

ON LINEAR SKEWING SCHEMES AND d -ORDERED VECTORS

H.A.G. Wijshoff

J. van Leeuwen

RUU-CS-83-7

March 1983



Rijksuniversiteit Utrecht

Vakgroep informatica

Princetonplein 5
Postbus 80.002
3508 TA Utrecht
Telefoon 030-53 1454
The Netherlands

ON LINEAR SKEWING SCHEMES AND d -ORDERED VECTORS

H.A.G. Wijshoff and J. van Leeuwen

Technical Report RUU-CS-83-7

March 1983

Department of Computer Science
University of Utrecht
P.O. Box 80.002, 3508 TA Utrecht
The Netherlands.

ON LINEAR SKEWING SCHEMES AND d -ORDERED VECTORS

H.A.G. Wijshoff and J. van Leeuwen

Department of Computer Science, University of Utrecht
P.O. Box 80.002, 3508 TA Utrecht, the Netherlands

Abstract Linear skewing schemes were introduced by Kuck et al. in the sixties, to provide a simple class of storage mappings for $N \times N$ matrices for use in vector processors with a large number of memory banks. Conditions on linear skewing schemes that guarantee conflict-free access to rows, columns and/or (anti-) diagonals are usually presented in terms of conditions on so-called d -ordered vectors. We shall argue that these formulations are mathematically imprecise, and revise and extend the existing theory. Several claims are proved to bound the minimum number of memory banks needed for successful linear skewing by e.g. the smallest prime number $\geq N$.

1. Introduction.

In most designs of large computers (such as the ILLIAC IV, see [1]) there is a (large) number of memory banks that can be accessed independently in parallel. The speed and efficiency of these machines for vector computations derives from the fact that in one memory cycle a full vector of M elements can be retrieved, one element from each of the M memory banks provided in the architecture. Kuck [9] (see also [2]) convincingly argued that non-trivial problems arise if an $N \times N$ matrix is to be stored such that all "vectors" of interest (rows, columns, diagonals etc.) can be stored and retrieved in one memory cycle, i.e., as M -vectors "without conflict". Any storage scheme s that maps the elements of an $N \times N$ matrix to a certain number of memory banks and provides for the conflict-free access to various

vectors of interest, is called a skewing scheme. We shall assume that $M \geq N$, and number the memory banks in use from 0 to $M-1$.

Clearly skewing schemes are only practical if they can be described by a small amount of tabular information or by a simple formula. General skewing schemes that fit this aim were called "periodic skewing schemes" by Shapiro [12], and a fundamental analysis of their behaviour was given only recently by the authors [15]. The simplest and (still) most commonly used periodic skewing schemes are the "linear" schemes of Budnik and Kuck [2], defined by

$$s(i, j) = a \cdot i + b \cdot j \pmod{M} \quad 0 \leq i, j \leq N-1$$

(see also Laurie [10]). In the literature several conditions on a , b and M have been formulated such that s is conflict-free on e.g. rows, columns and several diagonals. Very often M is assumed to be a prime number, so \mathbb{Z}_M is a field and various desirable properties of s are ensured almost regardless the values of a and b chosen.

In this paper we shall point at various weaknesses in the existing theory of linear skewing schemes, most notably in relation to the conditions that have been formulated for the conflict-free access of standard vectors from a matrix. In section 2 we discuss the equivalence of linear skewing schemes. In section 3 we review and slightly extend the known results for so-called d -ordered vectors, and point out that the concept is not applied correctly in the theory of linear skewing schemes. In section 4 we formulate precise conditions for a linear skewing scheme to be conflict-free on rows, columns and (circulant) backward and forward diagonals. In section 5 we study the use of linear skewing schemes for retrieving sets of elements by multiple (conflict-free) fetches.

2. Linear skewing schemes

In this section we shall prove several facts, allowing a full range

of values for i and j , i.e., we consider the behaviour of linear skewing schemes s with

$$s(i, j) = a \cdot i + b \cdot j \pmod{M} \quad 0 \leq i, j \leq M-1$$

We do so because in practice s is likely to be used for all $N \times N$ matrices with $N \leq M$. A simple but lacking observation is that s does not necessarily use all memory banks! From the elementary theory of linear congruences the following result is straightforward.

Proposition 2.1 A linear skewing scheme $s(i, j) = a \cdot i + b \cdot j \pmod{M}$ uses all memory banks 0 to $M-1$ if and only if $(a, b, M) = 1$.

