ON WALSH CODE ASSIGNMENT

Boris Tsybakov, *Member, IEEE*, Edward Tiedemann, *Senior Member, IEEE*, and Peter Gaal
QUALCOMM Incorporated, 5775 Morehouse Dr., San Diego, CA 92121-1714
{borist, etied, pgaal}@qualcomm.com

***Abstract*-The paper considers the problem of orthogonal variable spreading Walsh-code assignments. The aim of the paper is to give assignments that can avoid both complicated signaling from the BS to the users and blind rate and code detection amongst a great number of possible codes. The assignments considered here use a partition of all users into several pools. Each pool can use its own codes that are different for different pools. Each user has only a few codes assigned to it within the pool. The problem is stated in the paper as a combinatorial one expressed in terms of the assignment binary matrix M. Matrix M depends on *n*, the number of users in a pool; *k*, the total number of Walsh codes in the pool; and *l*, the number of Walsh codes assigned to each user within the pool. A solution to the problem is given as a construction of M, which has the assignment property defined in the paper. Two constructions of such M are presented. The constructions are optimal in the sense that they give the minimal number *l* for given *n* and *k*. The optimality follows from a proven necessary condition for the existence of M with the assignment property. We describe the implementation complexity associated with the presented optimal assignment.**

***Index Terms*- Walsh code, wireless communications, multiuser networks.**

1. INTRODUCTION

A direct-sequence code division multiple-access (CDMA) third generation wireless network (see [1]) employs the orthogonal variable spreading factor (OVSF) Walsh codes [2]-[7]. In OVSF systems, the mobile stations (MS's or users) that require higher transmission rate in the current frame of the forward channel (from the base station (BS) to the MS) should use shorter length codes. Information about which code the BS will use can be signaled to the MS on a dedicated channel or the MS can perform blind rate and code detection [8]. However, the signaling takes extra resources whereas performing the blind rate and code detection is complicated if there are a large number of possible codes out of which the BS has to choose a code for transmission. It is possible to reduce these difficulties by making the number of codes that can be used for transmission to each MS as small as possible (the Walsh codes that can be used for transmission to a particular MS are called below the codes *assigned* to this MS). This can be achieved if all users are partitioned into several pools, each pool can use its own codes taken from the set of all available codes, and a user of a pool monitors the pool codes assigned to it only.

For voice vocoder, there are four relative rates for transmission in a frame, the full rate, ½-rate, ¼-rate, and 1/8-rate. According to [9],[10], in a 3GPP2 system, the rates of one user can be modeled by a second-order Markov chain. The chain has the following steady-state probabilities: $q_1 = 0.2909$ for the full rate, $q_2 = 0.0388$ for the ½-rate, $q_3 = 0.0723$ for the ¼-rate, and $q_4 = 0.5980$ for the 1/8-rate. Since the Walsh code is a tree code, there are $k_1 = 64$ codes of the full rate, $k_2 = 128$ codes of the ½-rate, $k_3 = 256$ codes of the ¼-rates, and $k_4 = 512$ codes of the 1/8-rate [9].

It can be considered as four different code distributions over MS's. The distributions are denoted as S1, S2, S3, and S4. For S1, a different full rate code is dedicated to each user and when the BS transmits to any given user in any frame, the BS uses the code dedicated to this user.

For S2, a different ½-rate code is dedicated to each user and the BS uses such code in any frame where the user requests the ½-, ¼-, or 1/8- rates. The BS grants the full rate code from the set of available codes to a user only when the user requests it.

For S3, a different ¼-rate code is dedicated to each user and the BS uses such code in any frame where the user requests the ¼-rate or 1/8-rate transmission. The BS grants the full rate from the set of available codes to a user when the user requests the full or ½- rates.

For S4, a different 1/8-rate code is dedicated to each user and the BS uses such code in any frame where the user requests the 1/8-rate transmission. The BS grants the full rate from the set of available codes to a user when the user requests the full, ½-, or ¼- rates.

Granting a code is possible in S2, S3, and S4 if the BS has sufficient number of full rate codes in the set of available codes. Otherwise, some randomly chosen users that do not get a grant are blocked in the frame.

If for example, the total number of users is $U = 100$, then the probability for any given user to be blocked in a frame (denoted by pi(U) for the distributions Si, $i = 2,3,4$) is 1- $k_1 / U = 0.36$ for S1, 0.151 for S2, $2.302 \cdot 10^{-3}$ for S3, and $2.503 \cdot 10^{-4}$ for S4. Fig. 1 shows pi(U) as a function of U for Si, $i = 2,3,4$. The probabilities for the distributions Si, $i = 2,3,4$ are calculated as the ratio of the expectation of number of blocked users and U, i.e. as

$$\mathrm{p}i(U) = \frac{1}{U} \sum_{x=1+\left\lfloor (k_i - U)/2^{i-1} \right\rfloor}^{U} \{x - \left\lfloor (k_i - U)/2^{i-1} \right\rfloor\} \binom{U}{x} \left(\sum_{j=1}^{i-1} q_j \right)^x \left(1 - \sum_{j=1}^{i-1} q_j \right)^{U-x}. \quad (1.1)$$

In the calculation of (1.1), we suppose independence between users.

Also, it can be considered three additional code distributions denoted as S5, S6, and S7.

For S5, a different ½-rate code is dedicated to each user and the BS uses such code in any frame where the user requests the ½- or ¼- or 1/8- rates as it is in S2. But unlike in S2, the BS grants one additional ½-rate code from the set of available codes to a user when the user requests full rate; after the grant, the user has two ½-rate codes that are equivalent to one full rate code.

For S6, a different ¼-rate code is dedicated to each user and the BS uses such code in any frame where the user requests the ¼- or 1/8-rate transmission as it is in S3. But unlike in S3, the BS grants three additional ¼-rate codes from the set of available codes to a user when the user requests the full rate, and the BS grants one additional ¼-rate codes when the user requests the ½-rate.

For S7, a different 1/8-rate code is dedicated to each user and the BS uses such code in any frame where the user requests the 1/8-rate transmission as it is in S4. But not as in S4, the BS grants the additional seven 1/8-rate codes, three 1/8 rate codes, and one 1/8-rate code from its set of available codes to a user when the user requests the full rate, ½-rate, and ¼-rate, respectively.

The additional code granting is possible in S5, S6, and S7 if the BS has enough of additional codes in the set of available codes. Otherwise, some randomly chosen users that do not get the additional codes are blocked in the frame.

For S5, the blocking probability as a function of U is

$$\mathrm{p5}(U) = \frac{1}{U} \sum_{x=1+k_2-U}^{U} \{x - k_2 + U\} \binom{U}{x} q_1^x (1 - q_1)^{U-x}, \quad \binom{U}{x} \hat{=} \frac{U!}{x!(U-x)!}. \quad (1.2)$$

For S6 and S7, the blocking probabilities p5(U) and p5(U), respectively, are upperbounded as

$$\mathrm{p6}(U) \le \frac{1}{U} \sum_{x=1+\left\lfloor (k_3 - U)/3 \right\rfloor}^{U} \{x - \left\lfloor (k_3 - U)/3 \right\rfloor\} \binom{U}{x} \left(\sum_{j=1}^{2} q_j \right)^x \left(1 - \sum_{j=1}^{2} q_j \right)^{U-x} \quad (1.3)$$

and

$$\mathrm{p7}(U) \le \frac{1}{U} \sum_{x=1+\left\lfloor (k_4 - U)/7 \right\rfloor}^{U} \{x - \left\lfloor (k_4 - U)/7 \right\rfloor\} \binom{U}{x} \left(\sum_{j=1}^{3} q_j \right)^x \left(1 - \sum_{j=1}^{3} q_j \right)^{U-x}. \quad (1.4)$$

The inequalities in (1.3) and (1.4) are because we assumed granting three (instead of one) additional ¼-rate codes to a user when the user requested the ½-rate in case S6, seven (instead of three) additional 1/8-rate codes to a user when the user requested the ½-rate in case S7, and seven (instead of one) additional 1/8-rate codes to a user when the user requested the ¼-rate in case S7. Fig. 2 shows pi(U) or its upper bounds as functions of U for Si, $i = 5,6,7$.

The probabilities for S3 and S4 are small enough to allow using these distributions. However, the set of available codes in these cases may be too large; if $U = 100$, there are 39 codes in S3 and 51 codes in S4. Similar or worth situation is in the cases of the distributions S5, S6, and S7. For such large set of available codes, it is not easy to implement signaling or blind code detection. If there were a possibility to have the Walsh code pools with n users (for example, $n = 10, 14,$ or 18), assign to each user a small number l of codes (for example, $l = 3, 4,$ or 5, respectively), and get the blocking probability less than 0.01, then this would result in significantly easier implementation.

In this paper, we shall show how to design code assignments with minimum number l of codes for pools with given n and k, where n is the number of users assigned to a pool of codes, and k is the number of different codes included in a pool of codes. For example, if the total number of users is $U = 100$, we can place them in 10 pools with $n = 10$ users in each pool. For S3, a pool can have $k \le 4$ since the set of available codes has $\lfloor (k_3 - U)/4 \rfloor$=39 full-rate codes. For S4, a pool can have $k \le 5$ since the set of available codes has $\lfloor (k_4 - U)/8 \rfloor$=51 full-rate codes. Our assignment with the second construction having $n = 6$, $k = 4$, and $l = 2$ (see Section 3 below) will give, for example, the blocking probability for S3 about

$$\frac{1}{n}\sum_{x=1+k}^{n}(x-k)\frac{n!}{(n-k)!k!}(q_1+q_2)^x(1-q_1-q_2)^{n-x} = 0.003.$$

Similar results can be obtained for distributions S5, S6, and S7.

The problem of designing the optimal Walsh code assignment for pool users is combinatorial. In the rest of this section, we introduce notation, give a concept of the assignment property, and state the problem. For the clerity of the terminology used, we call the assigned code units as the full-rate codes as in S3 and S4.

