Lambda_{(y-1)\delta+l}$ is multiplied by α_y to obtain $\Lambda_{(y-1)\delta+l+1}$ for all $l = 1, \ldots, \eta - 1 - (y-1)\delta$. This generates the overall design under construction.

Example 5.8.1. Let $m = 7$ and $\eta = 4$. Then by the algorithmic procedure we obtain a $(7 \times 28)/2$ RGSLR shown in Figure 5.27, where every treatment in Λ_l was multiplied by $\alpha_1 = 2$ to obtain Λ_{l+1} for $l = 1, 2$. Similarly, every treatment in Λ_1 was multiplied by $\alpha_2 = 3$ to obtain Λ_4.

Example 5.8.2. Let $m = 9$ and $\eta = 3$. Then by the algorithmic procedure we obtain a $(9 \times 27)/2$ RGSLR shown in Figure 5.28 where each treatment in Λ_l was multiplied by $\alpha = 5$ to obtain Λ_{l+1} for all $l = 1, 2$.

On the basis of our new method of obtaining the general construction which utilizes the concept, undirected terrace, we give an enlarged table of starters in $\mathbb{Z}_m$ for various odd values of m shown in Table 5.2. However, for lack of space, the corresponding starter sets for $\mathbb{Z}_{15}$ is omitted from Table 5.2 and presented, here, separately, viz, $\{1, 15\}$, $\{15, 2\}$, $\{2, 14\}$, $\{14, 3\}$, $\{3, 13\}$, $\{13, 4\}$, $\{4, 12\}$, $\{12, 5\}$, $\{5, 11\}$, $\{11, 6\}$, $\{6, 10\}$, $\{10, 7\}$, $\{7, 9\}$, $\{9, 8\}$, $\{8, 1\}$.

5.8.4 Realizing a BSLR from the construction

Note that the design obtained via the construction in section 5.8.3 is an $(m \times \eta m)/2$ RGSLR for m treatments, where m is odd and η corresponds to the number of constituent

1 9	9 2	2 8	8 3	3 7	7 4	4 6	6 5	5 1	5 9	9 1	1 4	4 6	6 8	8 2	2 3	3 7	7 5	7 9	9 5	5 2	2 3	3 4	4 1	1 6	6 8	8 7
2 1	1 3	3 9	9 4	4 8	8 5	5 7	7 6	6 2	1 5	5 6	6 9	9 2	2 4	4 7	7 8	8 3	3 1	5 7	7 3	3 9	9 1	1 2	2 8	8 4	4 6	6 5
3 2	2 4	4 1	1 5	5 9	9 6	6 8	8 7	7 3	6 1	1 2	2 5	5 7	7 9	9 3	3 4	4 8	8 6	3 5	5 1	1 7	7 8	8 9	9 6	6 2	2 4	4 3
4 3	3 5	5 2	2 6	6 1	1 7	7 9	9 8	8 4	2 6	6 7	7 1	1 3	3 5	5 8	8 9	9 4	4 2	1 3	3 8	8 5	5 6	6 7	7 4	4 9	9 2	2 1
5 4	4 6	6 3	3 7	7 2	2 8	8 1	1 9	9 5	7 2	2 3	3 6	6 8	8 1	1 4	4 5	5 9	9 7	8 1	1 6	6 3	3 4	4 5	5 2	2 7	7 9	9 8
6 5	5 7	7 4	4 8	8 3	3 9	9 2	2 1	1 6	3 7	7 8	8 2	2 4	4 6	6 9	9 1	1 5	5 3	6 8	8 4	4 1	1 2	2 3	3 9	9 5	5 7	7 6
7 6	6 8	8 5	5 9	9 4	4 1	1 3	3 2	2 7	8 3	3 4	4 7	7 9	9 2	2 5	5 6	6 1	1 8	4 6	6 2	2 8	8 9	9 1	1 7	7 3	3 5	5 4
8 7	7 9	9 6	6 1	1 5	5 2	2 4	4 3	3 8	4 8	8 9	9 3	3 5	5 7	7 1	1 2	2 6	6 4	2 4	4 9	9 6	6 7	7 8	8 5	5 1	1 3	3 2
9 8	8 1	1 7	7 2	2 6	6 3	3 5	5 4	4 9	9 4	4 5	5 8	8 1	1 3	3 6	6 7	7 2	2 9	9 2	2 7	7 4	4 5	5 6	6 3	3 8	8 1	1 9

Figure 5.28: A $(9 \times 27)/2$ RGSLR for 9 treatments

SLRs used in the construction. If m is an odd prime and $\eta = \delta$, where $\delta = (m-1)/2$, then the construction gives a BSLR. This can be seen as follows.

Notice that if $\eta = 1$, we have an $(m \times m)/2$ RGSLR for m treatments, which is the basic design. Denote this basic design by Λ_1. Note that the starter used in the construction of Λ_1 consists of m sets, where $m = 2\delta + 1$. Let $D = \{\pm 1, \ldots, \pm\delta\}$ denote the set of differences (modulo m) between the elements contained in the starter sets, where the cardinality of D, $|D| = 2\delta = m - 1$. The elements of D correspond to the non-zero elements of $\mathbb{Z}_m$ and their multiplicities in the starter are 3 (higher) for each of $\pm\delta$ and 2 (lower) for the rest, as can be noticed in step 2 of the procedure in section 5.8.3. The multiplicity of each element of D in the starter corresponds to the number of sets in the starter that contain elements whose difference (modulo m) gives the specified element of D. This also corresponds to the treatment concurrence counts for any pair of treatments in Λ_1 whose difference (modulo m) is the specified element of D.

Columns $1, \ldots, \delta$ of Λ_1 are generated from a series of δ starter sets whose elements have the differences $\pm 1, \ldots, \pm\delta$, respectively. Similarly, another series of δ starter sets with the differences between their elements being $\pm\delta, \ldots, \pm 1$ are used to generate columns $\delta + 1, \ldots, 2\delta$, respectively. Finally, the remaining starter set for which the differences between its elements are $\pm\delta$ is used to generate column m. Moreover, each starter set

Table 5.2: Starters in $\mathbb{Z}_m$ for more odd values of m

m	starter
5	$\{1,5\}$ $\{5,2\}$ $\{2,4\}$ $\{4,3\}$ $\{3,1\}$
7	$\{1,7\}$ $\{7,2\}$ $\{2,6\}$ $\{6,3\}$ $\{3,5\}$ $\{5,4\}$ $\{4,1\}$
9	$\{1,9\}$ $\{9,2\}$ $\{2,8\}$ $\{8,3\}$ $\{3,7\}$ $\{7,4\}$ $\{4,6\}$ $\{6,5\}$ $\{5,1\}$
11	$\{1,11\}\{11,2\}\{2,10\}\{10,3\}\{3,9\}$ $\{9,4\}$ $\{4,8\}$ $\{8,5\}$ $\{5,7\}$ $\{7,6\}$ $\{6,1\}$
13	$\{1,13\}\{13,2\}\{2,12\}\{12,3\}\{3,11\}\{11,4\}\{4,10\}\{10,5\}\{5,9\}$ $\{9,6\}$ $\{6,8\}$ $\{8,7\}$ $\{7,1\}$

generates m distinct blocks, which form a column in Λ_1. Thus there are $m\delta$ distinct blocks in columns 1 to δ, which form a BIBD with treatment concurrence counts, unity. Similarly, there are $m\delta$ distinct blocks (the same set of overall blocks as before) in columns $\delta+1$ to 2δ, which also form a BIBD with treatment concurrence counts, unity. However, the last column, column m, contains m distinct blocks which form a RGD with treatment concurrence counts 0 and 1 (assuming $m > 3$). In particular, if $m = 3$, then column m consists of all the possible pairs of 3 treatments, each pair appearing exactly once (since there are 3 distinct pairs, where 3 coincides with $\binom{3}{2}$), thus forming a BIBD with treatment concurrence count 1. Hence the QBD of Λ_1 consists of a BIBD with treatment concurrence counts 2 and a RGD with treatment concurrence counts 0 and 1 (if $m > 3$), giving a BIBD-extended RGD. However, in the case where $m = 3$, the QBD of Λ_1 is trivially a BIBD with treatment concurrence counts, 3. In this case, let $\Lambda_1 = \Lambda$.

Moreover, each row and each column of Λ_1 (or Λ) contains each treatment exactly two times (an integer number of times). Hence Λ_1 is a RGSLR, since its QBD is a RGD while Λ is a BSLR, as its QBD is a BIBD.

Suppose $m > 3$ and $\eta > 1$. Call Λ_1 a constituent design in the BSLR, whose realization from the construction is sought. Each stage of permutation (involving multiplication) described in step 6 of the procedure in section 5.8.3 results in another constituent RGSLR. The QBD of the resulting constituent RGSLR also comprises a BIBD with the same overall blocks as the BIBD in Λ_1, hence same treatment concurrence counts, 2 and also a RGD (which are the blocks in its last column) with concurrence counts 0 and 1. Also, at each stage of permutation, a different pair of elements in D (elements with $\pm$ sign before them) become the difference identified with each of the m pairs of elements that make the last column of the particular constituent design generated. This continues until (if possible–depending on the value of η) all the pairs of elements in D that are relatively prime to m are identified in a similar manner. Note that after all the elements in D which are relatively prime to m have been identified in a similar manner, then for higher values of η

there are repeats.

Now, let m be an odd prime and $\eta = \delta$, then overall, by the permutations, each of the δ pairs of elements (involving $\pm$ sign) in D would have taken their turn of becoming the difference attributed to m distinct pairs of elements that constitute the blocks in the last column of the constituent design (a column with pairs of treatments of higher concurrences, 3) generated at a given stage of permutation. This is so since for prime m, each non-zero element of $\mathbb{Z}_m$ (which is an element of D) is relatively prime to m. Hence the $m\delta$ distinct blocks in the last columns of the δ constituent designs constitute a BIBD with $\lambda = 1$, which makes the QBD of the overall design to consist of δ copies of a BIBD each with $\lambda = 2$ and another BIBD with $\lambda = 1$ such that the QBD of the overall design is a BIBD whose treatment concurrence counts is $\lambda = 2\delta + 1$.

Moreover, we remind that each constituent design is a SLR (RGSLR), where each treatment appears twice per row and per column. Note that each row of the overall design is constituted by the corresponding rows in the δ constituent designs while each column in any of the constituent designs is also a column in the overall design. hence each treatment in the overall design appears 2δ times per row and twice per column (integer number of times). Thus the overall design obtained from our construction of RGSLR in this special case has a QBD that is a BIBD. In the special case that the QBD of a RGSLR is a BIBD, the resulting design is a BSLR.

Hence, if m is an odd prime and $\eta = \delta$, then the construction described in section 5.8.3 produces a BSLR.

Example 5.8.3. If $m = 3$, then $\delta = 1$. Let $\eta = 1$. Then the design resulting from the construction is a $(3 \times 3)/2$ BSLR for 3 treatments. Similarly, iif $m = 5$, then $\delta = 2$ such that if $\eta = 2$, we have a $(5 \times 10)/2$ BSLR for 5 treatments from the construction.

5.9 More RGSLRs of large sizes

In a similar manner to the case where v was even, when v is odd, RGSLRs of large sizes whose QBDs are BIBD-extended can also be obtained by extending a RGSLR with a BSLR. This involves putting a RGSLR and a BSLR of corresponding sizes in an array of appropriate size. The resulting design is also a RGSLR: see Theorem 5.1.1.

The aforementioned procedure is applicable if and only if the RGSLR and BSLR used for the construction conform in size and set of treatments, that is, if they have the same number of rows (or columns) and same block size and both of them contain the same treatments.

BSLRs for odd number of treatments whose constructions can be found in Chapter 3 of this thesis can be useful for such construction when there exist both designs of conformable sizes.

Example 5.9.1. A $(5 \times 15)/2$ RGSLR for 5 treatments can be obtained by putting a

$(5 \times 5)/2$ RGSLR and a $(5 \times 10)/2$ BSLR where each of them contains the same set of 5 treatments.

