Another proof of Cruse’s theorem and a new necessary condition for completion of partial Latin squares (Part 3.)*

Béla Jónás

Aug 17, 2022  Budapest, Hungary

Abstract

A partial Latin square of order \(n\) can be represented by a 3-dimensional chess-board of size \(n \times n \times n\) with at most \(n^2\) non-attacking rooks. Based on this representation, we apply a uniform method to prove the M. Hall’s, Ryser’s and Cruse’s theorems for completion of partial Latin squares. With the help of this proof, we extend the scope of Cruse’s theorem to compact bricks, which appear to be independent of their environment.

Without losing any completion you can replace a dot by a rook if the dot must become rook, or you can eliminate the dots that are known not to become rooks. Therefore, we introduce primary and secondary extension procedures that are repeated as many times as possible. If the procedures do not decide whether a PLSC can be completed or not, a new necessary condition for completion can be formulated for the dot structure of the resulting PLSC, the BUG condition.

MSC-Class: 05B15

Keywords: Latin square, partial Latin square

Abbreviations

| Abbreviation | Description                          |
|--------------|--------------------------------------|
| LS           | Latin square                         |
| LSC          | Latin super cube                     |
| d-LSC        | Latin super cube of dimension \(d\)  |
| PLS          | partial Latin square                 |
| PLSC         | partial Latin super cube             |
| RBC          | remote brick couple                  |
| RAC          | remote axis couple                   |
| BUG          | Bivalue Universal Grave              |

*This is the third part of our consecutive papers on Latin and partial Latin squares and on bipartite graphs. Each work holds the terminology and notation of the previous ones [3, 4].
1 Introduction

Generally, we define a set of candidates $P_{ij}$ for each empty cell $(i, j)$ of a PLS as follows: $k \in P_{ij}$ exactly if the cell $(i, j, k)$ of the proper PLSC has a dot. A PLSC can be projected to a PLS in the usual way, the only difference is that each empty cell of the PLS is connected to a set of candidates. The Goldwasser’s square [4] will be referred to as GW6 hereafter. When illustrating a smaller PLS, such as GW6, the candidates are written in the empty cells, but with smaller numbers, as is usual when solving Sodoku puzzles. The GW6 prepared in this way is depicted in the Figure 1.1. So, the 35 in the cell (5,4) means that $P_{34} = \{35\}$.

![GW6](image)

Figure 1.1.

2 Basic Results of Completion

Definition 2.1. Let $P$ be a PLSC and $T$ a brick of size $r \times s \times t$ containing all the rooks of $P$, where $T$ is permuted to the origin, $t < n$ and $t$ is on the symbol axis. The cover sheet of $T$ is the matrix of size $n \times n$, whose cell with coordinate $(i, j)$ is one if there are no rooks in the cells of $T$ with coordinate $(i, j, c)$, where $c \in \{1, 2, \ldots, t\}$, and zero otherwise. So, the matrix is partly filled, as illustrated in the Figure 2.1.

![Cover Sheet](image)

Figure 2.1.
The uniform method of proving M. Hall’s, Ryser’s and Cruse’s theorems consists in constructing a cover sheet in all three cases and extending it to a \((0,1)\)-matrix of size \(n \times n\) and decomposing into a sum of permutation matrices.

Let \(P\) be a PLSC of order \(n\) and suppose that \(P\) consists of some completely filled column layers, row layers or symbol layers permuted next to each other. The Figure 2.2 depicts these yellow \(n^2\)-bricks that contain \(s_n\), \(r_n\), and \(t_n\) rooks, respectively.

Let \(P\) be a PLSC where the first \(r\) row layers are completed. The projected version of this PLSC is a Latin rectangle of size \(r \times n\). An example of a Latin rectangle of size \(r \times n\) and the two conjugates of this PLS are shown in the Figure 2.3. In our case, they are conjugates of each other, so \(r = s = t = 5\) for the corresponding PLSCs in the Figure 2.2.

In case of \(n^2\)-bricks the matrix is completely filled and the example in the right-hand side of Figure 2.4 is the cover sheet of the PLS shown on the right-hand side of Figure 2.3.

**Theorem 2.2 (M. Hall [2]).** Every \(r \times n\) Latin rectangle, \(1 \leq r < n\), can be completed to a Latin square of order \(n\).

**Proof.** Let \(P\) be the PLSC form of the conjugated PLS denoted by \((j,k,i)\). Then each of the layers \(1, 2, \ldots, t\) contains \(n\) rooks. Let the cover sheet of \(P\) be the matrix \(M\). Each row and each column of \(M\) contains exactly \(n - t\) ones. Then, based on the König’s theorem [5], the matrix \(M\)
decomposes into the sum of \((n-t)\) permutation matrices, that is

\[ M = L_{t+1} + L_{t+2} + \ldots + L_n \]

where \(L_k\) is a permutation matrix for all \(k \in \{t+1, t+2, \ldots, n\}\). Put a rook in the cell of the PLSC \(P\) with coordinate \((i, j, k)\) if the element of the matrix \(L_k\) with coordinate \((i, j)\) is one, where \(k \in \{t+1, t+2, \ldots, n\}\). Then each layer of \(P\) contains \(n\) rooks, and all of the rooks are non-attacking, so we get a completion of \((j, k, i)\) conjugate of \(P\). The \((i, j, k)\)-conjugate of this completion is a completion of \(P\).

\[\square\]

**Remark 2.3.** If a PLSC \(P\) has a completion, then each element of the main class of \(P\) also has a completion.

To prove his theorem Ryser first proved a combinatorial theorem.