Let n denote the number of mobile stations (users) in a pool, k denote the total number of available different full rate Walsh codes in the pool, which equals the total number of pool users that could receive voice simultaniously in the same frame interval at the full rate, and l denote the number of full rate codes available to each pool user. The n users are numbered by $1,\ldots,n$. The k available Walsh codes are numbered by $1,\ldots,k$. The Walsh code numbers that are available (assigned) to the user i are denoted by $b_{ij} \in \{1,...,k\}$, $i = 1,...,n$, $j = 1,...,l$. For example, if $l = 3$ and $k \ge 5$, it can be that $b_{11} = 1$, $b_{12} = 3$, $b_{13} = 5$. This means that the Walsh codes 1, 3, and 5 are available to user 1.

A table (or matrix) $[b_{ij}]$, $i = 1,...,n$, $j = 1,...,l$ that indicates the assigned l Walsh codes to each user is called the *assignment table*. A given assignment table has the *assignment property* if for any different $i_1,...,i_k$ out of $\{1,...,n\}$, there exist $j_1,...,j_k$ such that all $b_{i_1 j_1},...,b_{i_k j_k}$ are different. This means that if an assignment table has the assignment property, the BS can choose the different full codes for any k (or less) MS's and simultaniously transmit to these k (or less) MS's in the frame.

In our consideration, the codes that are assigned to a user are described by a row of binary elements, of length k with l symbols '1' and $k - l$ symbols '0'. The row corresponding to the user i is the row i of an $n \times k$ matrix called the *assignment matrix*. The assignment matrix is denoted by $\mathbf{M} = \mathbf{M}(n,k,l)$. If at position j, $j = 1,...,k$, in row i of $\mathbf{M}$ has the symbol '1', this means that the code j is one of the Walsh codes assigned to the user i. For example, if Walsh codes 1, 3, and 5 are assigned to user $i = 1$ then, assuming $k = 5$, in our matrix representation, row 1 of $\mathbf{M}$ will be $[1 \quad 0 \quad 1 \quad 0 \quad 1]$.

Now we give a necessary and sufficient condition that a matrix $\mathbf{M}(n,k,l)$ and the correspoding assignment table have the assignment property. Let us denote by $\mathbf{K}$ the binary $k \times k$ matrix composed by some k rows of the $n \times k$ matrix $\mathbf{M}(n,k,l)$**.** *If for any k rows of $\mathbf{M}(n,k,l)$, which gives all rows of $\mathbf{K}$, there is a row permutation of $\mathbf{K}$ that gives a $k \times k$ matrix $\mathbf{G}$ with all* '1' *main diagonal, then $\mathbf{M}(n,k,l)$ and the corresponding assignment table have the assignment property.*

Here is an example of the table with $n = 6$, $k = 4$, $l = 2$ and its $\mathbf{M}(6,4,2)$ that have the assignment property,

$$[b_{ij}] = \begin{bmatrix} 1 & 2 \\ 2 & 3 \\ 1 & 3 \\ 1 & 4 \\ 2 & 4 \\ 3 & 4 \end{bmatrix}, \qquad \mathbf{M}(6,4,2) = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}.$$

The fact that the table and $\mathbf{M}(6,4,2)$ have the assignment property, can be simply checked by an exhaustive test.

Our problem is constructing the tables with the assignment property for given n, k, and l. We note that there exist combinations of n, k, and l for which the problem is trivial. For example, if $k = l$, the table that has the all identical rows $[1 \ \ldots \ k]$ has the assignment property. However, generally the problem is not trivial.

Another problem altogether is finding a simple algorithm that for the proposed table construction, allows to find the different Walsh codes to be assigned for any given k users. Again, for certain combinations of n, k, and l, the problem is trivial. For example, it is trivial if $n = k$ since a simple table has $1,\ldots,k$ main diagonal.

Also in terms of n, k, l, we want to find a necessary condition for existence of tables with the assignment property.

For each constructed table, it is important to determine whether or not it is optimal in the sense that it has the minimum number l for given n and k.

In the following, the problems are considered and treated in terms of finding binary matrices $\mathbf{M}$, $\mathbf{K}$, and $\mathbf{G}$. Section 2 gives a construction of $\mathbf{M}(n,k,l)$ for $k = n/2$, $n = 6+4i$, $i \in \{0,1,...\}$, $l = (k+1)/2$ with the assignment property. Section 3 gives a construction of $\mathbf{M}(n,k,l)$ for $k = (n/2)+1$, $n = 6+4i$, $i \in \{0,1,...\}$, $l = k/2$ with the assignment property. For $\mathbf{M}(n,k,l)$ presented in those two sections, there are the simple algorithms that allow choosing the Walsh codes for any given k users. First, the algorithms select the rows to be used in constructing $\mathbf{K}$ and then by cyclic shift of the rows of $\mathbf{K}$, they give $\mathbf{G}$. Section 4 gives a necessary condition for the existence of $\mathbf{M}$ with the assignment property. The condition $(n-k)k < nl_{\max}$ obtained for matrix $\mathbf{M}$, which can have no more than $l_{\max}$ '1' symbols in any of its rows. The obtained condition holds tightly for matrices $\mathbf{M}$ described in Sections 2 and 3, which shows that those matrices are optimal.

We note that if we remove any w rows from a matrix $\mathbf{M}(n,k,l)$ of our constructions, we obtain the matrix $\tilde{\mathbf{M}}(n-w,k,l)$ that still has the assignment property. This means that the constructions give the matrices with the assignment property for any $n \geq k$.

2. FIRST CONSTRUCTION

In this section by Theorem 1, we present our first construction of matrix $\mathbf{M}$ with the assignment property. First, we need to introduce some definitions and notation related to $n \times k$ matrix $\mathbf{M}$ that will be considered in Theorem 1.

The definition of matrix $\mathbf{M}$ for the first construction. The binary $n \times k$ matrix $\mathbf{M}$ with $k = n/2$, $n = 6+4i$, $i \in \{0,1,...\}$ (note that k is odd) is such that its row j is the $(j-1)$ position rightward horizontal cyclic shift of row $[1 \ \ldots \ 1 \ \ 0 \ \ldots \ 0]$ which has $l = (k+1)/2$ symbols '1' and $k-l$ symbols '0'. (The word "horizontal" is used here to distinguish between the cyclic shift of row symbols and the cyclic shift of rows. The latter will be used and defined later and will be called a vertical cyclic shift.) Below for illustration, we use an example of matrix $\mathbf{M}$ with $n = 22$, $k = 11$, $l = 6$.

We denote by $\mathbf{M}_{\mathrm{u}}$ the upper $k \times k$ submatrix of matrix $\mathbf{M}$ and by $\mathbf{M}_l$ the lower $k \times k$ submatrix of matrix $\mathbf{M}$. The matrices $\mathbf{M}_{\mathrm{u}}$ and $\mathbf{M}_l$ are identical.

We choose arbitrary k rows of $\mathbf{M}$ and compose of them a $k \times k$ matrix $\mathbf{K}$ that will be described later.

If neither row j and $j+k$ (they are identical), are chosen for $\mathbf{K}$, the row j of $\mathbf{M}_{\mathrm{u}}$ is called a *null row* and marked by symbol L, $j = 1,...,k$.

If only one of the rows j and $j+k$, is chosen for $\mathbf{K}$, the row j of $\mathbf{M}_{\mathrm{u}}$ is called a *single row* and is marked by symbol S.

If both rows j and $j+k$, are chosen for $\mathbf{K}$, the row j of $\mathbf{M}_{\mathrm{u}}$ is called a *double row* and is marked by symbol D.

In our example the matrices $\mathbf{M}_{\mathrm{u}}$ and $\mathbf{K}$ are the following:

$$\mathbf{M}_{\mathrm{u}} \hat{=} \begin{bmatrix} \mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & \text{the row 1 is the null row } L \\ 0 & \mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & \text{the row 2 is the single row } S \\ 0 & 0 & \mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & \text{the row 3 is the double row } D \\ 0 & 0 & 0 & \mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & 0 & \text{the row 4 is the double row } D \\ 0 & 0 & 0 & 0 & \mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & \text{the row 5 is the single row } S \\ 0 & 0 & 0 & 0 & 0 & \mathbf{1} & 1 & 1 & 1 & 1 & 1 & \text{the row 6 is the single row } S \\ 1 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & 1 & 1 & 1 & 1 & \text{the row 7 is the null row } L \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & 1 & 1 & 1 & \text{the row 8 is the single row } S \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & 1 & 1 & \text{the row 9 is the double row } D \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & 1 & \text{the row 10 is the null row } L \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & \text{the row 11 is the single row } S \end{bmatrix}, \qquad (2.1)$$

(We shall explain it later why one of the 1's in each row of $\mathbf{M}_{\mathrm{u}}$ is bold.)

$$\mathbf{K} = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & \text{row 13 of } \mathbf{M}, \text{ or row 2 of } \mathbf{M}_{\mathrm{u}} \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & \text{row 3 of } \mathbf{M}, \text{ or row 3 of } \mathbf{M}_{\mathrm{u}} \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & \text{row 14 of } \mathbf{M}, \text{ or row 3 of } \mathbf{M}_{\mathrm{u}} \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & \text{row 4 of } \mathbf{M}, \text{ or row 4 of } \mathbf{M}_{\mathrm{u}} \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & \text{row 15 of } \mathbf{M}, \text{ or row 4 of } \mathbf{M}_{\mathrm{u}} \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & \text{row 16 of } \mathbf{M}, \text{ or row 5 of } \mathbf{M}_{\mathrm{u}} \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & \text{row 6 of } \mathbf{M}, \text{ or row 6 of } \mathbf{M}_{\mathrm{u}} \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & \text{row 8 of } \mathbf{M}, \text{ or row 8 of } \mathbf{M}_{\mathrm{u}} \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & \text{row 9 of } \mathbf{M}, \text{ or row 9 of } \mathbf{M}_{\mathrm{u}} \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & \text{row 20 of } \mathbf{M}, \text{ or row 9 of } \mathbf{M}_{\mathrm{u}} \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & \text{row 11 of } \mathbf{M}, \text{ or row 11 of } \mathbf{M}_{\mathrm{u}} \end{bmatrix}.$$

Generally, in matrix $\mathbf{K}$, the order of rows is the same as in $\mathbf{M}_{\mathrm{u}}$ except that

(1) the L rows are omitted in $\mathbf{K}$ and

(2) instead of each D row of $\mathbf{M}_{\mathrm{u}}$, the matrix $\mathbf{K}$ has two identical D rows that follow one after another.