1 2	1 3	2 4	4 5	3 5	1 2	1 3	2 4	4 5	3 5	2 4	2 1	4 3	3 5	1 5
2 3	2 4	3 5	5 1	4 1	2 3	2 4	3 5	5 1	4 1	4 1	4 3	1 5	5 2	3 2
3 4	3 5	4 1	1 2	5 2	3 4	3 5	4 1	1 2	5 2	1 3	1 5	3 2	2 4	5 4
4 5	4 1	5 2	2 3	1 3	4 5	4 1	5 2	2 3	1 3	3 5	3 2	5 4	4 1	2 1
5 1	5 2	1 3	3 4	2 4	5 1	5 2	1 3	3 4	2 4	5 2	5 4	2 1	1 3	4 3

Figure 5.29: A $(5 \times 15)/2$ RGSLR for 5 treatments

Remark. The deign in Figure 5.29 involves adding the RGSLR in Figure 5.24 (which is precisely a $(5 \times 10)/2$ BSLR) to the $(5 \times 5)/2$ RGSLR in Figure 5.21, where each of them contains 5 treatments on the same set.

However, any other BSLR on the same set of 5 treatments that conforms in size, such as the design in Figure 3.2 in Chapter 3 of this thesis can also be used.

Chapter 6

Non-balanced Semi-Latin Rectangles with Larger Block Sizes

6.1 Introduction

This chapter considers semi-Latin rectangles (SLRs) whose row-column intersections (blocks) contain k treatments, where $k > 2$ and whose quotient block designs (QBDs) are not combinatorially balanced. As in the preceding chapter, we concentrate on SLRs whose QBDs are regular graph designs (RGDs), thus giving regular-graph semi-Latin rectangles (RGSLRs). We give constructions for RGSLRs of some small sizes whose QBDs are BIBD-extended. Some concepts such as undirected terrace are employed to obtain the designs. We also exploit the constructions given in Bailey and Monod (2001). Some designs are also obtained via block complementation. Moreover, adjoining BSLRs to an already obtained RGSLR, or adjoining the RGSLR to another after a suitable permutation of treatments, gives designs for larger sizes, just as in Chapter 5.

6.2 Construction of a $(5 \times 5)/3$ RGSLR for $v = 5$ treatments

We denote the set of treatments by $V = \{1, \ldots, 5\}$ and use the cyclic group $\mathbb{Z}_5$, the integers modulo 5 for the construction, where we regard $\mathbb{Z}_5$ as $\{1, \ldots, 5\}$. We start by obtaining a starter set using undirected terrace in a similar manner as in section 5.8.1 of Chapter 5. There are 5 sets that constitute the starter, but this time, since $k = 3$, each starter set is a 3-subset of $\mathbb{Z}_5$. Moreover, since for a design of this size, each treatment needs to appear 3 times in each row, that is, the parameter $n_r = 3$ and we are interested in putting the m starter sets in a single (the initial) row of an array of appropriate size, then the starter sets need to be a 3-resolution class, thus, each treatment needs to appear 3 times among the m sets, that is, in 3 sets.

We remind that the sequence $1,\ m,\ 2,\ m-1,\ 3,\ m-2,\ \ldots,(m+1)/2$ constitutes an undirected terrace for $\mathbb{Z}_m$. Thus 1, 5, 2, 4, 3 constitutes an undirected terrace for $\mathbb{Z}_5$. By considering the row/sequence as a circle that joins up the two ends and then extending the earlier procedure, viz, combining every 3 successive elements, we obtain the starter sets to be $\{1,5,2\}$, $\{5,2,4\}$, $\{2,4,3\}$, $\{4,3,1\}$ and $\{3,1,5\}$.

The procedure involves putting the starter sets in the 5 cells in the initial row of a 5×5 array and then developing the block formed by each starter set, cyclically, via addition of 1, modulo 5, to generate entries to fill succeeding cells in its column. We summarize the procedure for the construction in section 6.2.1.

6.2.1 Procedure for the construction

1. Denote the treatment set by $V=\{1,\ldots,5\}$.

2. Form a sequence of elements of $\mathbb{Z}_5$ that constitute an undirected terrace for $\mathbb{Z}_5$ and then using the elements of this sequence, obtain a 3-subset starter consisting of 5 sets by combining every 3 elements of the sequence in succession, regarding the sequence as a circle that joins up the two ends; and number the starter sets S_j, $j=1,\ldots,5$.

3. Create a 5×5 array and label its rows $i=1,\ldots,5$ and columns $j=1,\ldots,5$.

4. For $j=1,\ldots,5$, put S_j in the cell in position $(1,j)$ of the array and develop the block formed by S_j, cyclically, via addition of 1, modulo 5, thereby forming column j, for all j.

Comments. The QBD of the design under construction is a RGD if and only if its concurrence relationship is given by

$$2\lambda_1+2\lambda_2=30 \tag{6.1}$$

which gives the solution set $\{\lambda_1,\lambda_2\}=\{7,8\}$. Hence each treatment needs to concur 7 times with 2 treatments, each and 8 times with each of the other 2 treatments.

Notice that the pairwise differences, modulo 5, resulting from the pairs within the triples that make the starter sets are $\pm1,\pm2,\pm1$; $\pm2,\pm2,\pm1$; $\pm2,\pm1,\pm1$; $\pm1,\pm2,\pm2$; and $\pm2,\pm1,\pm2$ for S_j, $j=1,\ldots,5$, respectively. Hence, the overall multiplicities of the differences are 7 and 8 for ±1 and ±2, respectively. Moreover, these differences consist of all the non-zero elements of $\mathbb{Z}_5$.

Notice also that the multiplicities of the pairwise differences have identical values as λ_1 and λ_2, which must be so. The value, 30 on the right hand side of (6.1) is the sum of concurrences with any given treatment, which is given by $r(k-1)$, where $r=15$. Moreover, the sum of multiplicities of all pairwise differences balances this value–30, which must be. Hence, since the differences ±2 has a higher multiplicity–8, then, for the concurrence pattern, it follows that each pair of treatments within any triple that makes any starter

1	5	2	5	2	4	2	4	3	4	3	1	3	1	5
2	1	3	1	3	5	3	5	4	5	4	2	4	2	1
3	2	4	2	4	1	4	1	5	1	5	3	5	3	2
4	3	5	3	5	2	5	2	1	2	1	4	1	4	3
5	4	1	4	1	3	1	3	2	3	2	5	2	5	4

Figure 6.1: A $(5 \times 5)/3$ RGSLR for 5 treatments

set which has the differences ± 2 concurs in 8 blocks in the design while each pair with the differences ± 1 concurs in 7 blocks.

Moreover, $n_r = n_c = 3$ which is identical to k, the block size, where n_r and n_c have the same meaning as previously used, denoting the respective number of times each treatment appears per row and per column.

By using the procedure in section 6.2.1, we obtain the design shown in Figure 6.1

6.3 Construction of a $(3 \times 6)/4$ RGSLR where $v = 6$

We exploit the construction given in Bailey and Monod (2001), which uses balanced tournament designs (BTDs) via an exchange procedure to obtain RGSLRs for block size two when the number of columns is double the number of rows, the number of treatments is identical to the number of columns and the number of rows is not congruent to 2 modulo 3.

Notice that the number of rows in the design under construction satisfies the aforementioned congruence relationship and also have similar parameters (though differing in block size and also in the per row and per column replication numbers of its treatments, n_r and n_c, respectively) with the designs captured by the specified construction in Bailey and Monod (2001). Our construction also involves a row exchange but with a different exchange pattern as there are now more treatments per block.

Notice also that, for a design of this size, each treatment needs to appear 4 times per row ($n_r = 4$) and 2 times per column ($n_c = 2$), hence 12 times, overall. Hence $n_r = 2n_c = k$. Moreover, its QBD is a RGD if its concurrence relationship is governed by (6.2)

$$\lambda_1 + 4\lambda_2 = 36 \tag{6.2}$$

where $(\lambda_1, \lambda_2) = (8, 7)$. Hence each treatment needs to appear with precisely 1 treatment in 8 blocks and with the other 4 treatments in 7 blocks, each.

Let $V = \{1, \ldots, 5\} \cup \{\infty\}$ denote the treatment set. We use the cyclic group $\mathbb{Z}_5$ for the construction and regard $\mathbb{Z}_5$ as $\{1, \ldots 5\}$ such that $V = \mathbb{Z}_5 \cup \{\infty\}$. We give a summary of the construction procedure in the next section.

6.3.1 Procedure for the construction

1. Label the treatments $1, \ldots, 5, \infty$.

2. Create a 3×6 array and label its rows $i = 1, 2, \infty$ and the columns $j = 1, \ldots, 5, \infty$, where these (with the exception of the symbol ∞) are regarded as elements of $\mathbb{Z}_5$.

3. For $i = 1, 2$ and $j = 1, \ldots, 5$, put $S_{ij} = \{j \pm \varepsilon i\}_{\varepsilon=1}^{2}$, which is the set, $\{j+i, j-i, j+2i, j-2i\}$, where S_{ij} denotes the set of entries in the cell in position (i, j); and put $S_{\infty j}$ to consist of 2 copies of $\{j, \infty\}$, that is, $S_{\infty j}$ is a multiset of entries in the cell in position (∞, j).

4. For $j = 1, \ldots, 4$ and $i = i^{\bullet}$, exchange $\{j + i^{\bullet}, j - i^{\bullet}\} \subset S_{i^{\bullet} j}$ with one copy of $\{j, \infty\}$ in $S_{\infty j}$, where $i^{\bullet} \in \{2(2)j, -2(2)j\} \cap \{1, 2\}$ (or simply, $i^{\bullet} \in \{4j, -4j\} \cap \{1, 2\}$) is the unique element in the intersection region of the sets, thereby leaving one copy of $\{j, \infty\}$ in $S_{\infty j}$.

5. For $j = 5$ and $i = 1, 2$, exchange $\{5+i, 5-i\} \subset S_{i5}$ with one copy of $\{5, \infty\}$ in $S_{\infty 5}$, where $S_{\infty 5}$ is a multiset of entries in the cell in position $(\infty, 5)$ containing 2 copies of $\{5, \infty\}$ and after the exchanges, $S_{\infty 5}$ no longer contain the treatments $\{5, \infty\}$.

6. For $j = \infty$ and $i = 1, 2$, put $S_{i\infty} = \{\pm 3\alpha i/2\}_{\alpha=1}^{2}$, which is the set $\{3i/2, -3i/2, 3(2)i/2, -3(2)i/2\}$, or simply, $\{3i/2, -3i/2,\ 3i, -3i\}$; and put $S_{\infty\infty}$ to consist of 2 copies of $\{5, \infty\}$. Finally, for each $i \in \{1, 2\}$, exchange $\{3i/2, -3i/2\} \subset S_{i\infty}$ with one copy of $\{5, \infty\}$ in $S_{\infty\infty}$ and after the exchanges, $S_{\infty\infty}$ no longer contain the treatments $\{5, \infty\}$.

Comments. At the end of step 3; for each column $j \in \{1, \ldots, 5\}$, each treatment, $\tau \in V \backslash \{j, \infty\}$ appears once in each cell, that is, for each column, all the treatments except $\{j, \infty\}$ appear in each cell between rows 1 and 2 while the treatments j and ∞, each appears 2 times in the last cell which corresponds to the row label ∞. Hence each treatment appears 2 times in each column which corresponds to the parameter, n_c denoting the number of times each treatment should appear in each column of the design under construction. Similarly, for each row, $i \in \{1, 2\}$, each treatment except ∞ appears 4 times–once in each cell except one cell where the column label coincides with the label of that treatment. Moreover, in the last row, between columns 1 and 5, the treatment ∞ appears 10 times–twice in each cell, while every other treatment appears 2 times in a single cell and none in others.

1	∞	3	1	4	2	4	∞	5	∞	5	∞
3	4	4	5	5	1	1	2	2	3	3	2
3	4	2	∞	3	∞	1	2	5	∞	5	∞
5	2	1	3	2	4	3	5	4	1	1	4
2	5	4	5	5	1	5	3	1	4	4	1
1	∞	2	∞	3	∞	4	∞	2	3	3	2

Figure 6.2: A $(3 \times 6)/4$ RGSLR for 6 treatments

By the procedure given in section 6.3.1, the design is as shown in Figure 6.2.