**Theorem 2.4** (Ryser’s combinatorial theorem). Let \(A\) be a matrix of \(r\) rows and \(n\) columns, composed entirely of zeros and ones, where \(1 \leq r < n\). Let there be exactly \(k\) ones in each row, and let \(N(i)\) denote the number of ones in the \(i\)-th column of \(A\). If, for each \(i = 1, 2, \ldots, n\) holds, that

\[ k - (n-r) \leq N(i) \leq k \]

then \(n-r\) rows of zeros and ones may be adjoined to \(A\) to obtain a square matrix with exactly \(k\) ones in each row and in each column.

and then

**Theorem 2.5** (Ryser [6]). Let \(T\) be an \(r\) by \(s\) Latin rectangle based upon the integers \(1, 2, \ldots, n\). Let \(N(i)\) denote the number of times that the integer \(i\) occurs in \(T\). A necessary and sufficient condition in order that \(T\) may be extended to a Latin square of order \(n\) is that

\[ N(i) \geq r + s - n \]

for each \(i \in \mathbb{Z}_n\).
Using Theorem 2.4 we prove with the help of cover sheet.

**Proof.** Let \( P \) be a PLSC derived from the conjugate of the Latin rectangle \( T \) denoted by \((k, i, j)\).

Then each column layer of \( P \) contains exactly \( s \) rooks and each rook is in the layers \( 1, 2, \ldots, s \), as shown in the Figure 2.5 and \( N(i) \geq r + s \) for all \( i = 1, 2, \ldots, n \) due to the Ryser’s condition. Take the cover sheet of the yellow brick \( T \). The zeros do not matter, so they are hereafter omitted. For the matrix \( M \) holds that each of the first \( r \) columns contains \( n - s \) ones. In the row layer \( i \) of \( P \) there are \( N(i) \) rooks and obviously \( N(i) \leq s \). Thus, the matrix \( M \) contains \( R(i) = r - N(i) \) ones in each row, as the example shows in the Figure 2.6.

If \( k = (n - s) \), then there are \( k \) ones in each column of the matrix \( M \) and each row contains \( R(i) = r - N(i) \) ones.

Because of the Ryser’s condition

\[ N(i) \geq r + s - n \Rightarrow r - R(i) \geq r + s - n \Rightarrow (n - s) = k \geq R(i) \]

Since each row contains at most \( k \) ones and because of \( N(i) \leq s \) it holds that

\[ (n - s) \leq n - N(i) \Rightarrow k = (n - s) \leq n - N(i) = (n - r) + R(i) \]

ergo \( k - (n - r) \leq R(i) \). That is, based on Theorem 2.4, the grey part of \( M \) can be filled by ones such that each row and each column of \( M \) contains exactly \( k \) ones, as indicated in the Figure 2.6.
Let $M$ be the sum of the permutation matrices derived by the König’s theorem \cite{5}
\[ M = L_{s+1} + L_{s+2} + \ldots + L_n \]

Let $T_1$ be the vertical auxiliary brick of $T$ (green brick) and $T_2$ be the horizontal auxiliary brick of $T_1$ (grey brick). Now you can fill the layer $k$ based on $L_k$, where $k = s+1, s+2, \ldots, n$, therefore $(T_1 \cup T_2)$ and $(T \cup T_1)$ form completed $n^2$-bricks, as depicted in the Figure 2.7. Removing the "new" rooks from the brick $T_2$, the $n^2$-brick $(T \cup T_1)$ can be completed by M. Hall’s theorem \cite{2}. The corresponding conjugate of the result is a completion of the original (yellow) brick $T$.

![Figure 2.7.](image-url)

With the help of the method used in the previous proof, we prove the Cruse’s theorem.

**Theorem 2.6** (Cruse’s theorem). Let $P$ be a PLSC in $H_n^3$ for which $\text{ch}(P)$ is a real brick $T_0$ of size $r \times s \times t$ ($r, s, t < n$) with $c_0$ rooks. The brick $T_0$ can be embedded in a Latin square of order $n$ exactly if the following four conditions are met:

1) Each row layer of $T_0$ contains at least $s + t - n$ rooks

2) Each column layer of $T_0$ contains at least $r + t - n$ rooks

3) Each symbol layer of $T_0$ contains at least $r + s - n$ rooks

4) $c_0 \leq \text{cap}(n, r, s, t)$

Conditions 1)-4) are collectively referred to as the **Cruse’s condition system**. The conditions are obviously necessary due to the Distribution Theorem. We only prove that they are sufficient. First, we prove a lemma.

**Lemma 2.7.** Let $T$ be a matrix of size $p \times q$ and let $0 \leq c_1, c_2, \ldots, c_p \leq q$ be integers. Suppose that $N = c_1 + c_2 + \ldots + c_p$ and for integers $m$ and $M$ it holds that $0 \leq m, M \leq p$ and $mq \leq N \leq Mq$. Then $N$ ones can be placed in the cells of $T$ so that there are exactly $c_j$ ones in the column $j$ for each $j \in \{1, 2, \ldots, p\}$, and $m \leq N(i) \leq M$ for each $i \in \{1, 2, \ldots, q\}$, where $N(i)$ denotes the number of ones in the row $i$. 