This completes the definition of matrix $\mathbf{K}$.

In the considered example, the number of the null rows is 3, the number of the single rows is 5, and the number of the double rows is $d = 3$. In the general case, the number of D rows (denoted as d) equals the number of L rows. The number of S rows plus the twice number of D rows (i.e., plus $2d$) equals k.

Theorem 1. *Let the binary $n \times k$ matrix $\mathbf{M}$ and for any k chosen rows of $\mathbf{M}$, the binary $k \times k$ matrix $\mathbf{K}$ be as defined above. Then a vertical cyclc shift of $\mathbf{K}$ will give a $k \times k$ matrix that has an all* '1' *main diagonal.*

Proof. We need to prove that for any k rows chosen from $\mathbf{M}_{\mathrm{u}}$, there exists a vertical cyclic shift of $k \times k$ matrix $\mathbf{K}$ that transforms $\mathbf{K}$ into $k \times k$ matrix $\mathbf{G}$ with all '1' main diagonal. The proof itself is short. We also give an illustrative example.

One might presume that an S row of $\mathbf{M}_{\mathrm{u}}$ contributes to the main diagonal of $\mathbf{G}$ *the first* 1 out of its 1-*run*. (The 1-*run* of row i of $\mathbf{M}_{\mathrm{u}}$ is the sequence of 1's in row i columns $i \bmod k, ..., (i + l - 1) \bmod k$.) Also, one might presume that if a row is an L row, then one of the double rows can contribute one of it subsequent (i.e. not the first) 1's to the main diagonal of $\mathbf{G}$ instead of the L row**.** However, these

assumptions are not entirely correct. We shall show how, by successive steps, either double or single rows can contribute its subsequent 1's to the main diagonal of $\mathbf{G}$ instead of L rows.

All steps $1,...,d$ presented below can always be performed because of the following:

(i) In matrix $\mathbf{M}_{\mathrm{u}}$, the number of L rows equals the number of D rows for any particular choice of rows of $\mathbf{M}$ for matrix $\mathbf{K}$,

(ii) In matrix $\mathbf{M}_{\mathrm{u}}$, the number of L rows is not greater than $\frac{k-1}{2}$,

(iii) The 1-run in each row of $\mathbf{M}$ contains $\frac{k-1}{2}+1=\frac{k+1}{2}$ symbols '1'.

After the presentation of steps, it will be clear that $\mathbf{K}$ can be transformed into $\mathbf{G}$ by returning back to the initial state that is defined below.

In the presentation of the steps, we use a $k\times k$ matrix $\mathbf{V}(.,.)$ composed of some rows of matrix $\mathbf{M}_{\mathrm{u}}$. Step i, $i=1,...,d$ begins with the *start* i matrix $\mathbf{V}(i,0)$. During Step i, $\mathbf{V}(i,0)$ transforms successively into $k\times k$ matrices $\mathbf{V}(i,1),...,\mathbf{V}(i,f_i)$ where $\mathbf{V}(i,f_i)$ is the final matrix in the sequence. At the beginning of Step $i+1$, the final matrix $\mathbf{V}(i,f_i)$ transforms into the start $i+1$ matrix $\mathbf{V}(i+1,0)$.

Now, we show how the start 1 matrix $\mathbf{V}(1,0)$ is composed. Consider the vertical cyclic shifts of rows of $\mathbf{M}_{\mathrm{u}}$ such that D row occupies the top position. A vertical cyclic sift of rows of $\mathbf{M}_{\mathrm{u}}$ is defined as the matrix $\mathbf{M}'_{\mathrm{u}}$; the row m of $\mathbf{M}_{\mathrm{u}}$ becomes the row $(m+j)\,\mathrm{modulo}\,k$ of the matrix $\mathbf{M}'_{\mathrm{u}}$ where j is a given number from $1,...,k$ (see Fig. 3 for illustration with $\mathbf{M}_{\mathrm{u}}$ given by (2.1)).

For the considered example, such shifts are denoted $\mathbf{V}_1$, $\mathbf{V}_2$, $\mathbf{V}_3$ (When describing them, we omit the binary rows for brevity.),

$$\mathbf{V}_1=\begin{bmatrix}D\\L\\S\\L\\S\\D\\D\\S\\S\\L\\S\end{bmatrix},\qquad \mathbf{V}_2=\begin{bmatrix}D\\S\\S\\L\\S\\D\\L\\S\\L\\S\\D\end{bmatrix},\qquad \mathbf{V}_3=\begin{bmatrix}D\\D\\S\\S\\L\\S\\D\\L\\S\\L\\S\end{bmatrix}.$$

In the general case, the shifts are $\mathbf{V}_1,\ldots,\mathbf{V}_d$.

Out of $\mathbf{V}_1,\ldots,\mathbf{V}_d$, we select one matrix that has at least one L below the lowest D. The selected matrix is the start 1 matrix $\mathbf{V}(1,0)$ by definition. Also, we call $\mathbf{V}(1,0)$ the shift from the *initial state* $\mathbf{M}_{\mathrm{u}}$. (Later, we shall return to the initial state by performing the reverse shifts.) The selection of $\mathbf{V}(1,0)$ is not unique. In the considered example, we can select $\mathbf{V}_1$ or $\mathbf{V}_3$ to be $\mathbf{V}(1,0)$. Let us select $\mathbf{V}(1,0)=\mathbf{V}_1$ in our example,

$$\mathbf{V}(1,0)=\begin{bmatrix}
1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & 1 & 1 & \text{row 9 of initial state; } D\\
1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & 1 & \text{row 10 of initial state; } L\\
1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & \text{row 11 of initial state; } S\\
\mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & \text{row 1 of initial state; } L\\
0 & \mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & \text{row 2 of initial state; } S\\
0 & 0 & \mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & \text{row 3 of initial state; } D\\
0 & 0 & 0 & \mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & 0 & \text{row 4 of initial state; } D\\
0 & 0 & 0 & 0 & \mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & \text{row 5 of initial state; } S\\
0 & 0 & 0 & 0 & 0 & \mathbf{1} & 1 & 1 & 1 & 1 & 1 & \text{row 6 of initial state; } S\\
1 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & 1 & 1 & 1 & 1 & \text{row 7 of initial state; } L\\
1 & 1 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & 1 & 1 & 1 & \text{row 8 of initial state; } S
\end{bmatrix} \tag{2.2}$$

In $\mathbf{V}(1,0)$, let us denote the highest D row as D_1, the next D row, which is below D_1, as D_2 and so on, the lowest D row as D_d.

Now, we define the transformation of the final $k\times k$ matrix $\mathbf{V}(i, f_i)$ into the start $i+1$ $k\times k$ matrix $\mathbf{V}(i+1,0)$. The matrix $\mathbf{V}(i, f_i)$ consists of D, S, and L rows. The D rows in $\mathbf{V}(i, f_i)$ are numbered by $1,..., d_i$ from highest to lowest D. (It will be clear later that $d_i = d - i$ but now this is not essential.) We shift $\mathbf{V}(i, f_i)$ vertically such that its D row d_i becomes the top row. If at least one L stays below the D row $d_i - 1$ in the obtained matrix, then the obtained matrix is $\mathbf{V}(i+1,0)$. If there is no L below the D row $d_i - 1$, then we shift the obtained matrix vertically such that its D row $d_i - 1$ becomes the top row and so on. We do it up to the point when the lowest D row in obtained matrix has at least one L below it. The obtained matrix at this point is $\mathbf{V}(i+1,0)$.

Each row in our consideration has one *dedicated* symbol '1'. The important '1' is shown as the bold **1** in a row. The dedicated '1' can change its position in a row when the row participates in covering. In the initial state $\mathbf{M}_u$, the dedicated '1' of each row is the first '1' in its 1-run.

Now, we define the concept of covering that we need in presentation of steps. A non-null row r can cover a null row u if one of non-first 1's of r stays in the column where the dedicated '1' of the null row u stays. A row r, which covers the L row u, can be a single row or one of the double rows. When a row r covers a row u, the result is the following:
(a) row r replaces L row u and the row r is marked by S at this new position,
and additionally,
(b) row r continues occupy its old position but now as the L row, i.e., it changes its symbol S for L at the old position.

Thus, after covering, row r occupies two positions, one is its old position where now the row r is marked by L, and another is the position of the row u where now the row r is marked by S. After covering, the covered row u will not occupy any position and it will not take part in future consideration. The above explanation of covering will be clearer when we consider Step 1 below and illustrate it by example.

Now we define where the dedicated '1' will be in a row r after the row r covers a row u. A row r, which after covering does not change its position and becomes L, continues to have the same dedicated '1' that it had just before the covering. A row r, which after covering occupies the position of covered L row u and becomes S, gets the dedicated '1' at the position where the L row u had its dedicated 1 just before the covering. A covering is possible if the following condition satisfies: the position of dedicated 1 in the L row u is inside of the 1-run of the row r. Because of (i)-(iii), this condition is satisfied in all covering used in the steps described below.

In a trivial case that is not considered below, the chosen rows of $\mathbf{M}$ are the rows $1,..., k$. In this case, all rows of $\mathbf{K}$ are the rows C and $\mathbf{K}$ has all 1 main diagonal.

Step 1. The *final operation* of Step 1 starts when immediately below D_d row, there are the same number of L rows as it was lower (but maybe not immediately below) D_d row in $\mathbf{V}(1,0)$. The final

operation is the covering of L row by one row out of the double D_d row, if the number of L rows staying immediately below D_d row is 1. If the number of L rows staying immediately below D_d row is greater than 1, the final operation is the covering of the two lowest of those L rows by two rows of the double D_d row. After the final covering, we get $\mathbf{V}(1, f_1)$.