Remark. Notice that the cells in cols 5 and ∞ contain identical entries, where pairs of entries of the same kind (those entries where one is negative of the other, modulo 5) likewise the entries 5 and ∞ concur a higher number of times, 8 while other pairs concur 7 times, each. Hence the design is group-divisible with groups $\{5, \infty\}$, $\{1, 4\}$ and $\{2, 3\}$.

6.4 Construction of a $(4 \times 8)/6$ RGSLR, where $v = 8$

In situations where the number of rows of a SLR is congruent to 0 or 1 modulo 4, there is a construction method in Bailey and Monod (2001) which uses starter to obtain RGSLRs whose number of columns doubles the number of rows with the block size being 2 and the number of treatments being identical to the number of columns. We exploit this construction to obtain the design under construction.

Notice that, in the design under construction, the aforementioned congruence relationship regarding the number of rows is satisfied and the parameters (apart from the block size as well as n_r and n_c that is higher) conform to that of a design of a given size that can be obtained using this method.

Notice that this design requires each treatment to appear 3 times per column ($n_c = 3$) and 6 times per row ($n_r = 6$), and 24 times, overall. Furthermore, $n_r = 2n_c = k$. Hence, to have a QBD which is an RGD, a sensible choice of the concurrence relationship is shown in (6.3).

$$\lambda_1 + 6\lambda_2 = 120 \tag{6.3}$$

where $(\lambda_1, \lambda_2) = (18, 17)$. Hence each treatment needs to appear with precisely 1 treatment in 18 blocks and with the other 6 treatments in 17 blocks, each.

Let $V = \{1, \ldots, 8\}$ denote the set of treatments. We use the cyclic group $\mathbb{Z}_8$ for the construction and regard $\mathbb{Z}_8$ as $\{1, \ldots, 8\}$. Our method involves finding first, an initial 2–subset starter in $\mathbb{Z}_8$ (where the differences, modulo 8, between the pairs in the starter sets are $\pm 1, \pm 2, \pm 3$ and 4 (twice), consisting of all the non-zero elements of $\mathbb{Z}_8$): see Bailey and Monod (2001) for 2–subset starters; see also section 5.4.1 in Chapter 5 of this thesis. The starter sets are assigned labels, S_1, S_2, S_3 and S_4, respectively which are written down as a row considering it as a cyclic route originating from S_1, forming triples with every 3 consecutive set labels and on getting to S_4 joins it up with S_1 and then S_2, where necessary to form the last triple, thus giving 4 triples, overall. The entries from the pooled sets that constitute the 4 triples are then used to form the entries of the 4 cells in the initial column of a 4×8 array and each block thus formed in the initial column is then developed cyclically via addition of 1, modulo 8,to generate the remaining blocks for each row.

We now summarize the procedure for constructing the design in section 6.4.1.

6.4.1 Procedure for the construction

1. Label the treatments $1, \ldots, 8$.

2. Form an initial 2–subset starter in $\mathbb{Z}_8$ (consisting of four 2-subsets) and label them S_i, where $i = 1, \ldots, 4$; hence $S_i = \{x_i, y_i\}$, for all i and are such that, overall, the differences, $\pm(y_i - x_i)$, modulo 8, between x_i and y_i consist of $\pm1, \pm2, \pm3$ and 4(twice), which are all the non-zero elements of $\mathbb{Z}_8$.

3. For $i = 1, \ldots, 4$, put $S_{ii'i''} = \{x_i, y_i, x_{i'}, y_{i'}, x_{i''}, y_{i''}\}$, where $S_{ii'i''}$ is the ith set of pooled entries from 3 consecutive starter sets S_i, $S_{i'}$ and $S_{i''}$. For instance, when $i = 1$, then we have S_{123} obtained by pooling the entries in S_1, S_2 and S_3; and when $i = 4$ we join up S_4 with S_1 and S_2 by pooling their entries to obtain S_{412}, where each set of pooled entries contains 6 treatments, corresponding to the required block size.

4. Create a 4×8 array and label its rows $i = 1, \ldots, 4$ and columns $j = 1, \ldots, 8$.

5. For $i = 1, \ldots, 4$, put $S_{ii'i''}$ in the cell in position $(i, 1)$ of the array and develop the initial block formed there in row i, cyclically, via addition of 1 modulo 8 to generate the other blocks in its row.

Comments. For row $i = 1, \ldots, 4$, the set $S_{ii'i''}$ which contains pooled entries from three 2–subset starter sets constitutes the initial block. The cyclic development of the initial block in each row is akin to developing each 2–subset starter set contained in $S_{ii'i''}$ and then combining the generated entries for each block. Since each of these 2-subset starter sets generates each element of $\mathbb{Z}_8$ 2 times by the cyclic development, then it follows that each treatment appears 6 times per row, as required. Similarly, each treatment appears 3 times per column since by the pooling, each S_i, for $i = 1, \ldots, 4$ is contained in 3 $S_{ii'i''}$ sets, hence its entries are contained in these 3 sets. Note that, for the sets, S_i, $i = 1, \ldots, 4$ which constitute a 2-subset starter in $\mathbb{Z}_8$, $\bigcup_{i=1}^{4} S_i$ contains each element of V exactly once.

1	2	2	3	3	4	4	5	5	6	6	7	7	8	8	1
4	6	5	7	6	8	7	1	8	2	1	3	2	4	3	5
5	8	6	1	7	2	8	3	1	4	2	5	3	6	4	7
4	6	5	7	6	8	7	1	8	2	1	3	2	4	3	5
5	8	6	1	7	2	8	3	1	4	2	5	3	6	4	7
3	7	4	8	5	1	6	2	7	3	8	4	1	5	2	6
5	8	6	1	7	2	8	3	1	4	2	5	3	6	4	7
3	7	4	8	5	1	6	2	7	3	8	4	1	5	2	6
1	2	2	3	3	4	4	5	5	6	6	7	7	8	8	1
3	7	4	8	5	1	6	2	7	3	8	4	1	5	2	6
1	2	2	3	3	4	4	5	5	6	6	7	7	8	8	1
4	6	5	7	6	8	7	1	8	2	1	3	2	4	3	5

Figure 6.3: A $(4 \times 8)/6$ RGSLR for 8 treatments

Moreover, each pair of treatments with the difference 4, modulo 8, which has a higher multiplicity appears a higher number of times, 18 in the design while pairs with other differences concur fewer times, 17. The design is group-divisible with groups being $\{1, 5\}$, $\{2, 6\}$, $\{3, 7\}$ and $\{4, 8\}$.

Using the procedure in section 6.4.1, we obtain the design shown in Figure 6.3.

Remark. Notice that the sets $\{1, 2\}$, $\{4, 6\}$, $\{5, 8\}$ and $\{3, 7\}$ denoting S_1, S_2, S_3 and S_4, respectively, whose differences, modulo 8, are $\pm 1, \pm 2, \pm 3$ and 4 (twice) constitute the initial 2–subset starter in $\mathbb{Z}_8$ used for the construction. Notice also that $S_{123} = \{1, 2, 4, 5, 6, 8\}$, $S_{234} = \{3, 4, 5, 6, 7, 8\}$, $S_{341} = \{1, 2, 3, 5, 7, 8\}$ and $S_{412} = \{1, 2, 3, 4, 6, 7\}$.

6.5 Construction of $(m \times 2m)/k$ RGSLRs, where $k = 2(m-1)$, $m > 2$ and $v = 2m$

We exploit the constructions given in Bailey and Monod (2001), which involves the use of starters and balanced tournament designs (BTDs). We note, in particular, that, designs of the sizes whose constructions are given in sections 6.3 and 6.4 belong to the aforementioned class. Hence, we generalize the constructions given there to this general class. For each method used, the relevant restriction on the row parameter, m in this case, given in Bailey and Monod (2001) applies. That is, for the method that involves BTD, m is not congruent to 2 modulo 3; and for the method that involves starter, m is congruent to 0 or 1 modulo

4. Moreover, as also given in Bailey and Monod (2001), between the two methods, the constructions accommodate values of m that are not congruent to 2 or 11 modulo 12.

Notice that each treatment of the design is required to appear $n_r = 2(m-1)$ times per row, $n_c = m-1$ times per column, and $2m(m-1)$ times, overall.. The sum of concurrences is $2m(m-1)(2m-3)$ and the design will be a RGD if the concurrence relationship for pairs of treatments is governed by

$$\lambda_1 + 2(m-1)\lambda_2 = 2m(m-1)(2m-3) \tag{6.4}$$

which gives $\lambda_1 = 2(m-1)^2$ and $\lambda_2 = 2m^2 - 4m + 1$. Hence each treatment is required to concur with precisely 1 treatment in $2(m-1)^2$ blocks and with the rest $(2(m-1)$ of them) in $2m^2 - 4m + 1$ blocks, each.

6.5.1 Construction via BTDs

Let $V = \{1, \ldots, 2m-1\} \cup \{\infty\}$ denote the treatment set, where $(m \geq 3)$ is not congruent to 2 modulo 3. Let $w = 2m - 1$ be prime. We use the cyclic group $\mathbb{Z}_w$, the integers modulo w, and regard $\mathbb{Z}_w$ as $\{1, \ldots w\}$ such that $V = \mathbb{Z}_w \cup \{\infty\}$. We give an algorithmic procedure for constructing the design. The algorithm generalizes the procedure given in section 6.3.1.

An algorithmic procedure for the construction

1. Label the treatments $1, \ldots, w, \infty$, where $w = 2m - 1$ is prime.

2. Create an $m \times 2m$ array and label its rows $i = 1, \ldots, m-1, \infty$ and the columns $j = 1, \ldots, w, \infty$, where these (with the exception of the symbol ∞) are regarded as elements of $\mathbb{Z}_w$.

3. For $i = 1, \ldots, m-1$ and $j = 1, \ldots, w$, put $S_{ij} = \{j \pm \varepsilon i\}_{\varepsilon=1}^{m-1}$, which is the set, $\{j+i, j-i, j+2i, j-2i, \ldots, j+(m-1)i, j-(m-1)i\}$, where S_{ij} denote the set of entries in the cell in position (i, j); and put $S_{\infty j}$ to consist of $m - 1 = k/2$ copies of $\{j, \infty\}$, that is, $S_{\infty j}$ is a multiset of entries in the cell in position (∞, j).

4. For $j = 1, \ldots, w-1$ and $i = i_q^\bullet$, where $q = 1, \ldots, m-2$, exchange $\{j+i_q^\bullet, j-i_q^\bullet\} \subset S_{i_q^\bullet j}$ with one copy of $\{j, \infty\}$ in $S_{\infty j}$, for all q, where $i_q^\bullet \in \{2(q+1)j, -2(q+1)j\} \cap \{1, \ldots, m-1\}$ is the unique element in the intersection region of the sets for a fixed q; thereby leaving precisely one copy of $\{j, \infty\}$ left in $S_{\infty j}$, since $m - 2$ copies are exchanged by this procedure.

5. For $j = w$ and $i = 1, \ldots, m-1$, exchange $\{w+i, w-i\} \subset S_{iw}$ with one copy of $\{w, \infty\}$ in $S_{\infty w}$, where $S_{\infty w}$ is a multiset of entries in the cell in position (∞, w) containing $m-1$ copies of $\{w, \infty\}$ and after all the exchanges, $S_{\infty w}$ no longer contain the treatments $\{w, \infty\}$.

6. For $j = \infty$ and $i = 1, \ldots, m-1$, put $S_{i\infty} = \{\pm 3\alpha i/2\}_{\alpha=1}^{m-1}$, which is the set $\{3i/2, -3i/2,\ 3(2)i/2, -3(2)i/2, \ldots, 3(m-1)i/2, -3(m-1)i/2\}$; and put $S_{\infty\infty}$ to consist of $m-1$ copies of $\{w, \infty\}$. Finally, for each $i \in \{1, \ldots, m-1\}$, exchange $\{3i/2, -3i/2\} \subset S_{i\infty}$ with one copy of $\{w, \infty\}$ in $S_{\infty\infty}$ and after all the exchanges, $S_{\infty\infty}$ no longer contain the treatments $\{w, \infty\}$.