6
Proof. For each $j \in \{1, 2, \ldots, p\}$ place $c_j$ ones in the column $j$ from bottom to top and permute the columns so that the number of ones contained does not increase compared to the number in the previous column, shown in the left-hand side of Figure 2.8. Notice, that if there is a one in a cell, then there is a one in each cell below it in its column and in each cell on the left in its row, so the rectangle to the lower left corner is full of ones. If there are at most $M$ ones in the bottom row, then there are at most $M$ ones in each row, one side of the inequality holds. If there are more than $M$ ones in the bottom row, we push the ones of the columns $M + 1, \ldots, p$ upwards by 1 (green cells on the left-hand side of the Figure 2.8). There is no ones in the first row of these columns (yellow cells in the left-hand side of the Figure 2.8), otherwise there would be a rectangle of size $(M + 1) \times q$ full of ones. Since $N - M \leq (q - 1)M$, the procedure can be continued with the subsystem $T^*$ formed by disregarding the bottom line containing exactly $M$ ones. We reconstruct the monotonic decrease of the columns in $T^*$ by permuting the whole columns (including the cells below $T^*$) and continue the procedure with $T^*$. Finally, $N(i) \leq M$ for all $i \leq 1, 2, \ldots, q$.

Starting from the resulting structure of the previous procedure, we produce the matrix for which the other side of the inequality holds. Permute the rows so that the number of ones contained increases monotonically, with the fewest ones in the first row. If $N(1) \geq m$, then we are ready. So, $N(1) < m$. Permute the columns so that the ones in the first row are to the left. In the bottom row, going from right to left, find the first cell that contains a one (green cells on the right-hand side of the Figure 2.8). If there is no single in the lowest green cells of the columns $p, p - 1, \ldots, m + 1$, then the bottom row and thus each row contains at most $m$ ones, and the first row contains less, so it is a contradiction. Push the one found in the last row up into the first row, then continue the procedure from the beginning with the whole system thus obtained. Finally, $m \leq N(i)$ for all $i \in \{1, 2, \ldots, q\}$.

Remark 2.8. If $N/q$ is an integer, then it is possible that each row contains the same number of ones, namely $m = M = N/q$.
Now we prove a generalization of Ryser’s combinatorial theorem 2.4.

**Theorem 2.9.** Let $A$ be a rectangle of the size $r \times s$ of a matrix of order $n$. Suppose that each row of $A$ contains at least $s + k - n$ and at most $k$ ones, each column of $A$ contains at least $r + k - n$ and at most $k$ ones. Let $a_0$ be the number of ones in $A$. An illustration is shown in the Figure 2.9. If it holds for $a_0$ that

1) $(r + s - n)k \leq a_0$

2) $a_0 \leq rs - (n - k)(r + s - n)$

then the vertical auxiliary brick of $A$, the brick $B$, can be filled with ones such that each column of the brick $A \cup B$ contains exactly $k$ ones, each row contains at least $s + k - n$ and at most $k$ ones, consequently, $A \cup B$ satisfies the conditions of the Ryser’s combinatorial theorem.

![Figure 2.9.](image)

**Proof.** We show that the conditions for $a_0$ are also necessary. A filled brick $A \cup B$ contains $sk$ ones. The blue brick $B$ must contain at least $(n - r)(s + k - n)$ ones, otherwise it would have a row that contains less than $(s + k - n)$ ones, i.e.,

$$a_0 \leq sk - (n - r)(s + k - n)$$

The blue brick $B$ can contain at most $(n - r)k$ ones, otherwise there would be a row containing more than $k$, therefore $sk - (n - r)k = (r + s - n)k \leq a_0$. Let $S(j)$ denote the number of ones in
the column $j$ of brick $A$ for each $j \in \{1, 2, \ldots, s\}$. Let $c_j = k - S(j)$ for $j \in \{1, 2, \ldots, s\}$, then

$$N = sk - \sum_{j=1}^{s} S(j).$$

We show that for $m = (s + k - n)$, $M = k$, $p = s$ and $q = (n - r)$, the conditions of the Lemma 2.7 are satisfied, that is, $qm \leq N \leq qM$. Because of the condition 1) of Theorem 2.9

$$\sum_{j=1}^{s} S(j) \geq (r + s - n)k$$

ergo

$$N = sk - \sum_{j=1}^{s} S(j) \leq sk - (r + s - n)k = (n - r)k = qM$$

hence $N \leq qM$. By the condition 2) of Theorem 2.9

$$a_0 \leq rs - (n - k)(r + s - n).$$

Then because of

$$N = sk - \sum_{j=1}^{s} S(j),$$

$$N \geq sk - [rs - (n - k)(r + s - n)] = (n - r)(s + k - n) = qm,$$

so $qm \leq N$. That is, the conditions of the lemma are satisfied, so the brick $B$ can be filled with ones such that each column of the brick $A \cup B$ contains exactly $k$ ones and each row of the brick $A \cup B$ contains at least $s + k - n$ and at most $k$ ones, i.e., $A \cup B$ satisfies the condition of Ryser’s theorem.

Now we prove the Theorem 2.6.

**Proof.** Suppose that a brick $T$ of size $r \times s \times t$ satisfies the Cruse’s condition system. Rotate the brick $T$ so that $r \geq s \geq t$. We show that the conditions of the Theorem 2.9 are satisfied for the cover sheet of $T$. Let $k = (n - t)$. If there are at least $s + t - n$ rooks in each row layer, then there are at most $s - (s + t - n) = (n - t) = k$ ones in the corresponding row of the cover sheet. If there are at most $t$ rooks in each row layer, then there are at least $(s - t) = s + (n - t) - n = s + k - n$ ones in the corresponding row of the cover sheet. It can also be seen that the corresponding columns of the cover sheet has at most $r - (r + t - n) = (n - t) = k$ and at least $r + (n - t) = r + k - n$ ones.