Before the final operation, there is the *start operation* in which the S rows that are below D_d row in $\mathbf{V}(1,0)$ cover L rows that are below them. First, in the start operation, the lowest S row in $\mathbf{V}(1,0)$, which has L rows below it, covers the lowest L row. This gives $\mathbf{V}(1,1)$. Then the lowest S row in $\mathbf{V}(1,1)$, which has L rows below it (if such S row exists), covers the lowest L row out of them. This gives $\mathbf{V}(1,2)$ and so on. The start operation ends when immediately below D_d row in the current $\mathbf{V}(1, j)$, there are the same number of L rows as it was lower D_d row in $\mathbf{V}(1,0)$. In our notation, $j = f_1 - 1$ and the next is the final operation.

In our example in the start operation first, the S row 9 of the matrix $\mathbf{V}(1,0)$ (see (2.2)) covers the L row 10. After covering, we get the matrix

$$\mathbf{V}(1,1) = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & 1 & 1 & D \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & 1 & L \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & S \\ \mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & L \\ 0 & \mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & S \\ 0 & 0 & \mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & D \\ 0 & 0 & 0 & \mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & 0 & D \\ 0 & 0 & 0 & 0 & \mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & S \\ 0 & 0 & 0 & 0 & 0 & \mathbf{1} & 1 & 1 & 1 & 1 & 1 & L \\ 0 & 0 & 0 & 0 & 0 & 1 & \mathbf{1} & 1 & 1 & 1 & 1 & S \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & 1 & 1 & 1 & S \end{bmatrix}. \tag{2.3}$$

The first 6 rows of $\mathbf{V}(1,1)$ are the same as the first 6 rows of $\mathbf{V}(1,0)$. Since we are working here with the lower part of the matrix $\mathbf{V}(1,1)$ only, we shall write this part as

$$\mathbf{V}^{\downarrow}(1,1) = \begin{bmatrix} 0 & 0 & 0 & \mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & 0 & D \\ 0 & 0 & 0 & 0 & \mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & S \\ 0 & 0 & 0 & 0 & 0 & \mathbf{1} & 1 & 1 & 1 & 1 & 1 & L \\ 0 & 0 & 0 & 0 & 0 & 1 & \mathbf{1} & 1 & 1 & 1 & 1 & S \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & 1 & 1 & 1 & S \end{bmatrix}. \tag{2.4}$$

Then the S row 8 of the matrix $\mathbf{V}(1,1)$ (see (2.3)) covers the L row 9. After covering, we get $\mathbf{V}(1,2)$ with the lower part

$$\mathbf{V}^{\downarrow}(1,2) = \begin{bmatrix} 0 & 0 & 0 & \mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & 0 & D \\ 0 & 0 & 0 & 0 & \mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & L \\ 0 & 0 & 0 & 0 & 1 & \mathbf{1} & 1 & 1 & 1 & 1 & 0 & S \\ 0 & 0 & 0 & 0 & 0 & 1 & \mathbf{1} & 1 & 1 & 1 & 1 & S \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & 1 & 1 & 1 & S \end{bmatrix}. \tag{2.5}$$

Now, we can begin the final operation. In the final operation, one of rows of the double D row 7 of matrix $\mathbf{V}(1,2)$ coves the L row 8. This gives $\mathbf{V}(1, f_1) = \mathbf{V}(1,3)$, the end of Step 1, and the passage to Step 2,

$$\mathbf{V}(1, f_1) = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & 1 & 1 & D \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & 1 & L \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & S \\ \mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & L \\ 0 & \mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & S \\ 0 & 0 & \mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & D \\ 0 & 0 & 0 & \mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & 0 & S \\ 0 & 0 & 0 & 1 & \mathbf{1} & 1 & 1 & 1 & 1 & 0 & 0 & S \\ 0 & 0 & 0 & 0 & 1 & \mathbf{1} & 1 & 1 & 1 & 1 & 0 & S \\ 0 & 0 & 0 & 0 & 0 & 1 & \mathbf{1} & 1 & 1 & 1 & 1 & S \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & 1 & 1 & 1 & S \end{bmatrix}. \tag{2.6}$$

Step 2. The step begins with the transformation of the final $k \times k$ matrix $\mathbf{V}(i,3)$ into the start 2 $k \times k$ matrix $\mathbf{V}(2,0)$. Then the step repeats the starts and final operations described in Step 1 with the difference that it operates with $\mathbf{V}(2,0)$ instead of $\mathbf{V}(1,0)$ and it works with those rows of $\mathbf{V}(2,0)$ which are the lowest D row and the rows below it.

In our example,

$$\mathbf{V}(2,0) = \begin{bmatrix} 0 & 0 & \mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & D \\ 0 & 0 & 0 & \mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & 0 & S \\ 0 & 0 & 0 & 1 & \mathbf{1} & 1 & 1 & 1 & 1 & 0 & 0 & S \\ 0 & 0 & 0 & 0 & 1 & \mathbf{1} & 1 & 1 & 1 & 1 & 0 & S \\ 0 & 0 & 0 & 0 & 0 & 1 & \mathbf{1} & 1` & 1 & 1 & 1 & S \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & 1 & 1 & 1 & S \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & 1 & 1 & D \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & 1 & L \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & S \\ \mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & L \\ 0 & \mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & S \end{bmatrix}. \tag{2.7}$$

For the start operation, the S row 9 of $\mathbf{V}(2,0)$ (see (2.7)) covers the L row 10 giving

$$\mathbf{V}^{\downarrow}(2,1) = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & 1 & 1 & D \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & 1 & L \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & L \\ \mathbf{1} & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & S \\ 0 & \mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & S \end{bmatrix}. \tag{2.8}$$

Now, this is the turn of the final operation. In the final operation, two of rows of the double D row 7 of the matrix $\mathbf{V}(2,1)$ cover the L rows 8 and 9. This gives $\mathbf{V}(1, f_2) = \mathbf{V}(2,2)$, the end of Step 2, and the passage to Step 3,

$$\mathbf{V}(2,2)=\begin{bmatrix}0&0&\mathbf{1}&1&1&1&1&1&0&0&0&D\\0&0&0&\mathbf{1}&1&1&1&1&1&0&0&S\\0&0&0&1&\mathbf{1}&1&1&1&1&0&0&S\\0&0&0&0&1&\mathbf{1}&1&1&1&1&0&S\\0&0&0&0&0&1&\mathbf{1}&1`&1&1&1&S\\1&1&0&0&0&0&0&\mathbf{1}&1&1&1&S\\1&1&1&0&0&0&0&0&\mathbf{1}&1&1&L\\1&1&1&0&0&0&0&0&1&\mathbf{1}&1&S\\1&1&1&0&0&0&0&0&1&1&\mathbf{1}&S\\\mathbf{1}&1&1&1&1&0&0&0&0&0&1&S\\0&\mathbf{1}&1&1&1&1&1&0&0&0&0&S\end{bmatrix}. \qquad (2.9)$$

The last step is

Step d **.** This step begins with the transformation of the final $k\times k$ matrix $\mathbf{V}(d-1, f_{d-1})$ into the start d $k\times k$ matrix $\mathbf{V}(d,0)$. This is an identical transformation since $\mathbf{V}(d,0)=\mathbf{V}(d-1, f_{d-1})$. Then the step repeats the starts and final operations described in Step 1 with the difference that it operates with $\mathbf{V}(d,0)$ instead of $\mathbf{V}(1,0)$ and it works with all rows of $\mathbf{V}(d,0)$ since the lowest D row in $\mathbf{V}(d,0)$ is its first row. For the final operation, one of rows of only double D row of $\mathbf{V}(d,0)$ covers the only L row and the step ends with the matrix $\mathbf{V}(d, f_d)$. This completes the steps.

In our example, $d=3$ and $\mathbf{V}(d,0)=\mathbf{V}(2,2)$ (see (2.9)). The matrix $\mathbf{V}(3,1)$ appears after the S row 6 of the matrix $\mathbf{V}(3,0)=\mathbf{V}(2,2)$ covers the L row 7. The matrix $\mathbf{V}(3,2)$ appears after the S row 5 of the matrix $\mathbf{V}(3,1)$ covers the L row 6 and so on up to the matrix $\mathbf{V}(3,5)$. The final operation of Step $d=3$ transforms $\mathbf{V}(3,5)$ into $\mathbf{V}(d, f_d)$ with $f_d=6$,

$$\mathbf{V}(d, f_d)=\begin{bmatrix}0&0&\mathbf{1}&1&1&1&1&1&0&0&0&\text{row 3 of initial state; } S\\0&0&1&\mathbf{1}&1&1&1&1&0&0&0&\text{row 4 of initial state; } S\\0&0&0&1&\mathbf{1}&1&1&1&1&0&0&\text{row 5 of initial state; } S\\0&0&0&1&1&\mathbf{1}&1&1&1&0&0&\text{row 6 of initial state; } S\\0&0&0&0&1&1&\mathbf{1}&1`&1&1&0&\text{row 7 of initial state; } S\\0&0&0&0&0&1&1&\mathbf{1}&1&1&1&\text{row 8 of initial state; } S\\1&1&0&0&0&0&0&1&\mathbf{1}&1&1&\text{row 9 of initial state; } S\\1&1&1&0&0&0&0&0&1&\mathbf{1}&1&\text{row 10 of initial state; } S\\1&1&1&0&0&0&0&0&1&1&\mathbf{1}&\text{row 11 of initial state; } S\\\mathbf{1}&1&1&1&1&0&0&0&0&0&1&\text{row 1 of initial state; } S\\0&\mathbf{1}&1&1&1&1&1&0&0&0&0&\text{row 2 of initial state; } S\end{bmatrix}. \qquad (2.10)$$

After fulfilling all steps, we get a $k\times k$ matrix $\mathbf{G}$ with all '1' main diagonal by making the vertical cyclic shift of the matrix $\mathbf{V}(d, f_d)$ to the initial state. The matrix $\mathbf{G}$ gets all '1' main diagonal because for the successive steps, the double and single rows contribute the dedicated 1's to restore those 1's that were lost with the L rows. Since by vertical cyclic shifts that were used everywhere above, we did not change the relative order of rows that was in the initial state matrix $\mathbf{M}_{\text{u}}$, the obtained matrix $\mathbf{G}$ is a vertical cyclic shift of matrix $\mathbf{K}$.