Comments. Notice in step 4 that if $q = 1$, then we have $i_1^{\bullet} \in \{2(2)j, -2(2)j\} \cap \{1, \ldots, m-1\}$, which is identical to $i_1^{\bullet} \in \{4j, -4j\} \cap \{1, \ldots, m-1\}$. Similarly, if $q = m-2$, then $i_{m-2}^{\bullet} \in \{2(m-1)j, -2(m-1)j\} \cap \{1, \ldots, m-1\}$.

At the end of step 3, for each column $j \in \{1, \ldots, w\}$, each treatment, $\tau \in V \backslash \{j, \infty\}$ appears once in each cell between rows 1 and $m-1$ while the treatments j and ∞, each appears $m-1$ times in the last cell which corresponds to the row label ∞. Hence each treatment appears $m-1$ times in each column and this corresponds to the parameter, n_c denoting the number of times each treatment should appear in each column of the design under construction. Similarly, in each row $i \in \{1, \ldots, m-1\}$, between columns 1 and w, each treatment, $\beta \in V \backslash \{\infty\}$ appears once in every other cell except the cell in position (i, β), that is, the cell whose column label corresponds to β, for a fixed $i \in \{1, \ldots, m-1\}$. This means that, for each $i \in \{1, \ldots, m-1\}$, $\beta \in S_{ij}$, for each $j \neq \beta$. Hence it appears $w-1 = 2m-2$ times, which is identical to $2(m-1)$, the parameter n_r denoting the number of times each treatment should appear per row. However, in row ∞, each treatment (other than the treatment ∞) appears $m-1$ times in a single cell whose column label corresponds to the label of that treatment while the treatment with the label ∞ appears $(m-1)(2m-1)$ times, that is, $(m-1)$ times in each cell.

Moreover, at the end of step 6, in the full design, each cell in column w also appears as a cell in column ∞ (not necessarily in the same row). In column w, each pair of treatments of the same kind, $\{w+i, w-i\}$, which is identical to $\{i, -i\}$, that is $\{\pm i\}$ or equivalently, $\{i, w-i\}$ for all $i = 1, \ldots, m-1$ appears higher in the design. Similarly, the pair $\{w, \infty\}$ also appears higher. Each of such pairs appears $m-1$ times in each of columns w and ∞ (2 columns) and $m-2$ times in the rest of the columns, $2(m-1)$ of them. Hence each of such pairs concur in

$$\lambda_1 = 2(m-1) + 2(m-1)(m-2) = 2(m-1)^2$$

as required. Similarly, every other pair of treatments not of the same kind as above concur $m-1$ times in a single column and concurs $m-2$ times in other columns. Hence each of these pairs concur in

$$\lambda_2 = 1(m-1) + (m-2)(2m-1) = 2m^2 - 4m + 1.$$

as also required.

The design is thus group-divisible with groups $\{w, \infty\}$, $\{1, w-1\}$, $\{2, w-2\}, \ldots, \{m-1, m\}$.

6.5.2 Construction via starter

We generalize the ideas in section 6.4 to give construction for $(m \times 2m)/k$ RGSLRs, where $k = 2(m-1)$, $v = 2m$ and m (≥ 4) is congruent to 0 or 1 modulo 4. We use the cyclic group $\mathbb{Z}_{2m}$, the integers, modulo $2m$, and regard $\mathbb{Z}_{2m}$ as $\{1, \ldots, 2m\}$. We give an algorithmic procedure for the construction in the next section. The algorithm generalizes the procedure given in section 6.4.1.

6.5.3 Procedure for the construction

1. Label the treatments $1, \ldots, 2m$.

2. Form an initial 2–subset starter in $\mathbb{Z}_{2m}$ (consisting of m 2-subsets) and label them S_i, where $i = 1, \ldots, m$; hence $S_i = \{x_i, y_i\}$, for all i and are such that, overall, the differences, $\pm(y_i - x_i)$, modulo $2m$, between x_i and y_i consist of $\pm 1, \pm 2, \ldots, m$(twice), which are all the non-zero elements of $\mathbb{Z}_{2m}$.

3. For $i = 1, \ldots, m$, put $S_{ii+1\ldots i+m-2} = \{x_i, y_i, x_{i+1}, y_{i+1}, \ldots, x_{i+m-2}, y_{i+m-2}\}$, where $S_{ii+1\ldots i+m-2}$ is the ith set of pooled entries from $m-1$ consecutive starter sets $S_i, S_{i+1}, \ldots, S_{i+m-2}$. For instance, when $i = 1$, we have $S_{12\ldots m-1}$ obtained by pooling the entries in $S_1, \ldots, S_{m-1}$; and so on, and when you get to the end, join up S_m with S_1 until the number of sets whose entries are pooled is $m-1$. In particular, when $i = m$, we have $S_{m1\ldots m-2}$ obtained by pooling the entries in $S_m, S_1, \ldots, S_{m-2}$. Note that each set of pooled entries contains $2(m-1)$ treatments, which is the required block size.

4. Create an $m \times 2m$ array and label its rows $i = 1, \ldots, m$ and columns $j = 1, \ldots, 2m$.

5. For $i = 1, \ldots, m$, put $S_{ii+1\ldots i+m-2}$ in the cell in position $(i, 1)$ of the array and develop the initial block formed there in row i, cyclically, via addition of 1 modulo 2m to generate the other blocks in its row.

Comments. For $i = 1, \ldots, m$, the set $S_{ii+1\ldots i+m-2}$ which contains pooled entries from $m-1$ 2–subset starter sets constitutes the initial block. The cyclic development of the initial block in each row is akin to developing each 2–subset starter set contained in $S_{ii+1\ldots i+m-2}$ and then combining the generated entries for each block. Since each of these 2-subset starter sets generates each element of $\mathbb{Z}_{2m}$ 2 times by the cyclic development, then it follows that each treatment appears $2(m-1)$ times per row, as required. Similarly, each treatment appears $m-1$ times per column since by the pooling, each S_i, for $i = 1, \ldots, m$ is contained in $m-1$ $S_{ii+1\ldots i+m-2}$ sets, hence its entries are contained in these $m-1$ sets. Also S_i, where $i = 1, \ldots, m$ constitute a 2-subset starter in $\mathbb{Z}_{2m}$, where $\bigcup_{i=1}^{m} S_i$ contains each element of V exactly once.

Moreover, each pair of treatments with the difference m, modulo $2m$, which has a higher multiplicity concurs a higher number of times, $\lambda_1 = 2(m-1)^2$ in the design while pairs with the other differences, each concurs fewer times, $\lambda_2 = 2m^2 - 4m + 1$. Notice that λ_1 and λ_2 are quadratic factors in m. Note that λ_1 can be viewed this way: each pair with the difference m appears in $m-1$ blocks in 2 columns, each (columns c and $c+m$, where c is the first column it appears) and appears in $m-2$ blocks in each of the remaining $2m-2 = 2(m-1)$ columns. Hence

$$\lambda_1 = 2(m-1) + 2(m-1)(m-2) = 2(m-1)^2$$

Similarly, λ_2 can be seen by noting that each pair with any difference other than 4 appears $m-1$ times in exactly 1 column and $m-2$ times in each of the remaining $2m-1$ columns.Thus

$$\lambda_2 = 1(m-1) + (2m-1)(m-2) = 2m^2 - 4m + 1$$

The design is thus group-divisible with groups being $\{1, m+1\}, \{2, m+2\}, \ldots, \{m-1, 2m-1\}, \{m, 2m\}$.

Notice that λ_1 and λ_2 satisfy (6.4).

6.5.4 Construction of RGSLRs via complementation

Just like the complement of a BIBD is another BIBD, the complement of a RGD is also another RGD: see, for example, John and Williams (1982). Hence a RGSLR can be obtained from another RGSLR by complementing the within-block treatments, which we name block (cell) complementation, that is, replacing each block of the 'parent' RGSLR by those treatments that are missing there. In particular, given a RGSLR on a treatment set, V (where the cardinality of V is v) and having block size k, we employ the concept of block complementation to obtain another RGSLR on the same set of treatments but with block size $k' = v - k$ by putting in each block of the 'parent' design, the set, $V \backslash S_{ij}$ of treatments, where S_{ij} is the set of treatments in the cell in position (i, j) of the 'parent design', for all i and j. This procedure provides an alternative construction for obtaining a RGSLR with block size $v - k$ when there exists another RGSLR with block size k. It can be used to obtain RGSLRs of the class specified in section 6.5 whose direct methods of construction are given in sections 6.5.1 and 6.5.2.

Theorem 6.5.1. *Let h, p, k, v be positive integers.. For a fixed h, p, k, let Δ_1 denote an $(h \times p)/k$ RGSLR on the treatment set V, where $V = \{1, 2, ..., v\}$. Let S_{ij} denote the set of entries in the cell in position (i, j) of Δ_1, where $i = 1, \ldots, h$ and $j = 1, \ldots, p$. Let Δ_2 denote a design obtained from Δ_1 by putting S_{ij} to be $S'_{ij} = V \backslash S_{ij}$, for all i and j, where S'_{ij} is the set of entries in the corresponding cell, (i, j) in Δ_2. Then Δ_2 is an $(h \times p)/k'$ RGSLR on the same treatment set as Δ_1, where $k' = v - k$.*

Proof. We first investigate whether Δ_2 is a SLR.

Since Δ_1 is a SLR, then each treatment appears n_r times in each row and n_c times in each column, where $n_r, n_c \in \mathbb{Z}_+$. It follows that, if Δ_2 is a SLR, then each treatment needs to appear n_r^* times, say, in each row and n_c^* times, say, in each column, where $n_r^*, n_c^* \in \mathbb{Z}_+$.

Letl $\tau \in V$. Then for all $\tau \in V$, since Δ_1 is a SLR whose QBD is binary, it implies that, for all $i = 1, \ldots, h$, τ appears in n_r cells. Similarly, for all $j = 1, \ldots, p$, τ appears in n_c cells. Now, in Δ_2, for all $i = 1, \ldots, h$, every $\tau \in V$ appears in $p - n_r$ cells, which are, precisely, those cells it does not appear in Δ_1. Similar to this, in Δ_2, for all $j = 1, \ldots, p$, each $\tau \in V$ appears in $h - n_c$ cells, which are, precisely, those cells it does not appear in Δ_1. Notice that since Δ_1 is a SLR, then for all $i = 1, \ldots, h$ and $j = 1, \ldots, p$, $|S_{ij}| = k$, where $k > 0$, thus $S_{ij} \neq \emptyset$ for all i and j. Since $S_{ij} \subset V$, it follows that, $S'_{ij} = V \backslash S_{ij} \subset V$, for all i and j hence $S'_{ij} \neq \emptyset$ such that $k' = |S'_{ij}| > 0$, where $|S'_{ij}| = |V \backslash S_{ij}| = v - k$. Notice that $p - n_r > 0$, as $n_r < p$ since its QBD is binary and $\tau \in V$ does not appear in all the cells–the blocks being incomplete. Similarly, $h - n_c > 0$, as $n_c < h$. It follows from these discussion that Δ_2 is a SLR on the set, V of treatments.

Now, since Δ_1 is a RGSLR, then its QBD is a RGD, thus has two distinct treatment concurrence counts for all pairs of treatments. Denote by Λ_1, the QBD of Δ_1. Furthermore, denote by λ_1 and λ_2, the two classes of concurrences. Then $|\lambda_2 - \lambda_1| = 1$ or equivalently, $\lambda_2 = \lambda_1 \pm 1$. Notice that each $\tau \in V$ appears in hn_r blocks, overall, hence since there are hp blocks in Λ_1, it implies that τ does not appear in $hp - hn_r = h(p - n_r)$ blocks. Let $\tau_1, \tau_2, \tau_3 \in V$. Let τ_1 appear with τ_2 in λ_1 blocks and with τ_3 in λ_2 blocks. Moreover, τ_1, for instance, appears without τ_2 in $hn_r - \lambda_1$ blocks. Similarly, τ_1 appears without τ_3 in $hn_r - \lambda_2$ blocks.