The yellow brick $T_0$ contains at most $\text{cap}(n, r, s, t)$ rooks, so the yellow part of the gray cover sheet contains at least

$$rs - \text{cap}(n, r, s, t) = (r + s - n)(n - t) = (r + s - n)k$$

ones, ergo condition 1) of Theorem 2.9 is satisfied. Each symbol layer of the yellow brick has at least $(r + s - n)$, so the layers of $T_0$ contain a total of at least $t(r + s - n)$ rooks, so the yellow
Figure 2.10.

part of the cover sheet contains at most \(rs - t(r + s - n) = \text{cap}(n, r, s, k)\) ones, that is, condition 2) of Theorem 2.9 is satisfied. The cover sheet of the yellow brick can be filled into a matrix of size \(n \times n\), each row and column of which contains exactly \(k\) ones. With the help of the resulting matrix, the symbol layers \(t + 1, t + 2, \ldots, n\) can be completed with rooks such that each layer contains \(n\) rooks and the rooks are non-attacking. The resulting \(n^2\)-brick is denoted by \(T_{n-t}\) (grey brick) and a part of it, the vertical auxiliary brick of \(T_0\) is denoted by \(T_1\) (green brick). We prove that the conditions of the Ryser’s theorem are satisfied for the \(n\)-brick \((T_0 \cup T_1)\) of size \(r \times s \times n\). For each symbol layer \(\sigma\) of the yellow brick \(T_0\), \(N(\sigma) \geq r + s - n\) holds because of condition 3) of the Cruse’ condition system. Since this is a necessary condition for completion of a given symbol layer, it is obviously met for a completed layer and the brick \(T_1\) is a part of a completed \(n^2\)-brick. So, for each symbol layer of the brick \(T_1\) is satisfied that \(N(\sigma) \geq r + s - n\). Ergo, if you remove all rooks in the \(n^2\)-brick \(T_{n-t}\) outside brick \(T_1\), then the PLSC consisting of the \(n\)-brick \((T_0 \cup T_1)\) can be completed because of the Ryser’s theorem.

Remark 2.10. Since \(\text{cap}(n, r, s, t) = rs - (n - t)(r + s - n)\), the capacity condition for the yellow brick \(T_0\) ensures that the Ryser’s condition can be met for all suitable layers of the green auxiliary bricks of \(T_0\). But, \(\text{cap}(n, r, s, t) = st - (n - r)(s + t - n)\) and \(\text{cap}(n, r, s, t) = rt - (n - s)(r + t - n)\), so the Ryser’s condition can be met for all suitable layers of all auxiliary bricks of \(T_0\).

3 Embedding Compact Bricks

Let \(T_0\) be a brick of size \(a \times b \times c\). \(T_0\) can be artificially created, that means \(T_0\) is not necessarily a brick of a PLSC. Suppose that the rooks in \(T_0\) are non-attacking and the files containing rooks do not contain dots.

Definition 3.1. A layer of \(T_0\) is called possible if the number of parallel files of the layer containing dots or rooks are identical in both directions of the layer. This number is called the potential of the layer of \(T_0\).
The symbol layer 1 with potential 4 can be seen on the right-hand side of Figure 3.1. If you place a dot in the cell (3,3,1) then the symbol layer 1 is no longer possible. Let $N_i(i,T_0)$, $N_j(j,T_0)$ and $N_\sigma(\sigma,T_0)$ denote the potentials of the layers of $T_0$, where $i = \{1,2,\ldots,a\}$, $j = \{1,2,\ldots,b\}$ and $\sigma = \{1,2,\ldots,c\}$. Easy to check, that all layers of $T_0$ in the left-hand side of Figure 3.1 are possible.

**Definition 3.2.** The brick $T_0$ is called compact if each layer of $T_0$ is possible and

$$\sum_{i=1}^{a} N_i(i,T_0) = \sum_{j=1}^{b} N_j(j,T_0) = \sum_{\sigma=1}^{c} N_\sigma(\sigma,T_0)$$

This value is called the potential of the brick $T_0$.

The cases $c = n$, $b = c = n$ and $a = b = c = n$ are allowed. The potential of the compact brick $T_0$ is 15 in the left-hand side of Figure 3.1.

Let $T_0$ be a compact brick, where

$$\sum_{i=1}^{a} N_i(i,T_0) = \sum_{j=1}^{b} N_j(j,T_0) = \sum_{\sigma=1}^{c} N_\sigma(\sigma,T_0) = p_0$$

**Definition 3.3.** Let $P$ be an empty PLSC of order $n$, where each cell of $P$ has a dot and let $T_0$ be a compact brick with potential $p_0$ of size $a \times b \times c$, where $a, b, c \leq n$. Embedding of $T_0$ in $P$ means that we put $T_0$ into $P$ and in each file of $T_0$ that has rooks or dots we eliminate the dots outside $T_0$ and replace $n^2 - p_0$ dots by rooks outside $T_0$ in such a way that the rooks are non-attacking. The compact brick $T_0$ is embeddable if such a set of rooks exists. This set of rooks is called a full extension of $T_0$.