In our example, by making the vertical cyclic shift of matrix $\mathbf{V}(d, f_d)$ to the initial state, we get

$$\mathbf{G}=\begin{bmatrix}
\mathbf{1} & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & \text{row 11 of } \mathbf{K}, \text{ or row 11 of } \mathbf{M} \\
0 & \mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & \text{row 1 of } \mathbf{K}, \text{ or row 13 of } \mathbf{M} \\
0 & 0 & \mathbf{1} & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & \text{row 2 of } \mathbf{K}, \text{ or row 3 of } \mathbf{M} \\
0 & 0 & 1 & \mathbf{1} & 1 & 1 & 1 & 1 & 0 & 0 & 0 & \text{row 3 of } \mathbf{K}, \text{ or row 14 of } \mathbf{M} \\
0 & 0 & 0 & 1 & \mathbf{1} & 1 & 1 & 1` & 1 & 0 & 0 & \text{row 4 of } \mathbf{K}, \text{ or row 4 of } \mathbf{M} \\
0 & 0 & 0 & 1 & 1 & \mathbf{1} & 1 & 1 & 1 & 0 & 0 & \text{row 5 of } \mathbf{K}, \text{ or row 15 of } \mathbf{M} \\
0 & 0 & 0 & 0 & 1 & 1 & \mathbf{1} & 1 & 1 & 1 & 0 & \text{row 6 of } \mathbf{K}, \text{ or row 16 of } \mathbf{M} \\
0 & 0 & 0 & 0 & 0 & 1 & 1 & \mathbf{1} & 1 & 1 & 1 & \text{row 7 of } \mathbf{K}, \text{ or row 6 of } \mathbf{M} \\
1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & \mathbf{1} & 1 & 1 & \text{row 8 of } \mathbf{K}, \text{ or row 8 of } \mathbf{M} \\
\mathbf{1} & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & \mathbf{1} & 1 & \text{row 9 of } \mathbf{K}, \text{ or row 9 of } \mathbf{M} \\
1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & \mathbf{1} & \text{row 10 of } \mathbf{K}, \text{ or row 20 of } \mathbf{M}
\end{bmatrix}. \qquad (2.11)$$

It is clear that matrix $\mathbf{G}$ (2.11) is a vertical cyclic shift of matrix $\mathbf{K}$.
This is the end of the proof.

3. SECOND CONSTRUCTION

In this section by Theorem 2, we present our second construction of matrix $\mathbf{M}$ with the assignment property. We begin with introduction of some definitions and notation that are related to $n \times k$ matrix $\mathbf{M}$ which participates in Theorem 2.

The definition of matrix M for the second construction. The binary $n \times k$ matrix $\mathbf{M}$ with $k=(n/2)+1$, $n=6+4i$, $i \in \{0,1,...\}$ (note that k is even) is the following. The upper left $(k-1)\times(k-1)$ submatrix of $\mathbf{M}$ is such that its row j is the $(j-1)$-positions-rightward horizontal cyclic shift of the row $[1 \ \ldots \ 1 \ \ 0 \ \ldots \ 0]$ which has the number $l=k/2$ of symbols '1' and the number $k-l-1$ of symbols '0'. The last column of the upper $(k-1)\times k$ submatrix of $\mathbf{M}$ is an all '0' column. The lower left $(k-1)\times(k-1)$ submatrix of $\mathbf{M}$ is such that its row j is the $(j-1)$-positions- rightward horizontal cyclic shift of the row $[1 \ \ldots \ 1 \ \ 0 \ \ldots \ 0]$ which has the number $l-1$ of symbols 1 and the number $k-l$ of symbols '0'. The last column of the lower $(k-1)\times k$ submatrix of $\mathbf{M}$ is an all '1' column.

We choose arbitrary k rows of $\mathbf{M}$ and compose of them a $k \times k$ matrix $\mathbf{K}$, which will be described later.

For illustration, we use an example matrix $\mathbf{M}$ with $n=18$, $k=10$, $l=5$. Below is our example of matrix $\mathbf{M}$ where the chosen rows are marked by symbol C and non-chosen rows are marked by NC,

$$\mathbf{M} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & \text{row 1 is NC} \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & \text{row 2 is C} \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & \text{row 3 is NC} \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & \text{row 4 is NC} \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & \text{row 5 is C} \\ 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & \text{row 6 is C} \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & \text{row 7 is NC} \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & \text{row 8 is C} \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & \text{row 9 is NC} \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & \text{row 10 is C} \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & \text{row 11 is C} \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & \text{row 12 is C} \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & \text{row 13 is NC} \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & \text{row 14 is C} \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & \text{row 15 is C} \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & \text{row 16 is NC} \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & \text{row 17 is C} \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & \text{row 18 is NC} \end{bmatrix}.$$

The $k \times k$ matrix $\mathbf{K}$ for the considered example is

$$\mathbf{K} = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & \text{row 10 of } \mathbf{M} \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & \text{row 2 of } \mathbf{M} \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & \text{row 11 of } \mathbf{M} \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & \text{row 12 of } \mathbf{M} \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & \text{row 5 of } \mathbf{M} \\ 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & \text{row 6 of } \mathbf{M} \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & \text{row 15 of } \mathbf{M} \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & \text{row 16 of } \mathbf{M} \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & \text{row 8 of } \mathbf{M} \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & \text{row 17 of } \mathbf{M} \end{bmatrix}.$$

A general definition of matrix $\mathbf{K}$ will be given later after introduction of some notation. The upper $k \times k$ submatrix of $n \times k$ matrix $\mathbf{M}$ is denoted by $\mathbf{M}_u$ and for our example,

$$\mathbf{M}_u \hat{=} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

The submatrix of $\mathbf{M}_u$, which is the matrix $\mathbf{M}_u$ without its last column, is denoted as $\mathbf{M}_u^*$. For our example,

$$\mathbf{M}_u^* \hat{=} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

The lower $(n-k)\times k$ submatrix $\mathbf{M}_l$ of $n\times k$ matrix $\mathbf{M}$ is denoted as $\mathbf{M}_l$. For our example,

$$\mathbf{M}_l \hat{=} \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}.$$

There is a difference in the definitions of $\mathbf{M}_u$ and $\mathbf{M}_l$ between here and in Section 2. The difference is that in Section 2, $\mathbf{M}_u$ and $\mathbf{M}_l$ had the same dimensions and were identical.

The submatrix of $\mathbf{M}_l$, which is $\mathbf{M}_l$ without its last column, is denoted as $\mathbf{M}_l^*$. For our example,

$$\mathbf{M}_l^* \hat{=} \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Each row in the introduced matrices has the 1-run (it may be a cyclic 1-run, see the definition of 1-run and the first 1 of 1-run in the proof of Theorem 1). The row k in $\mathbf{M}_u^*$ has its first 1 in column 1. The row i, $1 \le i \le k-2$ in $\mathbf{M}_l^*$ has its first 1 in column $i+1$.

The two rows of $\mathbf{M}_u^*$ and $\mathbf{M}_l^*$ are called the *associated rows* if they have their first 1's in the same column. The associated rows are rows 1 and k in $\mathbf{M}_u^*$ and the rows i of $\mathbf{M}_u^*$ and $i-1$ of $\mathbf{M}_l^*$, $2 \le i \le k-1$.

We say that among two associated rows, one is the *first associate row* and another is the *second associate row*. The first associate row has its 1-run containing $\frac{k}{2}$ sumbols '1'. The second associate row has its 1-run containing $\frac{k}{2}-1$ sumbols '1'.

The row i of $\mathbf{M}_u^*$ is denoted as C (NC) if the row i of $\mathbf{M}_u$ is C (NC), $1 \le i \le k-2$. Similar, the row i of $\mathbf{M}_l^*$ is denoted as C (NC) if the row i of $\mathbf{M}_l$ is C (NC), $1 \le i \le k-2$.

Now we introduce a column symbolic matrix $\mathbf{V}$. The matrix $\mathbf{V}$ plays the same role here as in the proof of Theorem 1. However, the definition of $\mathbf{V}$ here is different. In the proof of Theorem 1, the matrix $\mathbf{V}$ was a binary $k \times k$ matrix. Now $\mathbf{V}$ is not binary, it is symbolic. The matrix $\mathbf{V}$ has the following *symbolic components*: S called the *single row*, L called the *null row*, D called the *double row*, $S \to L$ called the *replaced row*. There is a little difference in the sense of the same symbols S and D here and in the proof of Theorem 1. The difference will be clear later after component definitions. Here the symbols represent the rows of matrix $\mathbf{M}$. The symbol S represents one row. The sybmol $S \to L$ also represents one row. Each of the symbols D and L represents two rows. In the proof of Theorem 1, the symbols also represented the rows but D (similar to L) represented two identical rows and because of this, the rows that were represented by symbols were shown in $\mathbf{V}$. This allowed to treat $\mathbf{V}$ as a binary matrix. Here we have no such possibility. Nevertheless all time below, we shall associate the symbols and the rows which they represent. Since the rows of $\mathbf{M}$ one-to-one correspond to the rows of $\mathbf{M}_u^*$ and $\mathbf{M}_l^*$, below we shall say, when we need, "a symbol of $\mathbf{V}$ is associated to a row of $\mathbf{M}_u^*$ and $\mathbf{M}_l^*$" instead of "a symbol of $\mathbf{V}$ is associated to a row of $\mathbf{M}$".