We now investigate whether the QBD of Δ_2 is a RGD. Let Λ_2 denote the QBD of Δ_2. We note that, for any cell in position (i, j), if $\tau_1, \tau_2 \in S_{ij}$, then $\tau_1, \tau_2 \notin S'_{ij}$. Conversely, if $\tau_1, \tau_2 \in S'_{ij}$, then $\tau_1, \tau_2 \notin S_{ij}$. Hence $\tau_1, \tau_2 \in S'_{ij}$ if and only if $\tau_1, \tau_2 \notin S_{ij}$. Similarly, $\tau_1, \tau_3 \in S'_{ij}$ if and only if $\tau_1, \tau_3 \notin S_{ij}$. Let λ'_1 and λ'_2 denote the treatment concurrence counts for the pairs $\{\tau_1, \tau_2\}$ and $\{\tau_1, \tau_3\}$, respectively, in Λ_2, that is, the respective number of blocks that these pairs of treatments appear together in Λ_2. Let $\mathcal{B} = \{S_{ij}\}_{(i,j)=(1,1)}^{(h,p)}$, the set of all blocks in the design which has cardinality, $|\mathcal{B}| = hp$. We have that

$$\lambda'_1 = |\mathcal{B}| - (|A_1| + |A_2| - \lambda_1) \tag{6.5}$$

and

$$\lambda'_2 = |\mathcal{B}| - (|A_1| + |A_3| - \lambda_2) \tag{6.6}$$

where $A_l = \{S_{ij} : \tau_l \in S_{ij}\}$, for all $l = 1, 2, 3$. Note that $|A_l| = hn_r$, for all $l = 1, 2, 3$.

Hence from (6.5), $\lambda'_1 = hp - 2hn_r + \lambda_1$. Similarly, from (6.6), $\lambda'_2 = hp - 2hn_r + \lambda_2$. Notice that $|\lambda'_2 - \lambda'_1| = |\lambda_2 - \lambda_1| = 1$. This result holds for any set of treatments in V which have different concurrence counts in Λ_1, hence in Λ_2. Thus the QBD of Δ_2 is a RGD. It follows that Δ_2 is an $(h \times p)/k'$ RGSLR on the same treatment set as Δ_1, where $k' = v - k$.

4 6	5 7	6 8	7 1	8 2	1 3	2 4	3 5
5 8	6 1	7 2	8 3	1 4	2 5	3 6	4 7
3 7	4 8	5 1	6 2	7 3	8 4	1 5	2 6
5 8	6 1	7 2	8 3	1 4	2 5	3 6	4 7
3 7	4 8	5 1	6 2	7 3	8 4	1 5	2 6
1 2	2 3	3 4	4 5	5 6	6 7	7 8	8 1
3 7	4 8	5 1	6 2	7 3	8 4	1 5	2 6
1 2	2 3	3 4	4 5	5 6	6 7	7 8	8 1
4 6	5 7	6 8	7 1	8 2	1 3	2 4	3 5
1 2	2 3	3 4	4 5	5 6	6 7	7 8	8 1
4 6	5 7	6 8	7 1	8 2	1 3	2 4	3 5
5 8	6 1	7 2	8 3	1 4	2 5	3 6	4 7

Figure 6.4: A $(4 \times 8)/6$ RGSLR for 8 treatments obtained via block complementation

■

Example 6.5.1. As an illustration, by applying block complementation to the $(4 \times 8)/2$ RGSLR for 8 treatments shown in Figure 5.6 in Chapter 5, we obtain the $(4\times 8)/6$ RGSLR for 8 treatments shown in Figure 6.4.

Comments. Notice that the designs shown in Figures 6.3 and 6.4 obtained by a direct method via starter and block complementation, respectively, are isomorphic. This can be seen by applying the permutation, α_1 to the rows of the design shown in Figure 6.4, where

$$\alpha_1 = \begin{pmatrix} 1 & 2 & 3 & 4 \\ \\ 2 & 3 & 4 & 1 \end{pmatrix}$$

Similarly, if the permutation, α_2 (which reverses α_1) given below is applied to the rows of the design shown in Figure 6.3, then it leads to the other design.

$$\alpha_2 = \begin{pmatrix} 1 & 2 & 3 & 4 \\ \\ 4 & 1 & 2 & 3 \end{pmatrix}$$

Moreover, in general, a rearrangement of the pooled starter sets among the m cells of the initial column leads to a different design each time. In particular, if step 5 of the

1 3 4 ∞	1 3 4 5	1 2 4 5	1 2 4 ∞	2 3 5 ∞	2 3 5 ∞
2 3 4 5	1 2 3 ∞	2 3 4 ∞	1 2 3 5	1 4 5 ∞	1 4 5 ∞
1 2 5 ∞	2 4 5 ∞	1 3 5 ∞	3 4 5 ∞	1 2 3 4	1 2 3 4

Figure 6.5: A $(3 \times 6)/4$ RGSLR for 6 treatments obtained via block complementation

procedure in section 6.4.1 is modified slightly by putting S_{234}, S_{341}, S_{412} and S_{123} in the cells in positions $(1,1)$, $(2,1)$, $(3,1)$ and $(4,1)$, respectively, then we get exactly the design in Figure 6.4.

Example 6.5.2. From the first part of the design in Figure 5.20 in Chapter 5, that is, the first 6 columns which constitute a $(3 \times 6)/2$ RGSLR for 6 treatments; by block complementation, we obtain the design shown in Figure 6.5.

Notice that the design shown in Figure 6.5, obtained via block complementation, is identical to the design shown in Figure 6.2, obtained by a direct approach.

6.6 RGSLRs of larger sizes

As mentioned in section 6.1, a RGSLR of larger size can be obtained by adjoining a BSLR to a RGSLR, where both constituent designs exist and are of conformable sizes. Furthermore, by applying a suitable permutation of treatments to a RGSLR and then adjoining it to the 'parent' RGSLR also gives another RGSLR of larger size, if both designs also conform in size. We adopt this procedure to obtain designs of sizes larger than that of a given 'parent' RGSLR.

Example 6.6.1. Let $h = 5$, $p = 15$, $k = 3$ and $v = 5$. By putting a $(5 \times 5)/3$ RGSLR and a $(5 \times 10)/3$ BSLR shown in Figures 6.1 and 4.1, respectively, side by side, we obtain the $(5 \times 15)/3$ RGSLR shown in Figure 6.6.

Example 6.6.2. Let $h = 3$, $p = 12$, $k = 4$ and $v = 6$. By applying the permutation α to every treatment of the design in Figure 6.2 (the 'parent' design), where

$$\alpha = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & \infty \\ 3 & 4 & 5 & 1 & \infty & 2, \end{pmatrix}$$

and then putting the resulting design with the 'parent' design, side by side, we obtain the design shown in Figure 6.7.

1 5 2	5 2 4	2 4 3	4 3 1	3 1 5	1 2 5	2 3 1	3 4 2	4 5 3	5 1 4	1 3 4	2 4 5	3 5 1	4 1 2	5 2 3
2 1 3	1 3 5	3 5 4	5 4 2	4 2 1	5 1 4	1 2 5	2 3 1	3 4 2	4 5 3	5 2 3	1 3 4	2 4 5	3 5 1	4 1 2
3 2 4	2 4 1	4 1 5	1 5 3	5 3 2	4 5 3	5 1 4	1 2 5	2 3 1	3 4 2	4 1 2	5 2 3	1 3 4	2 4 5	3 5 1
4 3 5	3 5 2	5 2 1	2 1 4	1 4 3	3 4 2	4 5 3	5 1 4	1 2 5	2 3 1	3 5 1	4 1 2	5 2 3	1 3 4	2 4 5
5 4 1	4 1 3	1 3 2	3 2 5	2 5 4	2 3 1	3 4 2	4 5 3	5 1 4	1 2 5	2 4 5	3 5 1	4 1 2	5 2 3	1 3 4

Figure 6.6: A $(5 \times 15)/3$ RGSLR for 5 treatments

1 ∞	3 1	4 2	4 ∞	5 ∞	5 ∞	3 2	5 3	1 4	1 2	∞ 2	∞ 2
3 4	4 5	5 1	1 2	2 3	3 2	5 1	1 ∞	∞ 3	3 4	4 5	5 4
3 4	2 ∞	3 ∞	1 2	5 ∞	5 ∞	5 1	4 2	5 2	3 4	∞ 2	∞ 2
5 2	1 3	2 4	3 5	4 1	1 4	∞ 4	3 5	4 1	5 ∞	1 3	3 1
2 5	4 5	5 1	5 3	1 4	4 1	4 ∞	1 ∞	∞ 3	∞ 5	3 1	1 3
1 ∞	2 ∞	3 ∞	4 ∞	2 3	3 2	3 2	4 2	5 2	1 2	4 5	5 4

Figure 6.7: A $(3 \times 12)/4$ RGSLR for 6 treatments

Chapter 7

Conclusion

7.1 Introduction

This chapter showcases, in summary form, the main results of the work that has been done in this thesis, and also makes some relevant conclusion about the work. We consider semi-Latin rectangles (SLRs) whose quotient block designs (QBDs) are balanced incomplete-block designs (BIBDs), that is, the balanced semi-Latin rectangles (BSLRs) and those that their QBDs are not balanced–the non-balanced semi-Latin rectangles (NBSLRs), separately. For each case, we consider designs with block size 2 and those with block sizes larger than 2. Moreover, a table showing some sets of parameters that can give designs for each of BSLRs and RGSLRs is given. Some suggestions for further work are also given.

7.2 Balanced semi-Latin rectangles

We have developed some constructions for BSLRs of various classes and sizes, ranging from the case where the block size, k is 2 to the case where $k > 2$. For $k = 2$, we have employed, basically, two concepts, viz, graph distance and parallel classes to obtain basic designs for those experimental situations where the number of treatments, v is odd and even, respectively. An algorithm is given for each construction. These have been published: see Uto and Bailey (2020). When v is odd, the algorithm produces a BSLR with $h = v$ rows and $p = v\delta$ columns, where $\delta = (v-1)/2$ which is precisely a $(v \times v\delta)/2$ BSLR. Similarly, when v is even, the corresponding algorithm produces designs with $h = v/2$ rows and $p = v(v-1)/2$ columns. The two constructions via these algorithms serve as basic constructions for obtaining larger designs.

For $k > 2$; if v is odd and $k = 3$, we have given a construction procedure which involves a modification of the distance approach that was used to obtain designs for $k = 2$. We have also utilized the concepts of difference sets/difference families for $\mathbb{Z}_v$, the set of integers, modulo v (when they exist) to obtain designs for those values of k that these exist. In particular, if a difference set exists, then the design obtained from the construction has same number of rows as columns. We have always regarded $\mathbb{Z}_v$ as $\{1, \ldots, v\}$. Furthermore, if k is a prime power and v is the square of k, then the concept of affine resolvability via a complete set of MOLSs and a square array of order k can also be employed to obtain designs of appropriate sizes. The algorithm given for each of these procedures produces the basic design from which larger designs can also be obtained.

In both cases, that is, whether $k = 2$ or $k > 2$, if certain conditions are satisfied, that is, if in situations where multiple Latin squares are used for the basic construction and the number of Latin squares involved is a nonprime, then certain rearrangements of the basic design produce designs of different other sizes. In particular, if the number of Latin squares used in the construction is a perfect square, then it is possible to obtain, as one of the arrangements, a BSLR with same number of rows as columns. Moreover, designs of larger sizes can be obtained from the basic design by making multiple copies of it and then making appropriate juxtapositions of the copies, which may involve juxtaposing all the copies side by side, under or a combination of these.

A BSLR does not always exist. It exists only if

$$vhn_r = vpn_c = khp \tag{7.1}$$

and

$$\lambda(v-1) = hn_r(k-1) = pn_c(k-1), \tag{7.2}$$

that is, (7.1) and (7.2) give necessary conditions for a BSLR to exist. Moreover, when a BSLR exists, it is optimal over every other SLR of its size, thus giving the best design for experiment: see Uto and Bailey (2020). This is so since its QBD is a BIBD, which is known to be optimal (under a range of criteria) over all incomplete-block designs of its size.

We present, in Table 7.1, some sets of parameters that can make a BSLR alongside their constructions, if that has been covered by one of our methods. However, for any set of parameters in the table which our construction methods do not cover, we leave it blank. The Solution column contains some information regarding where the particular design or its construction can be found in the thesis, quoting the relevant Figure label, if the design has been written out or the relevant section, if the design is covered by one of our methods but is not written out. Note that, wherever JSTP appears in the Solution column in the table, it means that the design with the corresponding set of parameters also appears in Uto and Bailey (2020), which is a publication in JSTP–an acronym for Journal of Statistical Theory and Practice.