For example, the compact brick $T_0$ can be the closure hull of the dot structure of GW6 and the rooks (symbols in the white cells) can be considered as a full extension of the brick $T_0$. $T_0$ is the yellow brick in the Figure 3.2, its height is 6. The potential of $T_0$ is 12. Outside $T_0$ there are no rooks in the files containing dots in $T_0$ and each file has one rook that has no dots.
Definition 3.4. A compact brick $T_0$ with potential $p_0$ is called solvable or it has a solution if $p_0$ dots of $T_0$ can be replaced by rooks in such a way, that the rooks are non-attacking.

Remark 3.5. Obviously, a solution of $T_0$ and a full extension of $T_0$ combined form a Latin square.

Remark 3.6. After placing $T_0$ into an empty PLSC and eliminating the proper dots outside $T_0$, the compact brick $T_0$ is “independent of the environment”, i.e., the structure of dots inside $T_0$ has the information whether $T_0$ is solvable or not. It does not have any impact on $T_0$ if you replace a dot with a rook outside $T_0$.

Let $T_0$ be a compact brick of size $a \times b \times c$ of a PLSC of order $n$ with potential $p_0$ and let $N_i(i,T_0), N_j(j,T_0)$ and $N_z(\sigma,T_0)$ denote the potentials of the layers of $T_0$, where $i \in \{1, 2, \ldots, a\}$, $j \in \{1, 2, \ldots, b\}$ and $\sigma \in \{1, 2, \ldots, c\}$. From the definition of the potential of $T_0$ we have

$$\sum_{i=1}^{a} N_i(i,T_0) = \sum_{j=1}^{b} N_j(j,T_0) = \sum_{\sigma=1}^{c} N_z(\sigma,T_0) = p_0$$

Definition 3.7. The compact brick $T_0$ of size $a \times b \times c$ with potential $p_0$ satisfies the Cruse’s condition system if the followings hold for a given $n$.

1. $N_i(i,T_0) \geq b + c - n$ for all $1 \leq i \leq a$
2. $N_j(j,T_0) \geq a + c - n$ for all $1 \leq j \leq b$
3. $N_z(\sigma,T_0) \geq a + b - n$ for all $1 \leq \sigma \leq c$
4. $p_0 \leq \text{cap}(n,a,b,c)$

Theorem 3.8. Let $T_0$ be a compact brick of size $a \times b \times c$ with potential $p_0$, put $T_0$ into an empty PLSC of order $n$, where $a, b, c \leq n$. If $T_0$ satisfies the Cruse’s condition system, then $T_0$ is embeddable, in other words, there exists a full extension of $T_0$ in the PLSC given.
Proof. When you compose the cover sheet of the compact brick $T_0$, you need to treat the files containing rooks or dots in $T_0$ as if they would have a rook inside $T_0$. Then the proof of the Theorem 2.6 also proves this statement.

It is easy to check, that the closure hull of the dot structure of GW6, i.e., the yellow brick in the Figure 3.2, satisfies the Cruse’s condition system.

Cruse [3] proved that the capacity condition holds for the Cruse’s square and from Corollary 4.25 we know that GW6 also satisfies the capacity condition, ergo

**Corollary 3.9.** For $n \geq 6$ there exists a PLSC $P$ of order $n$ that contains the closure hull of the dot structure of GW6 in an embedded way and thus satisfies the capacity condition, but nevertheless $P$ is not completable.

Figure 3.3 shows examples for $n = 7$, $n = 8$ and $n = 9$.

![Figure 3.3](image)

**Remark 3.10.** It is easy to prove that for each compact brick $T_0$ there exists $N$, for example $N = \text{diam}(T_0)$, such that $T_0$ satisfies the Cruse’s condition system for all $n > N$.

### 4 Elimination, BUGs

The terms PLS and the corresponding PLSC, candidate and the corresponding dot, symbol and the corresponding rook are used in parallel, but the context should always make it clear which one we are thinking of. The examples normally are in PLS form, but they are always a projection of the current PLSC. If it improves clarity, we show all three relevant projections of the PLSC (primary conjugates). We always assume that the automatic elimination has already been performed, hence, the files containing rooks do not contain dots.

**Definition 4.1.** Let $P$ be a PLSC. A dot of $P$ is called *desolate* if there exists a file containing only the given dot. The cell is called *desolate* if it has a desolate dot. To treat a desolate cell means to replace the dot in the given cell with a rook and to eliminate the dots that are in the Hamming sphere of radius 1 whose center is the given desolate cell.
Let P be a PLSC.

Algorithm 4.2 (Primary Extension).

Step 1: Let \( j = 0 \) and \( P_0 = P \)

Step 2: If each file of \( P_j \) contains one rook, then stop (\( P_j \) is completed)

Step 3: If there is an eliminated file in \( P_j \), then stop (\( P_j \) is not completable)

Step 4: If there are no desolate cells, then stop

Step 5: Increment \( j \) by one, arbitrarily select a desolate cell, denote it by \( D_j \) and replace the dot in this cell by a rook

Step 6: Perform automatic elimination, denote the result by \( P_j \), and continue from Step 2

We eliminate dots if and only if there is a rook in one of the files containing the given dot, and we replace a dot with a rook if and only if the dot is desolate. If we stopped because there is an eliminated file, then \( P \) is not completable, and we do not define primary extensions for such a PLSC. In other cases, the result of the procedure will be denoted by \( P^* \) (\( P^* = P_0 = P \) if \( P \) has no desolate dots at all). \( P^* \) is either an LSC or \( P^* \) has at least two dots in each file that contains no rooks. It is obvious that \( P \subseteq P^* \). We prove, that \( P^* \) is unique, so the PLSC \( P^* \) can be considered the primary extension of \( P \).