Before we explain how the matrix $\mathbf{V}$ is constructed, we put down matrix $\mathbf{V}$ for our example of chosen and non-chosen rows of $\mathbf{M}$,

$$\mathbf{V} = \begin{bmatrix} S \to L & \text{the row 10 replacing the non - chosen row 1} \\ D & \text{the double row representing the rows 2 and 11} \\ S \to L & \text{the row 12 replacing the non - chosen row 3} \\ L & \text{the lost rows 4 and 13} \\ S & \text{the single row 5 since the row 14 is not chosen} \\ D & \text{the double row representing the rows 6 and 15} \\ S \to L & \text{the row 16 replacing the non - chosen row 7} \\ D & \text{the double row representing the rows 8 and 17} \\ L & \text{the lost rows 9 and 18} \end{bmatrix}.$$

The symbol $S \to L$ means that the first associate row is NC *but its second associate row is* C *and the second associate row, symbolically, occupies the position of the first associate row. The words "occupies the position of the first associate row" mean that if the first associate* NC *row is the row i in $\mathbf{M}_u$ (or in $\mathbf{M}_u^*$), the symbol $S \to L$ that represents the second associate row, occupies the position (row) i in matrix $\mathbf{V}$*. In the considered example,

the second associate row 10 occupies in $\mathbf{V}$ the position 1 of the first associate row 1,
the second associate row 12 occupies in $\mathbf{V}$ the position 3 of the first associate row 3,
and
the second associate row 16 occupies in $\mathbf{V}$ the position 7 of the first associate row 7.

The symbol D of the double row means that both first and second associate rows are C *and both occupy in $\mathbf{V}$ the position of the first associate row*. In our example, the D rows are for
the first associate row 2 and its second associate row 11 both occupying the position 2 in $\mathbf{V}$,
the first associate row 6 and its second associate row 15 both occupying the position 6 in $\mathbf{V}$,
and
the first associate row 8 and its second associate row 17 both occupying the position 8 in $\mathbf{V}$.

The symbol S of the single row means that the first associate row is C *and occupies its position in $\mathbf{V}$ but its second associate row is* NC. In our example, the first associate row 5 occupies in $\mathbf{V}$ its position 5 and its second associate row 14 is NC.

The symbol L of the null row means that both first associate row and its second associate row are NC. *The symbol L occupies the position of the first associate row in $\mathbf{V}$*. In our example, the first associate row 9 and its second associate row 18 are NC and the symbol L occupies the position 9 in $\mathbf{V}$.

Generally, the matrix $\mathbf{V}$ was defined above by the *italicized* sentences.

Now we can define the matrix $\mathbf{K}$ in the general case. The number of rows in $\mathbf{K}$ is the sum of the numbers of the symbols $S \to L$, symbols S, and two times number of the symbols D in $\mathbf{V}$. The rows of $\mathbf{K}$ are the rows of $\mathbf{M}$ that are represented by the symbols $S \to L$, S, and D of $\mathbf{V}$. The order of rows in

$\mathbf{K}$ is the same as the order of the symbols $S \to L$, S, and D in $\mathbf{V}$. Since the symbol D represents two rows of $\mathbf{M}$ (one from $\mathbf{M}_u$ and another from $\mathbf{M}_l$), the row from $\mathbf{M}_u$ stays in $\mathbf{K}$ above the row from $\mathbf{M}_l$. The rows of $\mathbf{M}$ that are represented by symbols L do not participate in $\mathbf{K}$.

Theorem 2. *Let the binary $n \times k$ matrix $\mathbf{M}$ and for any k chosen rows of $\mathbf{M}$, the binary $k \times k$ matrix $\mathbf{K}$ be defined as above in this section. Then a vertical cyclc shift of $\mathbf{K}$ is a $k \times k$ matrix that has all* 1 *main diagonal.*

Proof. We need to prove that the $k \times k$ matrix $\mathbf{K}$ of chosen rows of $\mathbf{M}$ can be transformed into a $k \times k$ matrix $\mathbf{G}$ with all 1 main diagonal by changing cyclically the order of its rows.

We shall show that by successive steps that are similar to the steps in the proof of Theorem 1, it is possible to use the non-first 1's of double, single, and replaced rows instead of the first 1's of the null rows to contribute to the main diagonal of $\mathbf{G}$. Unlike in the proof of Theorem 1, in the initial steps, here we shall work with the rows of $\mathbf{M}_u^*$ and $\mathbf{M}_l^*$ and only at the end, we shall extend the considered rows by one more column to get the rows of $\mathbf{M}$.

First, we present the successive steps, then we show that the steps are possible under the theorem conditions, and after these, it will be clear that $\mathbf{K}$ can be transformed into $\mathbf{G}$ by returning to the initial state of matrix $\mathbf{V}$. Before step 1, we do a vertical cyclic shift of initial state matrix $\mathbf{V}$ getting the matrix $\mathbf{V}(1,0)$ that has the top row D and has at least one L row below the lowest D row.

The presentation of steps is similar to the presentation of steps in Theorem 1 proof. For this reason, we point out only the difference in the presentations and do not repeat the step description.

The difference is the following:

When in the proof of Theorem 1, we covered a row by one of the double D rows, we used any row from D since the rows of D were identical. Now, the rows of D are not identical, one is a first associate row and another is a second associate row. Here, we cover a row by the first associate row of D. As it will be shown later, a cover by a first associate row in a step is always possible because of the relations between parameters n, k, and l.

With a row that denoted by symbol $S \to L$, we do the same as with a row S in Theorem 1 proof. In Theorem 1 proof, the symbol $S \to L$ did not participate since there were no difference between two rows with the first 1 at the same column.

If the matrix $\mathbf{V}(1,0)$ has only one component D and the second associate row of D is the last row of $\mathbf{M}_u^*$, we use the second associate row as the last row of $\mathbf{G}$ adding to it the additional column 1 at the end (i.e., adding the additional component 1 at the end of the row). If the matrix $\mathbf{V}(1,0)$ has only one component D but the second associate row of D is not the last row of $\mathbf{M}_u^*$, we use the second associate row for covering, add to it an additional column 1 at the end, and use the extended row as the last row of $\mathbf{G}$.

After the last step, we add the additional final component 0 to each first associate row represented by the final matrix $\mathbf{V}(d, f_d)$ and we add the additional final component '1' to each second associate row. The rows extended by the final component will be the rows of $\mathbf{G}$.

In a trivial case, the matrix $\mathbf{V}(1,0)$ can have only one component D and no components L. This case occurs when the chosen rows of $\mathbf{M}$ are its rows $1,...,k$. It is evident that in this case the chosen rows give $\mathbf{G}$.

All steps here can be fulfilled because of the following properties (A), (B), and (C):

(A) *The maximum number of components L in $\mathbf{V}(1.0)$ is* $\frac{k}{2}-1$. To show this, we note that the matrix $\mathbf{V}(1,0)$ has the maximum number of components L (the number is denoted as $v_{\max}$) if there are k rows C in $\mathbf{M}$ and the rest rows of $\mathbf{M}$, which are not C, give Ls only. Hence, we have $2v_{\max} + k = n$. Since $n = 2k - 2$, we have $v_{\max} = \frac{k}{2} - 1$.

(B) Let us consider a lower part of matrix $\mathbf{V}(i, j)$, i.e. the part, which includes the lowest component D and components which are lower than this D and which contain L s among them. For example, this part can be such as

$$\begin{bmatrix} D \\ S \\ L \\ L \\ S \to L \\ L \end{bmatrix}.$$

Doing a step, first we have to convert this part into

$$\begin{bmatrix} D \\ L \\ L \\ L \\ S \\ S \to L \end{bmatrix}$$

and then to cover two lowest L s by the rows from D. *The rows from D can cover these lowest two L* s. We shall show this not only for the example but in general case.

If D has its first 1 in 1-run in column i and the number of L located under D is v, then the lowest L has its first 1 in 1-run in column $i+v$. Since $v \le \frac{k}{2} - 1$ and the number of 1's in the first associate row row of D is $\frac{k}{2}$, the first associate row can cover the lowest L. It is clear also that the second associate row can cover the row L, which is above the lowest L.

(C) *For a given number of components L of matrix $\mathbf{V}(1,0)$ (here this number is denoted by v),*

$$d_{\min} = v + 1 \tag{3.1}$$

where $d_{\min}$ is the minimum number of the components D of the matrix $\mathbf{V}$. To prove (3.1), we consider the following three special cases of the brother rows 1 and k of $\mathbf{M}_u^*$:

(*j*) The associated rows are C,

(*jj*) The associated rows are NC,

and

(*jjj*) One of the associated rows is C and another is NC.

Case (*j*). We denote by $\mathbf{M}_u^{**}$ the $(k-2)\times(k-1)$ submatrix of $\mathbf{M}_u^*$, which does not contain the rows 1 and k and the column k of the matrix $\mathbf{M}_u^*$. We denote by $\mathbf{M}_l^{**}$ the $(k-2)\times(k-1)$ submatrix of $\mathbf{M}_l^*$, which does not contain the column k of the matrix $\mathbf{M}_l^*$. We shall consider $\mathbf{M}_u^{**}$ and $\mathbf{M}_l^{**}$ instead of $\mathbf{M}_u^*$ and $\mathbf{M}_l^*$ since it is the same what to do, to mark the rows of $\mathbf{M}_u^*$ and $\mathbf{M}_l^*$ by C and NC or to mark the rows of $\mathbf{M}_u^{**}$ and $\mathbf{M}_l^{**}$ by C and NC. The matrix $\mathbf{M}_u^{**}$ contains the first associate rows and the matrix $\mathbf{M}_l^{**}$ contains their second associate rows.

To prove (3.1), first we assume that $v = 0$. In the considered case (*j*), the rows 1 and k of $\mathbf{M}_u^*$ are the rows C already. The case $v = 0$ occurs only if all the rest $k-2$ rows C are the rows of $\mathbf{M}_u^*$ or all $k-2$ rows C are the rows of $\mathbf{M}_l^{**}$. In this case, we have that the number of components D of the matrix $\mathbf{V}(1,0)$ (this number is denoted by d) is 1. Hence, $d_{\min} = 1$, $v = 0$ and (3.1) holds.

Second, we assume that $v = \frac{k}{2} - 1$. If $v = \frac{k}{2} - 1$, the total number of unmarked rows in both $\mathbf{M}_u^{**}$ and $\mathbf{M}_l^{**}$ is $2k - 4 - 2v = k - 2$. Now, the rest $k-2$ rows C must be these $k-2$ unmarked rows. Hence, $d = \frac{k-2}{2} + 1$, $v = \frac{k}{2} - 1$ and (3.1) holds. Note that v can not be greater than $\frac{k}{2} - 1$.