The table consists of parameter sets for designs with $h \leq p$, h and p being the number of rows and columns, respectively. We note that if a design with $h < p$ exists, then there also exist a corresponding design with the values of h and p swapped (also swapping the values of n_r and n_c) such that the new design has $h' = p$ rows and $p' = h$ columns, where $p > h$. Furthermore, the various parameters have their usual meanings, for instance, $r = hn_r = pn_c$ denotes the replication number of each treatment in the design while $b = hp$

denotes the number of blocks. n^* specifies the number of different possible arrangements into a BSLR (based on our methods) with the same value of b and other parameters while toggling with the values of h and p (also adjusting the values of n_r and n_c) in situations that $h \leq p$, for each set of parameters in the table. However, for a given set of parameters, if $h = p$, and the set of parameters gives a basic design, then the value of $n^* = 1$, as can be seen from the table. Furthermore, if $h < p$, then for each set of parameters that satisfy this, the value of h can be swapped with p (likewise n_r and n_c) to give another set of parameters that can make an equivalent design–a design of the same size as would be obtained via transposition of the former, which also generates the same number of designs (possibilities) as with the former set of parameters, where in this case, there are more rows than columns.

Now, whether $h < p$ (as in the table) or $h' > p'$ (after swapping), there may be a possibility of obtaining a design which contains identical number of rows as columns. Let $h < p$. Suppose $p = h\epsilon$, and the basic construction involves ϵ constituent Latin squares, then if ϵ is a perfect square, it may be possible to have one of the arrangements to have identical number of rows as columns. But if ϵ is a nonprime that is not a perfect square, there are also other possibilities: see Uto and Bailey (2020). In all these cases, r remains constant since the various QBDs are the same. Hence, overall, with the same QBD, if $h \neq p$ in the original set of parameters in the table, a total of $2n^*$ designs can be generated if ϵ is a nonprime that is not a perfect square (n^* with $h < p$ and also another n^* with $h' > p'$, the swapped parameters). This is so since by the construction methods, for every $(h \times p)/k$ BSLR there is a corresponding $(h' \times p')/k$ BSLR, where $h' = p$ and $p' = h$ which can be obtained by changing the order of juxtaposition of the constituent designs. However, if ϵ is a perfect square, then, overall, the number of different arrangements becomes $2(n^* - 1) + 1 = 2n^* - 1$.

For instance, by our construction methods, a design such as the $(3 \times 60)/2$ BSLR in S/N 14 of Table 7.1 can have its blocks arranged into a $(6 \times 30)/2$ BSLR and also a $(12 \times 15)/2$ BSLR. By the construction methods, a $(60 \times 3)/2$ BSLR, $(30 \times 6)/2$ BSLR and $(15 \times 12)/2$ BSLR can be obtained by changing the other of juxtaposition each time, thus giving $2n^* = 6$ designs, overall.

7.3 Non-balanced semi-Latin rectangles

When no BSLR exists, good SLRs can be found among RGSLRs (if they exist), particularly, if the number of blocks, hp is reasonably large. This is so since the QBD of a RGSLR is a RGD, and RGDs, when they exist, are known to contain the D-optimal (or A-optimal or E-optimal) design, provided the number of blocks is reasonably large: see Cheng (1992). Cakiroglu (2018), under the A–optimality and D–optimality, asserts that, RGDs, under the condition of having large number of blocks, contain the A-optimal (or D-optimal) design, if any exists. Moreover, extending RGDs with copies of BIBDs produce designs with good statistical properties: see Cakiroglu (2018). If the QBD of a RGSLR consists of a RGD and BIBD, we call it BIBD-extended RGSLR; however, in the case where a BSLR is adjoined to RGSLR, we call it a BSLR-extended RGSLR.

For even v and $k = 2$, we have extended the constructions in Bailey and Monod (2001) to obtain BIBD-extended RGSLRs for some values of $h = m$ rows and $p = m(\theta + 2)$ columns, where $v = 2m$ and $\theta = 1, 2, 4$. In particular, for $\theta = 1$, the construction involves a Trojan square and for the other values of θ it involves some suitable permutations of the treatments of the 'parent' design–the design obtained from the constructions in Bailey and Monod (2001). For the permutation, if $\theta = 2$, then exactly 1 permutation is required and if $\theta = 4$, then 2 permutations are required, and the permutations are applied successively, one after the other, to the treatments in the 'parent' design. Moreover, for odd v, we have obtained constructions for designs with $h = p = m$ and $v = m$ using the concept of starter in $\mathbb{Z}_m$. This is further generalized to designs with more columns using the concept of permutation. Starter sets in $\mathbb{Z}_m$ for small odd values of m up to 15 are given. Undirected terrace provides a more convenient approach to generating the starter sets. Also, for the case that v is odd, under certain conditions, as shown in section 5.8.4, a BSLR can be realized from our construction of RGSLRs.

Moreover, for $k > 2$, our constructions are based on certain concepts like undirected terrace. We have also exploited the constructions in Bailey and Monod (2001) to obtain more constructions and this has been generalized to give a direct construction for designs of sizes that can also be obtained via block complementation. In one of these direct methods, there is some requirement that the order of the group be prime; and designs obtained using this method is found to be identical to that obtained via block complementation. The other direct construction produces designs that are isomorphic to those obtained via complementation. However, with a slight modification of the procedure, it produces same design as complementation would do.

In both cases, that is, for $k \geq 2$, larger designs can be obtained by adjoining a BSLR of conformable size to a RGSLR or adjoining a RGSLR to another after a suitable permutation of treatments, that is, for a given 'parent' RGSLR, a suitable permutation is applied to its treatments to obtain another RGSLR which is then adjoined to the 'parent' design.

Table 7.2 shows sets of parameters that can give a RGSLR, where x denotes the number of treatments that can concur with another a higher number, λ^* of times while y denotes the number of treatments that concur with it less number, λ' of times, that is we have assumed each treatment of the design to concur with x treatments a higher number of times, λ^* and with y treatments less number of times, λ'. Just like in Table 7.1, Table 7.2 contains parameter values for which $h \leq p$. Furthermore, if an $(h \times p)/k$ RGSLR for v treatments exists, then there also exists a corresponding $(p \times h)/k$ RGSLR. Hence, for each combination of parameters in the table that make a design, another set of parameters not listed in the table can be obtained by exchanging the values of h and p, if h and p are different values.

7.4 Some important general remarks concerning the designs

Algorithms are given for the constructions and we note, in general, that among the designs obtained, there are some whose number of rows are identical to the number of columns. Being a SLR, the parameters, n_r and n_c, denoting the number of times each treatment appears in each row and in each column, respectively, are not all equal to 1. Hence a SLR can have identical number of rows as columns without being a semi-Latin square (SLS). Note in particular that it can only be a SLS if these two parameters are each equal to 1. Moreover, if in addition to n_r and n_c being 1, the block size, k is also 1, then the design is trivially a Latin square (LS).

For a fixed set of parameters, h, p and k, a $(p \times h)/k$ SLR can be obtained from an $(h \times p)/k$ SLR by transposing the $(h \times p)/k$ SLR: see Uto and Bailey (2020).

Moreover, we have established that, for both BSLRs and RGSLRs, block complementation can be a useful concept for obtaining a new SLR (within either of these two classes) from another, hence providing a convenient means of obtaining SLRs of these classes, particularly, for much higher values of k. For BSLRs, in the case where the design has precisely 2 rows or 2 columns and $k = v/2$, we have also considered other forms of complementation, which we name column complementation and row complementation, respectively. Hence, given any BSLR or RGSLR, a new BSLR or RGSLR, as the case may be, can be obtained by employing the concept of block complementation, while in particular cases of BSLRs, where there are 2 rows or 2 columns, column and row complementations, respectively, can also be useful.

Among some classes of designs obtained, we have found some designs to be isomorphic. Furthermore, complementation works in every case in obtaining a design of the class given in section 6.5. However, the direct construction method given there that uses an exchange algorithm works only for prime values of w and it also produces identical designs as would be obtained via complementation. The method which involves a cyclic development of initial blocks formed by pooled starter sets produce designs which are isomorphic to those that would be produced via complementation, but with a slight modification of the procedure which involves rearrangement/repositioning of the starter sets among the cells in the initial column, it can produce designs that are identical to that obtained via complementation.

7.5 Suggestions for further work

Efforts can be geared towards obtaining some more general constructions for good SLRs to fill in the gap for designs of those sizes not covered by this work, particularly, for situations where $k > 2$ and a BSLR fails to exist. More investigations can be made into the class of RGSLRs.

Moreover, it is also worth investigating the isomorphism classes of SLRs.

In experimental situations where the outcome of one observation might be affected by other treatments in the same cell, there is need to consider obtaining designs that suit this situation. Also, if the rows correspond to time, then there might be a carry-over effects from the treatments in the same column but in the previous time. It might be of interest to consider how to obtain designs for this purpose.

Table 7.1: Table of parameters for some BSLRs

S/N	v	k	h	p	n_c	n_r	λ	r	Construction	b	n^*	Solution
1	3	2	3	3	2	2	3	6	**distance**	9	1	Figure 3.1
2	3	2	3	6	2	4	6	12	Two copies of 1+ juxtaposition beside	18	1	Figure 3.26
3	4	2	2	6	1	3	2	6	**parallel class**	12	1	Figure 3.5
4	4	2	2	12	1	6	4	12	two copies of 3 + juxtaposition beside	24	2	Figure 3.31
5	4	2	4	6	2	3	4	12	two copies of 3 + juxtaposition underneath	24	2	Figure 3.32
6	4	2	2	18	1	9	6	18	three copies of 3 + juxtapositions beside	36	2	Section 3.6
7	4	2	6	6	3	3	6	18	three copies of 3 + juxtaposition underneath	36	2	Figure 3.34
8	5	2	5	10	2	4	5	20	**distance**	50	1	Figure 3.2; **JSTP**
9	6	2	3	15	1	5	3	15	**parallel class**	45	1	Figure 3.6; **JSTP**
10	6	2	3	30	1	10	6	30	two copies of 9 + juxtaposition beside	90	2	Section 3.6
11	6	2	6	15	2	5	6	30	two copies of 9 + juxtaposition underneath	90	2	Section 3.6
12	6	2	3	45	1	15	9	45	three copies of 9 + juxtaposition beside	135	2	Section 3.6
13	6	2	9	15	3	5	9	45	three copies of 9 + juxtaposition underneath	135	2	Section 3.6
14	6	2	3	60	1	20	12	60	four copies of 9 + juxtaposition beside	180	3	Section 3.6
15	6	2	6	30	2	10	12	60	two copies of 10 + juxtaposition underneath	180	3	Section 3.6
16	6	2	12	15	4	5	12	60	four copies of 9 + juxtaposition underneath	180	3	Section 3.6
17	7	2	7	21	2	6	7	42	**distance**	147	1	Figure 3.3
18	7	2	7	42	2	12	14	84	two copies of 17 + juxtaposition beside	294	2	Section 3.6
19	7	2	14	21	4	6	14	84	two copies of 17 + juxtaposition underneath	294	2	Section 3.6
20	7	2	7	63	2	18	21	126	three copies of 17 + juxtaposition beside	441	2	Section 3.6