Suppose that there is at least one desolate dot in \( P \). Let \( A_1 \) and \( A_2 \) be two runs of the procedure 4.2, and the result of \( A_1 \) is \( R_1 \), the result of \( A_2 \) is \( R_2 \). The run \( A_1 \) defines a cell sequence \( D_1, D_2, \ldots, D_k \) and the run \( A_2 \) defines a cell sequence \( E_1, E_2, \ldots, E_m \). The procedure starts with a desolate cell, so the cells \( D_1 \) and \( E_1 \) are desolate. Both sequences contain all the cells in that order, in which the procedure treated them. When the procedure is completed, the treated cells contain non-attacking rooks.

Corollary 4.3. The Hamming distance of any two cells of the sequence \( D_1, D_2, \ldots, D_k \) or of the sequence \( E_1, E_2, \ldots, E_m \) is at least 2.

Statement 4.4. The runs \( A_1 \) and \( A_2 \) produce the same result in the following sense. If \( R_1 \) contains an eliminated file, then \( R_2 \) also contains an eliminated file. If \( R_1 \) is an LSC (completed PLSC), then \( R_2 \) also is an LSC and \( R_1 = R_2 \). Otherwise, \( R_1 = R_2 \), including both the rook and dot structures.

Proof. Let \( S_1(D_j) \) denote the Hamming sphere of radius 1 whose center is \( D_j \), where \( j \in \{1, 2, \ldots, k\} \). Let \( F(j) = S_1(D_1) \cup S_1(D_2) \cup \ldots \cup S_1(D_j) \), where \( j \in \{1, 2, \ldots, k\} \). If we arbitrarily pick a cell from the sequence \( D_1, D_2, \ldots, D_k \) in \( P \) (prior to any run of the procedure) and treat it then we eliminate all dots in \( S_1(D_j) \). Due to Corollary 4.3 and because of \( S_0(D_j) = \{D_j\} \), we do not eliminate dots in any cell of the sequence \( D_1, D_2, \ldots, D_k \). Therefore, the elimination in the set \( F(j) \) can be done in any order for any \( j \in \{1, 2, \ldots, k\} \). If the result of the run \( A_1 \) is that there is an eliminated file, then the result of the run \( A_2 \) is that there is an eliminated file.
Otherwise, run A₂ stops because there are no more desolate dots. If a desolate dot is eliminated, an eliminated file is created. So, cell D₁ has not been eliminated, but then cell D₂ occurs in run A₂ and is treated, so cell D₂ becomes desolate. If it has been eliminated, then there is an eliminated file if not, then it is treated and cell D₃ becomes desolate, etc... If all cells in the sequence D₁, D₂, ..., Dₖ have been treated, then we know from the run A₁, that there is an eliminated file. So, the run A₂ finds eliminated file, which is a contradiction. The roles of runs A₁ and A₂ are symmetric, so if run A₁ did not stop because of an eliminated file, then run A₂ do not stop because of an eliminated file. Suppose that neither A₁ nor A₂ found an eliminated file. Then we do not eliminate the dot in cell E₁ in run A₁, otherwise an eliminated file is created. This means, that cell E₁ appears in the cell sequence of run A₁ at the latest when there is no other desolate cell. Let D₁ = E₁. Similarly, we do not eliminate the dot in cell E₂ in run A₁, otherwise an eliminated file is created. So, cell E₂ appears in the cell sequence of run A₁ at the latest when the cell D₁ = E₁ is treated, because we know from the run A₂ that E₂ is desolate if E₁ is treated. Let D₂ = E₂. We do not eliminate the dot in cell E₃ in run A₁, otherwise an eliminated file is created. So, cell E₃ appears in the cell sequence of run A₁ at the latest when the cell D₁ = E₁ and D₂ = E₂ are treated, because we know from the run A₂ that E₃ is desolate if E₁ and E₂ are treated in any order. And so on. Consequently, any cells treated in run A₂ occurs in run A₁. The roles of A₁ and A₂ are symmetric, so two runs result in the same rook structure, which also implies the same dot structure due to elimination of the dots in cells of F(k). Ergo, P* is unique.

Corollary 4.5. An LSC Q is a completion of P if and only if it is a completion of P*. Thus, if P* is an LSC, then P has exactly one completion, namely P*.

Although in the primary extension some dots were replaced by rooks and some dots were eliminated, the result may still contain more dots that cannot become rooks in any extension.

Definition 4.6. Let P be a PLSC. A dot of P is deceptive if at least one of the next two conditions holds for the given dot:

1) (T₀, T₃) is a stuffed RBC of P and the dot is either in T₀ or in T₃

2) (T₀, T₂) is a balanced RBC of a layer of P and one brick of the RBC is perfect and the dot is in the other brick

Remark 4.7. If we replace a deceptive dot by a rook, then we create an overloaded RBC or an underweighted perfect brick, thus the PLSC we get is not completable.

Remark 4.8. We define the same set of deceptive dots in any order we find them, so the set of deceptive dots of a PLSC is unique, so that the deceptive dots can also be eliminated in any order.

Let P be a PLSC. Suppose that the primary extension of P exists. Denote it by P*.
Algorithm 4.9 (Secondary extension).

Step 1: Let \( j = 0 \) and \( P_0^* = P^* \)

Step 2: If the capacity condition does not hold for \( P_j^* \), then stop.