Third, we assume that v is in the interval $0 < v < \frac{k}{2} - 1$. There are $k - 2 - v$ unmarked rows in $\mathbf{M}_u^{**}$ and the same number of the unmarked rows in $\mathbf{M}_l^{**}$. These $k - 2 - v$ rows are not enough to give all the rest $k-2$ rows C. Hence, it must be at least one D in $\mathbf{V}$. We mark by C the first and second associate rows from $\mathbf{M}_u^{**}$ and $\mathbf{M}_l^{**}$ to get D. After this, we need to choose $k-4$ rows C out of $k - 2 - v - 1$ unmarked rows in $\mathbf{M}_u^{**}$ and $k - 2 - v - 1$ unmarked rows in $\mathbf{M}_l^{**}$.

If $k - 4 = k - 2 - v - 1$, then $d_{\min} = 2$. Hence, $d_{\min} = 2$, $v = 1$ and (3.1) holds.

If $k - 4 > k - 2 - v - 1$, it must be at least one more D. Taking this into account, we need to choose $k-6$ rows C out of $k - 2 - v - 1 - 1$ unmarked rows of $\mathbf{M}_u^{**}$ and $k - 2 - v - 1 - 1$ unmarked rows of $\mathbf{M}_l^{**}$. If $k - 6 = k - 2 - v - 1 - 1$, then $d_{\min} = 3$. Hence, $d_{\min} = 3$, $v = 2$ and (3.1) holds.

Generally, if for a given v, we had to choose x of Ds then we need the rest $k - 2 - 2v$ rows C out of $k - 2 - v - x$ unmarked rows of $\mathbf{M}_u^{**}$ and $k - 2 - v - x$ unmarked rows of $\mathbf{M}_l^{**}$. If $k - 2x - 2 = k - 2 - v - x$, all the rest $k - 2 - 2v$ rows C can be the rows of $\mathbf{M}_u^{*}$, or all the rest $k - 2 - 2v$ rows C can be the rows of $\mathbf{M}_l^{**}$. Hence, $d_{\min} = 1 + x$, $v = x$ and (3.1) holds.

Case (*jj*). The brother rows 1 and k of $\mathbf{M}_u^{*}$ give one D. Now, we need to mark by C some k rows out of $k-2$ rows of $\mathbf{M}_u^{**}$ and $k-2$ rows of $\mathbf{M}_l^{**}$. This case looks like Case (*j*) but in Case (*j*), we needed to mark $k-2$ rows by C instead of k.

The $k-2$ rows of $\mathbf{M}_u^{**}$ (the $k-2$ rows of $\mathbf{M}_l^{**}$) are not enough to give all k rows C. Hence it must be at least one D in matrix $\mathbf{V}(1,0)$. After D is chosen, we need to mark $k-2$ rows by C out of $k-3$ rows of $\mathbf{M}_u^{**}$ and $k-3$ rows of $\mathbf{M}_l^{**}$. Again, it must be at least one additional D in matrix $\mathbf{V}(1,0)$. After this last D is chosen, the matrix $\mathbf{V}(1,0)$ has two D. Also after this last D is chosen, we need to mark $k-4$ rows by C out of $k-4$ rows of $\mathbf{M}_u^{**}$ and $k-4$ rows of $\mathbf{M}_l^{**}$.

If $v_1 = 0$ where v_1 is the number of L chosen out of $k-4$ rows of $\mathbf{M}_u^{**}$ and $k-4$ rows of $\mathbf{M}_l^{**}$, we have $d_{\min} = 2$. Hence, $d_{\min} = 2$, $v = 1$ and (3.1) holds.

If $v_1 = \frac{k-4}{2}$, we have $d_{\min} = \frac{k-4}{2} + 2$. Hence, $d_{\min} = \frac{k-4}{2} + 2$, $v = \frac{k-4}{2} + 1$ and (3.1) holds. Note that v_1 can not be greater than $\frac{k-4}{2}$.

Now, we assume that v_1 is in the interval $0 < v_1 < \frac{k-4}{2}$. After we choose x of Ds additionally, we need to choose the rest $k - 4 - 2x$ rows C out of $k - 4 - v_1 - x$ rows of $\mathbf{M}_u^{**}$ and $k - 4 - v_1 - x$ rows of $\mathbf{M}_l^{**}$. The value of x that gives $d_{\min}$ is the solution of the equation $k - 4 - 2x = k - 4 - v_1 - x$. Hence, $d_{\min} = 2 + v_1$, $v = 1 + v_1$ and (3.1) holds.

Case (*jjj*). The brother rows 1 and k of $\mathbf{M}_u^{*}$ give component S or $S \to L$ in vector $\mathbf{V}$. Now, we need to mark by C some $k-1$ rows out of $k-2$ rows of $\mathbf{M}_u^{**}$ and $k-2$ rows of $\mathbf{M}_l^{**}$. One D must be in the matrix $\mathbf{V}(1,0)$ since we need $k-1$ rows C and $k-1$ is greater than $k-2$ rows of each $\mathbf{M}_u^{**}$ and

$\mathbf{M}_l^{**}$. After choosing D, we need to mark by C some $k-3$ rows out of $k-3$ rows of $\mathbf{M}_u^{**}$ and $k-3$ rows of $\mathbf{M}_l^{**}$.

If $v=0$, we have $d_{\min}=1$ and (3.1) holds.

If $v=\frac{k}{2}-2$, we have $d_{\min}=\frac{k}{2}$ and (3.1) holds. Note that v can not be greater than $\frac{k}{2}-2$.

Now, we assume that v is in the interval $0<v<\frac{k}{2}-2$. After we choose x of Ds additionally, we need to choose the rest $k-3-2x$ rows C out of $k-3-v-x$ rows of $\mathbf{M}_u^{**}$ and $k-3-v-x$ rows of $\mathbf{M}_l^{**}$. The value of x that gives $d_{\min}$ is the solution of the equation $k-3-2x=k-3-v-x$. Hence, $d_{\min}=1+x$, $v=x$ and (3.1) holds.

This is the end of the proofs of (3.1) and Theorem 2.

4. NECESSARY CONDITION

Here a necessary condition for the assignment property of matrix $\mathbf{M}$ is presented by the following theorem.

Theorem 3. *In a binary $n\times k$ matrix $\mathbf{M}$, let $l_{\max}$ be the number of symbols* 1 *in a row that has the maximum number of symbols* 1. *Then the condition*

$$(n-k)k<nl_{\max} \tag{4.1}$$

is necessary for the assignment property of $\mathbf{M}$.

Proof. Let us denote by N_i the number of symbols 1 in the column i of $\mathbf{M}$ and by N the number of symbols 1 in the matrix $\mathbf{M}$. We have

$$N=\sum_{i=1}^{k}N_i \tag{4.2}$$

and

$$nl_{\max}\geq N. \tag{4.3}$$

(We note that if all rows of $\mathbf{M}$ have an equal number of symbols 1, then (4.3) holds with equality.)

Suppose that the number of symbols 0 in a column of $\mathbf{M}$ is greater or equal k. Choose the k rows each of which has 0 in this column. By any permutation of rows, the $k\times k$ matrix $\mathbf{K}$ composed of the chosen rows can not be reduced to a $k\times k$ matrix with all 1 main diagonal. For this reason, the necessary condition for the assignment property of $\mathbf{M}$ is that the number of symbols 0 in each column of $\mathbf{M}$ must be less than k, i.e.,

$$n-N_i<k. \tag{4.4}$$

(We note that (4.4) is not the sufficient condition.)

Now from (4.2), (4.3), and (4.4), we have

$$nl_{\max}>\sum_{i=1}^{k}(n-k)=k(n-k), \tag{4.5}$$

completing the proof.

We can apply Theorem 3 to the matrix $\mathbf{M}$ from Theorem 1. The matrix $\mathbf{M}$ has $(n/2)+1$ symbols '1' in its columns and l symbols '1' in its rows. Let us check how the condition (4.1) satisfies for the matrix $\mathbf{M}$ from Theorem 1. If $n=2k$ and $l=(k+1)/2$ as in Theorem 1, we have $n-(nl/k)=k-1$. According to (4.1), $n-(nl/k)$ must be less than k as it is for $\mathbf{M}$ from Theorem 1. This means that (4.1) holds tightly for $\mathbf{M}$ from Theorem 1. Can one make up a more efficient matrix $\mathbf{M}$ in the sense that for given n and k linked by the equation $n=2k$, one can get l which is less than $(k+1)/2$ at least by 1?

No, it is impossible according to (4.1). Thus, the matrix $\mathbf{M}$ from Theorem 1 is optimal in the case of equal number of symbols '1' in the rows.

The same way, it is easy to check that the matrix $\tilde{\mathbf{M}}(n-1,k,l)$, which is the Theorem-1 matrix $\mathbf{M}(n,k,l)$ without one of its row, is still optimal. Thus, Theorem 1 gives a construction of optimal matrices $\mathbf{M}$ not only for $n=6+4i$, but also for $n=6+4i-1$, $i\in\{0,1,...\}$.

Also, we can apply Theorem 3 to the matrix $\mathbf{M}$ from Theorem 2. The matrix $\mathbf{M}$ has $n/2$ symbols 1 in its columns and l symbols 1 in its rows. Now, let us check how the condition (4.1) satisfies for the matrix $\mathbf{M}$ from Theorem 2. If $n=2k-2$ and $l=k/2$ as in Theorem 2, we have $n-(nl/k)=k-1$. This means again that (4.1) holds tightly for $\mathbf{M}$ from Theorem 2 and this $\mathbf{M}$ is optimal in the same sense as the matrix $\mathbf{M}$ from Theorem 1.

Also, the same way, it is easy to check that the matrix $\tilde{\mathbf{M}}(n-1,k,l)$, which is the Theorem-2 matrix $\mathbf{M}(n,k,l)$ without one of its row, is still optimal for $n\geq 12$. Thus, Theorem 2 gives a construction of optimal matrices $\mathbf{M}$ not only for $n=6+4i$, $i\in\{0,1,...\}$ but also for $n=6+4i-1$, $i\in\{2,3,...\}$.