S/N	v	k	h	p	n_c	n_r	λ	r	Construction	b	n^*	Solution
21	7	2	21	21	6	6	21	126	(rearrangement of 20) or three copies of 17+ juxtaposition underneath	441	2	Section 3.6
22	7	2	7	84	2	24	28	168	four copies of 17 + juxtaposition beside	588	3	Section 3.6
23	7	2	14	42	4	12	28	168	two copies of 18 + juxtaposition underneath	588	3	Section 3.6
24	7	2	21	28	6	8	28	168	juxtaposition of transpose of 17 beside 21	588	3	Section 3.6
25	8	2	4	28	1	7	4	28	**parallel class**	112	1	Figure 3.7
26	8	2	4	56	1	14	8	56	two copies of 25 + juxtaposition beside	224	2	Section 3.6
27	8	2	8	28	2	7	8	56	two copies of 25 + juxtaposition underneath	224	2	Section 3.6
28	8	2	4	84	1	21	12	84	three copies of 25 + juxtapositions beside	336	2	Section 3.6
29	8	2	12	28	3	7	12	84	three copies of 25 + juxtaposition underneath	336	2	Section 3.6
30	9	2	9	36	2	8	9	72	**distance**	324	2	Figure 3.4
31	9	2	18	18	4	4	9	72	**rearrangement of 30**	324	2	Figure 3.24; **JSTP**
32	9	2	9	72	2	16	18	144	two copies of 30 + juxtaposition beside	648	2	Section 3.6
33	9	2	18	36	4	8	18	144	two copies of 30 + juxtaposition underneath	648	2	Section 3.6
34	9	2	9	108	2	24	27	216	three copies of 30 + juxtaposition beside	972	3	Section 3.6
35	9	2	18	54	4	12	27	216	three copies of 31 + juxtaposition beside	972	3	Section 3.6
36	9	2	27	36	6	8	27	216	three copies of 30 + juxtaposition underneath	972	3	Section 3.6
37	9	2	9	144	2	32	36	288	four copies of 30 + juxtaposition beside	1296	2	Section 3.6
38	9	2	18	72	4	16	36	288	two copies of 33 + juxtaposition beside	1296	2	Section 3.6
39	4	3	4	4	3	3	8	12	**difference set** or **block complementation** of a Latin square of order 4	16	1	Figure 4.18

S/N	v	k	h	p	n_c	n_r	λ	r	Construction	b	n^*	Solution
40	6	3	2	10	1	5	4	10	**distance** + **column complementation**	20	1	Figure 4.19
41	6	3	2	20	1	10	8	20	two copies of 40 + juxtaposition beside	40	2	Section 4.7
42	6	3	4	10	2	5	8	20	two copies of 40 + juxtaposition underneath	40	2	Figure 4.21
43	6	3	2	30	1	15	12	30	three copies of 40 + juxtaposition beside	60	2	Section 4.7
44	6	3	6	10	3	5	12	30	three copies of 40 + juxtaposition underneath	60	2	Section 4.7
45	6	3	2	40	1	20	16	40	two copies of 41 + juxtaposition beside	80	3	Section 4.7
46	6	3	4	20	2	10	16	40	two copies of 41 + juxtaposition underneath	80	3	Section 4.7
47	6	3	8	10	4	5	16	40	four copies of 40 + juxtaposition underneath	80	3	Section 4.7
48	7	3	7	7	3	3	7	21	**difference set**	49	1	Figure 4.6
49	7	3	7	14	3	6	14	42	two copies of 48 + juxtaposition beside or **difference family**	98	1	Section 4.4
50	7	3	7	21	3	9	21	63	three copies of 48 + juxtaposition beside or **distance**	147	1	Figure 4.2
51	7	3	7	28	3	12	28	84	four copies of 48 + juxtaposition beside or **difference family**	196	2	Figure 4.9
52	7	3	7	35	3	15	35	105	five copies of 48 + juxtaposition beside	245	1	Section 4.4
53	7	3	7	42	3	18	42	126	two copies of 50 + juxtaposition beside	294	2	Section 4.3
54	7	3	14	21	6	9	42	126	two copies of 50 + juxtaposition underneath	294	2	Section 4.3
55	8	3	8	56	3	21	48	168	**difference family**	448	1	Section 4.4
56	8	3	8	112	3	42	96	336	two copies of 55 + juxtaposition beside	896	2	Section 4.4
57	8	3	16	56	6	21	96	336	two copies of 55 + juxtaposition underneath	896	2	Section 4.4
58	8	3	8	168	3	63	144	504	three copies of 55 + juxtapositions beside	1344	2	Section 4.4
59	8	3	24	56	9	21	144	504	three copies of 55 + juxtaposition underneath	1344	2	Section 4.4

S/N	v	k	h	p	n_c	n_r	λ	r	Construction	b	n^*	Solution
60	9	3	3	12	1	4	3	12	**MOLS/affine resolvable designs**	36	2	Figure 4.11
61	9	3	6	6	2	2	3	12	**rearrangement of 60**	36	2	Figure 4.14
62	9	3	3	24	1	8	6	24	two copies of 60 + juxtaposition beside	72	2	Section 4.5
63	9	3	6	12	2	4	6	24	two copies of 60 + juxtaposition underneath	72	2	Section 4.5
64	9	3	3	36	1	12	9	36	three copies of 60 + juxtaposition beside	108	3	Section 4.5
65	9	3	6	18	2	6	9	36	three copies of 61 + juxtaposition beside	108	3	Section 4.5
66	9	3	9	12	3	4	9	36	three copies of 60 + juxtaposition underneath	108	3	Section 4.5
67	9	3	3	48	1	16	12	48	four copies of 60 + juxtaposition beside	144	3	Section 4.5
68	9	3	6	24	2	8	12	48	four copies of 61 + juxtaposition beside	144	3	Section 4.5
69	9	3	12	12	4	4	12	48	rearrangement of 67 (or 68) or four copies of 60 + juxtaposition underneath or a Latin square with symbols the constituent Latin squares in 60	144	3	Figure 4.23
70	9	3	3	60	1	20	15	60	five copies of 60 + juxtaposition beside	180	3	Section 4.5
71	9	3	6	30	2	10	15	60	five copies of 61 + juxtaposition beside	180	3	Section 4.5
72	9	3	12	15	4	5	15	60	juxtaposition of transpose of 60 beside 69	180	3	Section 4.5
73	13	3	13	26	3	6	13	78	**difference family**	338	1	Figure 4.8
74	6	4	3	15	2	10	18	30	block complementation of 9	45	1	Figure 4.16
75	6	4	3	30	2	20	36	60	block complementation of 10	90	2	Section 4.6
76	6	4	6	15	4	10	36	60	block complementation of 11	90	2	Section 4.6
77	6	4	3	45	2	30	54	90	block complementation of 12	135	2	Section 4.6
78	6	4	9	15	6	10	54	90	block complementation of 13	135	2	Section 4.6
79	7	4	7	7	4	4	14	28	block complementation of 48	49	1	Figure 4.15
80	13	4	13	13	4	4	13	52	**difference set**	169	1	Figure 4.7

Table 7.2: Table of parameters for some RGSLRs

S/N	v	k	h	p	n_c	n_r	x	y	λ'	λ^*	r	Construction	b	Solution
1	5	2	5	5	2	2	2	2	2	3	10	Starter sets + cyclic development of initial blocks	25	Figure 5.21
2	5	2	5	15	2	6	2	2	7	8	30	Adjoining a BSLR to a RGSLR	75	Figure 5.29
3	6	2	3	6	1	2	1	4	1	2	6	Adapting BM (2001)-Method 2 (BTD)	18	Figure 5.3, if the symbol ∞ is replaced by 6
4	6	2	3	9	1	3	4	1	1	2	9		27	
5	6	2	3	12	1	4	2	3	2	3	12	Adjoining another RGSLR to the design in S/N 3 (∞ retained) after a suitable permutation	36	Figure 5.18
6	6	2	3	18	1	6	3	2	3	4	18	Adjoining two RGSLRs to the design in S/N 3 (∞ retained) after suitable permutations	54	Figure 5.20
7	6	2	3	21	1	7	1	4	4	5	21	Adjoining a $(3 \times 15)/2$ BSLR to the design in S/N 3	63	Figure 5.5
8	6	2	6	9	2	3	3	2	3	4	18	Transposition of the design in S/N 3 (∞ retained) and adjoining of two RGSLRs to it after suitable permutations	54	Section 5.6
9	6	2	6	12	2	4	4	1	4	5	24		72	
10	6	2	9	12	3	4	1	4	7	8	36		108	
11	6	2	9	18	3	6	4	1	10	11	54		162	

S/N	v	k	h	p	n_c	n_r	x	y	λ'	λ^*	r	Construction	b	Solution
12	7	2	7	7	2	2	2	4	2	3	14	Starter sets + cyclic development of initial blocks	49	Figure 5.23
13	7	2	7	14	2	4	4	2	4	5	28	Adjoining another RGSLR to the design in S/N 12 after a suitable permutation	98	Figure 5.25
14	7	2	7	28	2	8	2	4	9	10	56	Adjoining four RGSLRs after suitable permutations	196	Figure 5.27
15	8	2	4	8	1	2	1	6	1	2	8	Adapting BM (2001)-Method 1 (Starter)	32	Figure 5.6
16	8	2	4	12	1	3	5	2	1	2	12	Adjoining a $(4 \times 4)/2$ Trojan square to the design in S/N 15	48	Figure 5.9; see also Figure 5.10
17	8	2	4	16	1	4	2	5	2	3	16		64	
18	8	2	4	36	1	9	1	6	5	6	36	Adjoining a $(4 \times 28)/2$ BSLR to the design in S/N 15	144	Figure 5.8
19	8	2	8	12	2	3	3	4	3	4	24		96	
20	8	2	8	16	2	4	4	3	4	5	32		128	
21	8	2	8	20	2	5	5	2	5	6	40		160	
22	8	2	12	16	3	4	6	1	6	7	48		192	
23	8	2	12	20	3	5	4	3	8	9	60		240	
24	8	2	12	24	3	6	2	5	10	11	72		288	
25	9	2	9	9	2	2	2	6	2	3	18	Starter sets + cyclic development of initial blocks	81	Sections 5.7 and 5.8
26	9	2	9	27	2	6	6	2	6	7	54	Adjoining three RGSLRs after suitable permutations	243	Figure 5.28
27	10	2	5	10	1	2	1	8	1	2	10	Adapting BM (2001)-Method 1 (Starter)	50	Figure 5.13

S/N	v	k	h	p	n_c	n_r	x	y	λ'	λ^*	r	Construction	b	Solution
28	10	2	5	20	1	4	2	7	2	3	20	Adjoining another RGSLR to the design in S/N 27 after a suitable permutation	100	Figure 5.12
29	10	2	5	30	1	6	3	6	3	4	30	Adjoining two RGSLRs to the design in S/N 27 after suitable permutations	150	Figure 5.19
30	18	2	9	18	1	2	1	16	1	2	18	Adapting BM (2001)-Method 1 (Starter)	162	Figure 5.16
31	18	2	9	36	1	4	2	15	2	3	36	Adjoining another RGSLR to the design in S/N 30 after a suitable permutation	324	Figure 5.15
32	5	3	5	5	3	3	2	2	7	8	15	Starter sets + cyclic development of initial blocks; also by block complementation	25	Figure 6.1
33	5	3	5	15	3	9	2	2	22	23	45	Adjoining a BSLR to a RGSLR	75	Figure 6.6
34	6	3	2	4	1	2	3	2	1	2	4		8	
35	6	3	2	6	1	3	2	3	2	3	6		12	
36	6	3	2	8	1	4	1	4	3	4	8		16	
37	6	3	4	6	2	3	4	1	4	5	12		24	
38	6	3	4	8	2	4	2	3	6	7	16		32	
39	6	4	3	6	2	4	1	4	7	8	12	Direct construction via an exchange algorithm; also block complementation	18	Figures 6.2 and 6.5
40	6	4	3	12	2	8	2	3	14	15	24	Adjoining a RGSLR to another after a suitable permutation	36	Figure 6.7
41	8	4	2	8	1	4	3	4	3	4	8		16	
42	8	4	4	8	2	4	6	1	6	7	16		32	
43	8	6	4	8	3	6	1	6	17	18	24	Pooled starter sets + cyclic development of initial blocks; also block complementation	32	Figures 6.3 and 6.4

Bibliography

Abel, R. J. R. (1994). Forty-Three Balanced Incomplete Block Designs. *Journal of Combinatorial Theory, Series A 65*, 252–267.

Agrawal, H. (1966). Some Methods of Construction of Designs for Two-Way Elimination of Heterogeneity, 1. *Journal of the American Statistical Association 61*(316), 1153–1171.

Ai, M., K. Li, S. Liu, and D. K. J. Lin (2013). Balanced incomplete Latin square designs. *Journal of Statistical Planning and Inference 143*, 1575–1582.

Anderson, I. (1997). *Combinatorial designs and tournaments.* Oxford University Press, Oxford.

Ash, A. (1981). Generalized Youden Designs: Construction and Tables. *Journal of Statistical Planning and Inference 5*(1), 1–25.