Step 3: If there are no deceptive dots in the PLSC, then stop.

Step 4: Eliminate the deceptive dots and perform primary extension

Step 5: If primary extension created an eliminated file, then stop

Step 6: Increment \( j \) by one, denote the result by \( P_j^* \) and continue from Step 2

If we stopped in Step 2 or Step 5, then \( P \) is not completable, and we do not define secondary extensions for such a PLSC. In other cases, let the result of the procedure is denoted by \( P^{**} \) \((P^{**} = P_0^{**} = P^* \text{ if } P^* \text{ has no deceptive dots at all}). P^{**} \) is either an LSC or has at least two dots in each file that contains no rooks. The primary extensions \( P_j^* \) are unique and

\[
P \subseteq P_1^* \subseteq P_2^* \subseteq \ldots \subseteq P_j^* \subseteq \ldots \subseteq P^{**}
\]

The set of deceptive dots also are unique, so \( P^{**} \) is unique, thus \( P^{**} \) can be considered the secondary extension of \( P \).

Corollary 4.10. An LSC \( Q \) is a completion of \( P \) if and only if it is a completion of \( P^{**} \). Hence, if \( P^{**} \) is an LSC, then \( P \) has exactly one completion, namely \( P^{**} \).

We apply the primary and secondary extension algorithms on the Cruse’s square and on its conjugates. The Cruse’s square has no desolate dots, so its primary extension is itself, as it can be checked in the Figure 4.1.

![Figure 4.1](image)

Since no new rooks are created and as we will see later in Statement 4.23 and Corollary 4.25, the capacity condition holds for Cruse’s square, ergo the capacity condition is also satisfied after each elimination step. Let \( T_0 \) be the yellow brick of size \( 4 \times 4 \times 4 \), let the remote mate of \( T_0 \) is the green brick \( T_3 \) of size \( 2 \times 2 \times 2 \). The capacity of the RBC \((T_0, T_3)\) is \(\frac{4 \cdot 4 \cdot 4 + 2 \cdot 2 \cdot 2}{6} = 12\).
The RBC contains 12 rooks, so it is stuffed. Thus, we can eliminate the candidates 1, 2, 3 and 4 in the empty cells of the yellow brick. So, in the empty cells of the yellow brick only 5 and 6 can occur as candidates. In the same way, we can eliminate the candidates 5 and 6 in the empty cells of the green brick (in our case there are no such candidates), so only 1, 2, 3 and 4 can appear as candidates in the empty cells of the green square. The eliminated state is shown in the Figure 4.2.

There are no dots in the yellow brick, so every layer of the brick is perfect. If $T$ is a layer of the yellow brick and contains $4+4-6 = 2$ rooks, then the candidates can be eliminated in the remote mate of $T$ in the given layer. Layer 1 is shown in the middle of Figure 4.3. The yellow brick in layer 1 has two rooks and its Ryser-number is 2, so candidates 1 and the corresponding dots in the part of the layer below the green brick can be eliminated.

The column layer 1 shown in the middle of Figure 4.2 also contains two rooks. The layer can be seen on the left-hand side of Figure 4.3. The two candidates that can be eliminated are the dots with coordinates $(5, 6, 1)$ and $(6, 5, 1)$. These dots are the candidates with coordinates $(6, 1, 5)$ and $(5, 1, 6)$ shown in the middle of Figure 4.2. The row layer 2, shown in the middle of Figure 4.2 also contains two rooks. The layer can be seen on the right-hand side of Figure 4.3. The two candidates that can be eliminated are the dots with coordinates $(5, 6, 2)$ and $(6, 5, 2)$. These dots are the candidates with coordinates $(2, 5, 6)$ and $(2, 6, 5)$ shown in the middle of Figure 4.2. If we perform all the eliminations allowed by the perfect yellow layers with two rooks, we obtain the following squares in the Figure 4.4.

**Definition 4.11.** A structure of dots in a PLSC is called *minimum candidate structure*, or mCS for short if each file of the PLSC that contains no rooks, contains exactly two dots.
Remark 4.12. The number of dots in an mCS is even, twice the number of empty cells. Consequently, if the dot structure of a PLSC is an mCS, then each conjugate of the proper PLS has the same number of empty cells.

Remark 4.13. The Latin square \(GW6\) has an mCS, since all empty cells of the three primary conjugates contain exactly two candidates as depicted in Figure 4.5.

Let \(P\) be a PLSC that has an mCS. Let the dots of \(P\) be the vertices of a graph and there is an edge between two vertices if and only if there is a file containing these two vertices. The resulting graph is 3-regular.

Definition 4.14. The component of the graph representation of an mCS and the structure of dots corresponding to this component is called Bivalue Universal Grave, for short BUG (the name comes from Sudoku).

Remark 4.15. The 3-dimensional image of the mCS structure of the \(GW6\) is illustrated on the right-hand side of Figure 4.6. The colored version on the left-hand side helps to identify the row layers, layer 6 is green, layer 5 is red and layer 4 is blue, but this coloring has nothing to do with the vertex coloring of the graph. In the graph representation of the mCS structure of the \(GW6\) the dots of each row are connected, and the row layers are connected to each other (black edges), so the whole graph is a connected. The corresponding graph of the mCS structure in the secondary extension of the Cruse’s square also is connected, therefore, both are BUGs.

Definition 4.16. A BUG has a solution or is solvable if half of the dots of the BUG can be replaced with rooks so that there is exactly one rook in each file of the BUG.
An immediate necessary condition for completion of a PLSC follows.