Linear skewing schemes with $(a, b, M) = 1$ shall be called proper, and will be the only ones we consider. (If a scheme is not proper, one can factor out (a, b, M) and do the skewing in $M/(a, b, M)$ memory banks.)

By varying the choice of a and b it appears that there are up to $(M-1)^2 = O(M^2)$ different linear skewing schemes. Many do not differ in an essential manner.

Definition. The (linear) skewing schemes s and s' are equivalent if there is a permutation π on $0 \dots M-1$ such that $s = \pi \circ s'$. (Thus, two skewing schemes are equivalent if they are identical after a suitable renaming of the memory banks.)

The equivalence-concept is geared to the linearity of the schemes we consider, as it ensures that bank 0 is always named the same. (Observe that necessarily $\pi(0) = 0$ in the definition.)

Definition. The null-set of a linear skewing scheme $s(i, j) = a \cdot i + b \cdot j$

$(\text{mod } M)$ is the set $L_s = \{(i, j) \mid a \cdot i + b \cdot j \equiv 0 \pmod{M}, 0 \leq i, j < M\}$

Lemma 2.2 Two linear skewing schemes s and s' are equivalent if and only if $L_s = L_{s'}$.

Proof.

Let s and s' be equivalent, $s = \pi \circ s'$. It follows that $\pi(0) = 0$ and (hence) s and s' have identical null-sets. Conversely, let $L_s = L_{s'}$ and consider an arbitrary (i, j) with $s(i, j) = k$ and $s'(i, j) = l$, some k and l . We claim that all elements mapped to bank k by s are mapped to bank l by s' and conversely. It is sufficient to prove this for s . Let $s(i', j') = k$. Because of linearity it follows that $((i' - i) \text{ mod } M, (j' - j) \text{ mod } M) \in L_s$ and (hence) $\in L_{s'}$. Again because of linearity, $((i' - i) \text{ mod } M + i) \text{ mod } M, ((j' - j) \text{ mod } M + j) \text{ mod } M = (i', j')$ is mapped to the same bank as (i, j) under s' , i.e., $s'(i', j') = l$ as was to be shown. This means that s and s' are identical up to the names of the memory banks used. \square

Now consider two proper linear skewing schemes $s(i, j) = a \cdot i + b \cdot j \pmod{M}$ and $s'(i, j) = a' \cdot i + b' \cdot j \pmod{M}$. Define the determinant $\Delta(s, s')$ by

$$\Delta(s, s') = \begin{vmatrix} a & b \\ a' & b' \end{vmatrix}$$

Theorem 2.3 Two proper linear skewing schemes s and s' are equivalent if and only if $\Delta(s, s') \equiv 0 \pmod{M}$.

Proof.

By lemma 2.2 it is necessary and sufficient for s and s' to be equivalent that $a \cdot i + b \cdot j \equiv 0 \pmod{M} \Leftrightarrow a' \cdot i + b' \cdot j \equiv 0 \pmod{M}$, for all $0 \leq i, j < M$. Restricting to $j=0$ ($i=0$) first, we need that $a \cdot i \equiv 0 \pmod{M}$

$\Leftrightarrow a' \cdot i \equiv 0 \pmod{M}$ and $b \cdot j \equiv 0 \pmod{M} \Leftrightarrow b' \cdot j \equiv 0 \pmod{M}$,
and thus obtain that $\gcd(a, M) = \gcd(a', M)$ and $\gcd(b, M) = \gcd(b', M)$.

Writing $u = \gcd(a, M)$ and $v = \gcd(b, M)$ it follows that $a = u \cdot a_1$,
 $b = v \cdot b_1$, $a' = u \cdot a_2$ and $b' = v \cdot b_2$ for suitable integers a_1, a_2, b_1 and
 b_2 that are relatively prime to M . Note that $\gcd(u, v) = 1$ (because
 s is proper and thus $(a, b, M) = 1$) and (hence) $u \cdot v \mid M$. By a_1^{-1}, \dots
we shall denote the inverses mod M of a_1, \dots .