Bailey, R. A. (1984). Quasi-complete Latin squares: construction and randomization. *Journal of the Royal Statistical Society: Series B (Methodological) 46*(2), 323–334.

Bailey, R. A. (1988). Semi-Latin Squares. *Journal of Statistical Planning and Inference 18*, 299–312.

Bailey, R. A. (1992). Efficient Semi-Latin Squares. *Statistica Sinica 2*, 413–437.

Bailey, R. A. (2004). *Association schemes: Designed experiments, algebra and combinatorics.* Cambridge University Press, Cambridge.

Bailey, R. A. (2009). Variance and concurrence in block designs, and distance in the corresponding graphs. *The Michigan Mathematical Journal 58*(1), 105–124.

Bailey, R. A. and P. J. Cameron (2009). Combinatorics of optimal designs. In S. Huczynska, J. D. Mitchell, and C. M. Roney-Dougal (Eds.), *Surveys in Combinatorics 2009*, Volume 365, pp. 19–73. Cambridge University Press, Cambridge.

Bailey, R. A. and P. E. Chigbu (1997). Enumeration of semi-Latin squares. *Discrete Mathematics 167/168*, 73–84.

Bailey, R. A. and H. Monod (2001). Efficient Semi-Latin Rectangles: Designs for Plant Disease Experiments. *Scandinavian Journal of Statistics 28*(2), 257–270.

Bailey, R. A., H. Monod, and J. P. Morgan (1995). Construction and optimality of affine-resolvable designs. *Biometrika 82*(1), 187–200.

Bailey, R. A. and G. Royle (1997). Optimal Semi-Latin Squares with Side Six and Block Size Two. *Proceedings of the Royal Society: Mathematical, Physical and Engineering Sciences 453*(1964), 1903–1914.

Bedford, D. and R. M. Whitaker (2001). A new construction for efficient semi-Latin squares. *Journal of Statistical Planning and Inference 98*, 287–292.

Bose, R. C. (1938). On the Application of the Properties of Galois Fields to the Problem of Construction of Hyper-Graeco-Latin Squares. *Sankhyā: The Indian Journal of Statistics 3*(4), 323–338.

Bose, R. C. (1939). On the Construction of Balanced Incomplete Block Designs. *Annals of Eugenics 9*, 353–399.

Bose, R. C. (1942). A note on the resolvability of balanced incomplete block designs. *Sankhyā: The Indian Journal of Statistics 6*(2), 105–110.

Bose, R. C. and K. R. Nair (1941). On Complete Sets of Latin Squares. *Sankhyā: The Indian Journal of Statistics 5*(4), 361–382.

Bose, R. C., S. S. Shrikhande, and E. T. Parker (1960). Further Results on the Construction of Mutually Orthogonal Latin Squares and the falsity of Euler's Conjecture. *Canadian Journal of Mathematics 12*, 189–203.

Cakiroglu, S. A. (2018). Optimal regular graph designs. *Statistics and Computing 28*, 103–112.

Caliński, T. and S. Kageyama (2003). *Block Designs: A Randomization Approach: Volume II: Design.* Springer-Verlag, New York.

Cameron, P. J. (1994). *Combinatorics: topics, techniques, algorithms.* Cambridge University Press, Cambridge.

Cheng, C.-S. (1978). Optimality of certain asymmetrical experimental designs. *The Annals of Statistics 6*(6), 1239–1261.

Cheng, C.-S. (1992). On the optimality of (M.S)-optimal designs in large systems. *Sankhyā: The Indian Journal of Statistics, Series A 54*, 117–125.

Cheng, C.-S. and R. A. Bailey (1991). Optimality of some two-associate-class partially balanced incomplete-block designs. *The Annals of Statistics 19*(3), 1667–1671.

Cheng, C.-S. and C.-F. Wu (1981). Nearly balanced incomplete block designs. *Biometrika 68*(2), 493–500.

Choi, K. C. and S. Gupta (2008). Confounded row–column designs. *Journal of statistical planning and inference 138*, 196–202.

Colbourn, C. J. (1996). Youden Designs, Generalized. In C. J. Colbourn and J. H. Dinitz (Eds.), *The CRC Handbook of Combinatorial Designs*, pp. 508–511. CRC press, Boca Raton.

Constantine, G. M. (1986). On the optimality of block designs. *Annals of the Institute of Statistical Mathematics 38*(1), 161–174.

Courrieu, P. (2005). Fast computation of Moore-Penrose inverse matrices. *Neural Information Processing-Letters and Reviews 8*(2), 25–29.

Darby, L. A. and N. Gilbert (1958). The Trojan square. *Euphytica 7*, 183–188.

Das, A. (2002). An introduction to optimality criteria and some results on optimal block design. *Design Workshop Lecture Notes, ISI, Kolkata*, 1–21.

Dash, S., R. Parsad, and V. Gupta (2014). Efficient Row-Column Designs with Two Rows. *Journal of the Indian Society of Agricultural Statistics 68*(3), 377–390.

Datta, A., S. Jaggi, C. Varghese, and E. Varghese (2014). Structurally Incomplete Row-Column Designs with Multiple Units per Cell. *Statistics and Applications 12*(1&2), 71–79.

Datta, A., S. Jaggi, C. Varghese, and E. Varghese (2015). Some Series of Row-Column Designs with Multiple Units per Cell. *Calcutta Statistical Association Bulletin 67*(265-266), 89–99.

Datta, A., S. Jaggi, C. Varghese, and E. Varghese (2016). Series of Incomplete Row-Column Designs with Two Units per Cell. *Metodološki Zvezki 13*(1), 17–25.

Datta, A., S. Jaggi, E. Varghese, and C. Varghese (2017). Generalized confounded row–column designs. *Communications in Statistics-Theory and Methods 46*(12), 6213–6221.

Donev, A. N. (1998). Construction of non-standard row-column designs. In R. Payne and P. Green (Eds.), *Proceedings of the 13th Symposium in Computational Statistics: COMPSTAT*, pp. 275–280. Springer-Verlag, Berlin.

Durier, C., H. Monod, and A. Bruetschy (1997). Design and analysis of factorial sensory experiments with carry-over effects. *Food Quality and Preference 8*(2), 141–149.

Eccleston, J. A. and A. Hedayat (1974). On the theory of connected designs: characterization and optimality. *The Annals of Statistics 2*(6), 1238–1255.

Edmondson, R. N. (1998). Trojan square and incomplete Trojan square designs for crop research. *The Journal of Agricultural Science 131*, 135–142.

Godolphin, J. (2019a). Conditions for connectivity of incomplete block designs. *Quality and Reliability Engineering International 35*(5), 1279–1287.

Godolphin, J. (2019b). Construction of row–column factorial designs. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 81*(2), 335–360.

Hanani, H. (1961). The Existence and Construction of Balanced Incomplete Block Designs. *The Annals of Mathematical Statistics 32*(2), 361–386.

Hedayat, A. S., J. Stufken, and W. G. Zhang (1995). Contingently and Virtually Balanced Incomplete Block Designs and their Efficiencies under various Optimality Criteria. *Statistica Sinica 5*, 575–591.

Jacroux, M. (1980). On the E-optimality of regular graph designs. *Journal of the Royal Statistical Society. Series B (Methodological) 42*(2), 205–209.

Jacroux, M. A. (1978). On the properties of proper (M, S) optimal block designs. *The Annals of Statistics 6*(6), 1302–1309.

John, J. A. (1981). Efficient cyclic designs. *Journal of the Royal Statistical Society. Series B (Methodological) 43*(1), 76–80.

John, J. A. and T. J. Mitchell (1977). Optimal Incomplete Block Designs. *Journal of the Royal Statistical Society, Series B (Methodological) 39*(1), 39–43.

John, J. A. and E. R. Williams (1982). Conjectures for Optimal Block Designs. *Journal of the Royal Statistical Society, Series B (Methodological) 44*(2), 221–225.

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

John, P. W. M. (1980). *Incomplete block designs*, Volume 1. Marcel Dekker, Inc., New York.

Jones, B. and J. A. Eccleston (1980). Exchange and interchange procedures to search for optimal designs. *Journal of the Royal Statistical Society: Series B (Methodological) 42*(2), 238–243.

Kadowaki, S. and S. Kageyama (2009). Existence of affine α-resolvable PBIB designs with some constructions. *Hiroshima Mathematical Journal 39*, 293–326.

Keedwell, A. D. and J. Dénes (2015). *Latin Squares and their Applications*. Elsevier, Amsterdam.

Kiefer, J. (1975). Balanced Block Designs and Generalized Youden Designs, I. Construction Patchwork. *The Annals of Statistics 3*(1), 109–118.

Kreher, D. L., G. F. Royle, and W. Wallis (1996). A family of resolvable regular graph designs. *Discrete Mathematics 156*, 269–275.

Morgan, J. P. (2007). Optimal Incomplete Block Designs. *Journal of the American Statistical Association 102*(478), 655–663.

Nelder, J. A. (1965). The analysis of randomized experiments with orthogonal block structure. I. Block structure and the null analysis of variance. *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences 283*(1393), 147–162.

Parsad, R. (2006). A Note on Semi-Latin Squares. *Journal of Indian Society of Agricultural Statistics 60*(2), 131–133.

Penrose, R. (1955). A generalized inverse for matrices. *Proceedings of the Cambridge Philosophical Society 51*, 406–413.

Plemmons, R. J. and R. E. Cline (1972). The generalized inverse of a nonnegative matrix. *Proceedings of the American Mathematical Society 31*(1), 46–50.

Preece, D. A. (1996). Youden Squares. In C. J. Colbourn and J. H. Dinitz (Eds.), *The CRC Handbook of Combinatorial Designs*, pp. 511–514. CRC Press, Boca Raton.

Preece, D. A. and G. H. Freeman (1983). Semi-Latin Squares and Related Designs. *Journal of the Royal Statistical Society, Series B (Methodological) 45*(2), 267–277.

Raghavarao, D. (1971). *Constructions and Combinatorial Problems in Design of Experiments.* John Wiley & Sons, Inc., New York.

Raghavarao, D. and L. V. Padgett (2005). *Block Designs: Analysis, Combinatorics, and Applications.* World Scientific, New Jersey.

Rao, V. R. (1958). A note on balanced designs. *The Annals of Mathematical Statistics 29*(1), 290–294.

Rojas, B. and R. F. White (1957). The Modified Latin Square. *Journal of the Royal Statistical Society, Series B (Methodological) 19*(2), 305–317.

Ruiz, F. and E. Seiden (1974). On Construction of Some Families of Generalized Youden Designs. *The Annals of Statistics 2*(3), 503–519.

Searle, S. R. (1982). *Matrix Algebra useful for Statistics.* John Wiley & Sons, Inc., New York.

Shah, B. V. (1959). A generalisation of partially balanced incomplete block designs. *The Annals of Mathematical Statistics 30*(4), 1041–1050.

Shah, K. R. and B. K. Sinha (1989). *Theory of Optimal Designs*. Springer-Verlag, New York.

Shah, K. R. and B. K. Sinha (1996). Row–column designs. In S. Ghosh and C. R. Rao (Eds.), *Handbook of Statistics 13*, pp. 903–937. Elsevier, New York.

Shrikhande, S. S. (1951). Designs for Two-Way Elimination of Heterogeneity. *The Annals of Mathematical Statistics 22*(2), 235–247.

Soicher, L. H. (2013). Optimal and efficient semi-Latin squares. *Journal of Statistical Planning and Inference 143*, 573–582.

Stinson, D. R. (2004). *Combinatorial Designs: Constructions and Analysis*. Springer-Verlag, New York.

Street, A. P. and D. J. Street (1987). *Combinatorics of Experimental Design*. Oxford University Press, New York.

Tianyao, S. and T. Yu (2010). Optimal efficiency balanced designs and their constructions. *Journal of statistical planning and inference 140*, 2771–2777.

Uto, N. P. and R. A. Bailey (2020). Balanced semi-Latin rectangles: Properties, existence and constructions for block size two. *Journal of Statistical Theory and Practice 14*(51).

Williams, E. R. and J. A. John (1996). Row-column factorial designs for use in agricultural field trials. *Journal of the Royal Statistical Society: Series C (Applied Statistics) 45*(1), 39–46.