5. CONCLUSION

The paper considered the problem of orthogonal variable spreading Walsh-code assignments. The aim of the paper was to present such assignments that can avoid complicated signaling from BS to users or blind rate and code detection if there are a lot of possible codes out of those BS has to choose a code for transmission. The assignments considered here use a partition of all users into several pools. Each pool can use its own codes that are different for different pools and taken from the general set of available codes. Each user has only a few codes assigned to him.

The considered problem is stated in the paper as a combinatorial problem expressed in terms of the assignment binary matrix $\mathbf{M}$. A solution of the problem is given by construction of $\mathbf{M}$ that has the assignment property defined in the paper. In fact, two such constructions of $\mathbf{M}$ are presented. The matrices $\mathbf{M}$ depend on n, the number of users in a pool; k, the total number of Walsh codes in the pool; and l, the number of Walsh codes assigned to each user. The presented constructions of matrices $\mathbf{M}$ are optimal in the sense that they give the minimal number l for given n and k. The optimality follows from a proven necessary condition for existence of $\mathbf{M}$ with the assignment property.

Additionally, we described the complexity of the presented optimal assignment. The complexity is the number of operations needed in order to find a suitable code channel assignment for the k chosen MSs. It was shown that to determine the code channel assignment, the BS should cyclically shift a defined $k\times k$ matrix $\mathbf{K}$ up to the point when the shifted $\mathbf{K}$ has all '1' main diagonal. This means that the complexity is less than k therefore it is much less than $k!$.

Our constructions of $\mathbf{M}$ with the assignment property are not unique. For example, if $\mathbf{M}$ is a matrix having the assignment property, then the matrix $\mathbf{M}'$ that is obtained by a row and column permutation of $\mathbf{M}$ has the assignment property, as well.

Finally we note that in the case of n,k,l such that $\binom{k}{l}=n$, the tables $[b_{ij}]$, $i=1,...,n,\quad j=1,...,l$ corresponding to the matrices $\mathbf{M}$ from Theorem 1 and 2 have their rows that are the distinguishable unordered combinations of numbers $1,...,k$ taken l at a time. This occurs for $n=10$ in the case of Theorem 1 and for $n=6$ in the case of Theorem 2.

References

[1] A. J. Viterbi, "*CDMA: Principles of spread spectrum communication*," Addison-Wesley, New York, 1995.

[2] F. Adachi, M. Sawahashi, and K. Okawa, "Tree-structured generation of orthogonal spreading codes with different lengths for forward link of DS-CDMA mobile radio," *Electron. Lett*., vol. 33, No. 1, pp.27-28, Jan. 1997.

[3] E. H. Dinan and B.Jabbari, "Spreading codes for direct sequence CDMA and Wideband CDMA Cellular Networks," *IEEE Communications Magazine*, pp. 48-54, Sept., 1998.

[4] T. Minn and K.-Y. Siu, "Dynamic assignment of orthogonal variable-spreading-factor codes in W-CDMA," *IEEE Journal on Selected Areas in Communications*, vol. 18, No. 8, pp. 1429-1440, August 2000.

[5] Y.-C. Tseng, C.-M. Chao, and S.-L. Wu, "Code placement and replacement strategies for wideband CDMA OVSF code tree managemant," *IEEE Global Telecommunications Conference* (*GLOBECOM'01*), vol. 1 pp. 562-566, 2001.

[6] M. Dell'Amico, M. L. Merani, and F. Maffioli, "Efficient algorithms for the assignment of OVSF codes in wideband CDMA," *IEEE International Conference on Communications* (*ICC 2002*), vol. 5 pp. 3055-3060, 2002.

[7] S. Tsai, F. Khaleghi, S.-J. Oh, and V. Vanghi, "Allocation of Walsh codes and quasi-orthogonal functions in cdma2000 forward link," *IEEE Vehicular Technology Conference* (*VTC 2001*), vol. 2 pp. 747-751, 2001.

[8] 1xEV-DV "*Evaluation methodology*," 3GPP2, July 25, 2001.

[9] TIA/EIA 3GPP2 C.S0002-B, "*Physical layer standard for cdma2000 spread spectrum systems, Release B*," Jan. 2001.

[10] TIA/EIA Interim standard, "*Markov service option (MSO) for cdma2000 spread spectrum systems*," April 2001.

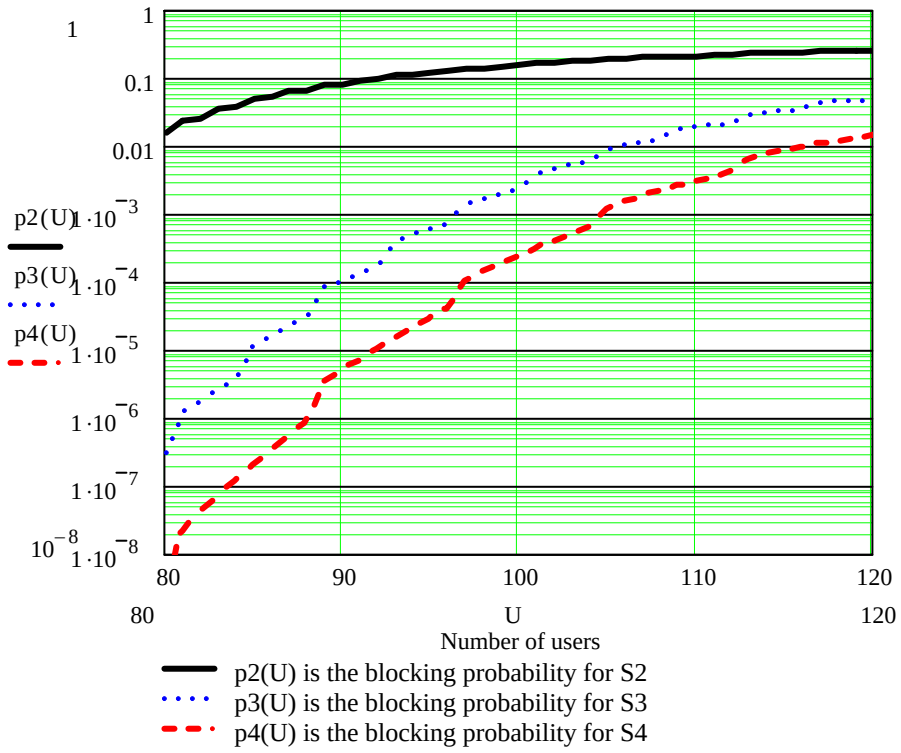

Fig. 1. The blocking probability p*i*(*U*) as a function of the number of uses for assignments S*i*, $i = 2,3,4$.

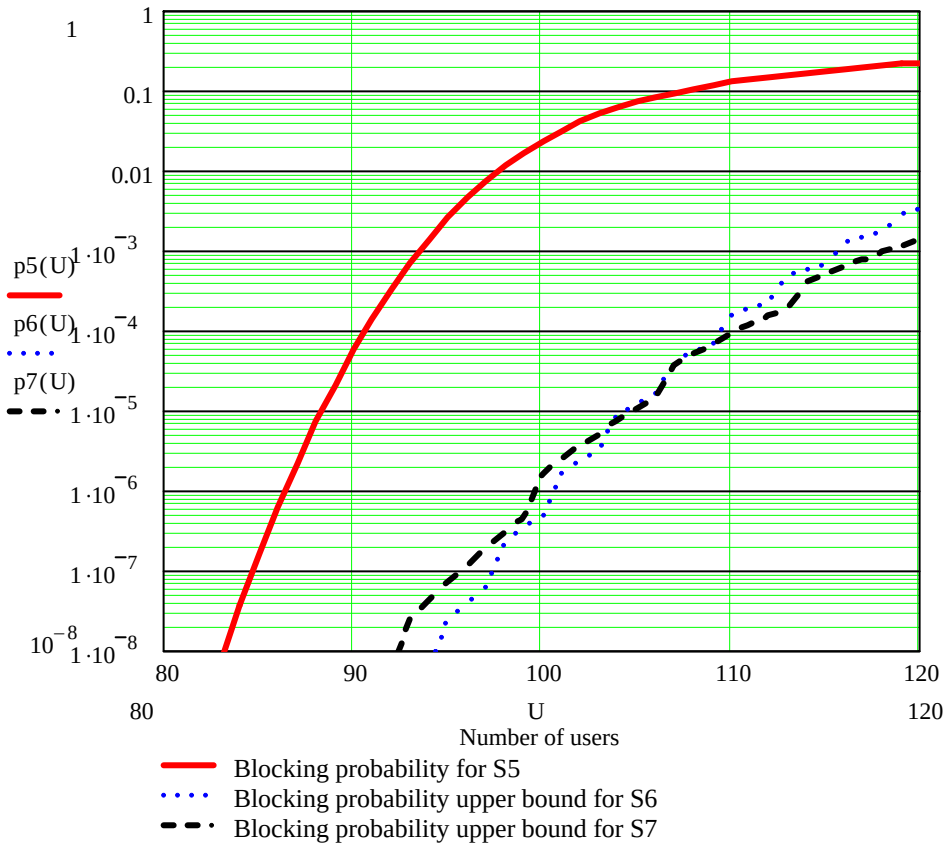

Fig. 2. The blocking probability p*i*(*U*) as a function of the number of uses for assignment S5 and the upper bounds (1.3) and (1.4) for assignments S6 and S7.

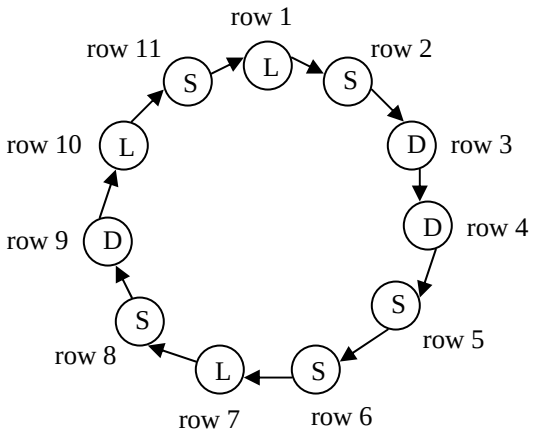

Fig. 3. An illustration of the vertical cyclic shifts of matrix $\mathbf{M}_u$ given by (2.1).