**Theorem 4.17** (BUG condition). *A necessary condition for the completion of a PLSC is that all BUGs in the secondary extension of the PLSC must be solvable.*

From the graph representation it is evident that a BUG is solvable if and only if the vertices of the corresponding graph can be colored with two colors, which is equivalent to the graph being bipartite, which is true exactly if the graph does not contain a cycle with an odd number of edges. There is cycle of length 9 with vertices $A - B - C - D - E - F - G - H - I$ in the Figure 4.6. Therefore, the $GW6$ is not completable.

We have the similar problem in the secondary extension of the Cruse’s square on the left-hand side of Figure 4.7. If you want to color the vertices of the cycle

$$(1, 2, 5), (1, 6, 5), (1, 6, 4), (5, 6, 4), (5, 2, 4), (5, 2, 6), (1, 2, 6)$$

alternately red and blue, you will get stuck at candidate 6 of cell $(1, 2)$, because it already has a red and a blue neighbor, so the given cycle is of odd length, namely 7. There could exist more cycles is of odd length, another example

$$(1, 2, 5), (6, 2, 5), (6, 2, 4), (6, 5, 4), (1, 5, 4), (1, 5, 6), (1, 2, 6)$$

is shown on the right-hand side of Figure 4.7. Thus, Cruse’s square has no completion either.

**Remark 4.18.** It can be seen that the length of the smallest odd cycle of a BUG is 7.

We now summarize our observations on BUGs. If, starting from any cell of a BUG, considering one of the two candidates as a symbol, we can solve all the cells of the BUG without inconsistency, then
starting from the same cell, considering the other candidate as a symbol, we can solve the BUG and each cell of the BUG has the other symbol as during the first solution,

starting from the given cell, the BUG has exactly two disjoint solutions,

connected bipartite graphs are uniquely partitioned into two color classes, so starting from any cell in the BUG we get one of the previous two solutions.

**Theorem 4.19** (BUG theorem). A BUG has exactly 0 or 2 solutions.

**Remark 4.20.** The statement is obviously also true for Sudoku squares. Sudoku puzzles have precisely one completion, so they cannot contain BUGs. This is the reason why the Sudoku technique BUG+1 came about.

As we mentioned before Cruse [3] proved that the capacity condition holds for the Cruse’s square. To be precise he proved, that the characteristic matrix $M$ of the Cruse’s square can be extended to a triply stochastic matrix $M^*$ by replacing some 0’s with $1/2$s. The symbol layers of the triply stochastic matrix are depicted by Cruse in his proof. Like all triply stochastic matrices, $M^*$ satisfies the capacity condition and contains all the 1’s of $M$ which are corresponding to the rooks of the original square. So, each RBC of the Cruse’s square satisfies the capacity condition, even if you add some $1/2$s to the number of rooks. Easy to check, that the $1/2$s in the Cruse’s proof are in the cells that contain dots in the proper BUG depicted in the middle square of Figure 4.4. Thus, each file of the matrix contains either one 1 or two $1/2$s. The other cells of the files contain 0. Hence, it is clear from the construction, that the proper matrix is triply stochastic. We don’t know how Cruse identified the cells that contain $1/2$s in his proof, maybe by ignoring the deceptive candidates that destroy immediately the completability, but his reasoning can be extended in the following way.
**Definition 4.21.** A component of the dot structure of a PLSC is called \( k \)-uniformly-distributed if each file of the component contains exactly \( k \) dots for some \( k > 1 \), or simply uniformly-distributed if \( k \) has no role.

**Definition 4.22.** A PLSC is uniformly-distributed if all components of the dot structure are uniformly-distributed, but the values of \( k \) may vary from component to component.

**Statement 4.23.** Let \( P \) be a PLSC. If you can eliminate some dots such that the remaining PLSC is uniformly-distributed and has no eliminated files, then \( P \) satisfies the capacity condition.

**Proof.** If you replace each rook by 1 and each dot of a \( k \)-uniformly-distributed component by \( 1/k \), then you get a triple stochastic matrix. \( \square \)

**Remark 4.24.** The PLSC derived from secondary extension of the Cruse’s square and the PLSC derive from GW6 each contain one 2-uniformly-distributed mCSs.

**Corollary 4.25.** The secondary extension of the Cruse’s square, shown in the middle of Figure 4.4 and the PLS GW6, shown in the Figure 3.2, both satisfy the capacity condition (and still have no completions).

**References**

[1] Cruse A.B., A Number-Theoretic Function Related to Latin Squares, J. Combinatorial Theory (A) 19, pp. 264–277 (1975)

[2] Marshall Hall, An existence theorem for Latin squares, Bull. Amer. Math. Soc., 51, 1945, 387–388.

[3] B. Jónás, Distribution of rooks on a chess-board representing a Latin square partitioned by a subsystem (Part 1.), arXiv:2208.04113v1 [math.CO] (Submitted on Aug 08, 2022).

[4] B. Jónás, Analysis of subsystems with rooks on a chess-board representing a partial Latin square (Part 2.), arXiv:2208.06166v1 [math.CO] (Submitted on Aug 12, 2022).

[5] D. König, Theorie der endlichen und unendlichen Graphen, Chelsea, 1950. 170–178.

[6] H.J. Ryser, A combinatorial theorem with an application to Latin rectangles, Proc. Amer. Math. Soc. 2 (1951) 550–552.