Now $u \cdot a_1 \cdot i + v \cdot b_1 \cdot j \equiv 0 \pmod{M} \Leftrightarrow v \cdot j = -a_1 \cdot b_1^{-1} \cdot u \cdot i + \lambda M$ for
some integer λ , and we get valid solutions i, j whenever $v \mid a_1 \cdot b_1^{-1} \cdot u \cdot i$. But
 v has no common divisor $\neq 1$ with u nor with a_1 and b_1^{-1} (the latter
because any common divisor is also a divisor of M and a_1 and b_1 are
relatively prime to M) and thus we get valid solutions precisely when
 $v \mid i$, which means $i = \mu \cdot v$ for $\mu = 0, 1, \dots, \frac{M}{v} - 1$. The corresponding j -
values follow after substitution. For the equivalence of s and s' to
hold this means that we must have

$$\begin{aligned} u a_2 \cdot i + v \cdot b_2 \cdot j &= u a_2 i + b_2 (-a_1 b_1^{-1} u i + \lambda M) \equiv \\ &\equiv -(u a_1 \cdot b_2 \cdot b_1^{-1} - u a_2) i \pmod{M} \equiv \\ &\equiv -(a \cdot b_2 - a' b_1) b_1^{-1} \cdot \mu v \pmod{M} \equiv \\ &\equiv -(a \cdot b' - a' b) b_1^{-1} \cdot \mu \pmod{M} \equiv \\ &\equiv -\Delta(s, s') b_1^{-1} \mu \pmod{M} \equiv \\ &\equiv 0 \pmod{M} \end{aligned}$$

which is true if and only if $\Delta(s, s') \equiv 0 \pmod{M}$. By a complete-
ly analogous argument one shows that this condition is necessary and
sufficient for $u a_2 i + v b_2 j \equiv 0 \pmod{M} \Rightarrow u a_1 i + v b_1 j \equiv 0 \pmod{M}$.

Conversely, let $\Delta(s, s') = a'b' - a'b \equiv 0 \pmod{M}$. It follows that $\gcd(a, M) \mid a'b$ and hence, by properness, that $\gcd(a, M) \mid a'$. Likewise $\gcd(a', M) \mid a$, and we obtain that necessarily $\gcd(a, M) = \gcd(a', M)$. By the same token we get that $\gcd(b, M) = \gcd(b', M)$. It is easily seen that the preceding argument can now be reversed completely to show that indeed $a \cdot i + b \cdot j \equiv 0 \pmod{M} \Leftrightarrow a' \cdot i + b' \cdot j \equiv 0 \pmod{M}$, i.e., that s and s' are equivalent. \square

Let $s(i, j) = a \cdot i + b \cdot j \pmod{M}$ be called an (a, b) scheme (cf. Lawrie [10]), $a \neq 0$.

Theorem 2.4 Every proper (a, b) scheme is equivalent to a (proper) $(\gcd(a, M), c)$ scheme for some c depending on a, b and M .

Proof

Write $u = \gcd(a, M)$ and let $a = u \cdot a_1$ for a suitable a_1 relatively prime to M . By Theorem 2.3 it suffices to show that there is a solution x to the equation

$$\begin{vmatrix} a & b \\ u & x \end{vmatrix} = ax - bu \equiv 0 \pmod{M}$$

. Take $x = a_1^{-1} b \pmod{M}$ and observe that

$$ax - bu = u a_1 \cdot a_1^{-1} b - bu \equiv 0 \pmod{M}$$

as was to be shown. \square

In case M is prime it follows that every (a, b) scheme is equivalent to a $(1, c)$ scheme. It is easy to see that a $(1, c)$ scheme and a $(1, c')$ scheme are equivalent if and only if $c = c'$, and we get the following result.

Proposition 2.5 For M prime there are precisely $M-1$ essentially different (i.e., non-equivalent) linear skewing schemes.

Proof.

When M is prime, every linear skewing scheme is proper. Equivalence classes are represented by $(1, c)$ schemes, and there clearly are $M-1$ of those (with $c \neq 0$). \square

In general, i.e., when M is not necessarily prime, the equivalence classes are easy to characterize as well. By Theorem 2.4 every proper linear skewing scheme is equivalent to a (d, c) scheme for some $d|M$ and some c . Note that properness now translates to the simple condition that $\gcd(d, c) = 1$.

Lemma 2.6 Every equivalence class of proper linear skewing schemes is uniquely represented by a (d, c) scheme with $0 < d, c < M$, $(d, c) = 1$, $d|M$ and fixed value of $c \pmod{\frac{M}{d}}$.

Proof.

First we show that no (proper) (d, c) scheme can be equivalent to a (proper) (e, f) scheme for $d|M$, $e|M$ and $d \neq e$. For otherwise by Theorem 2.3

$$\begin{vmatrix} d & c \\ e & f \end{vmatrix} = df - ec \equiv 0 \pmod{M}$$

and (as in the proof of Theorem 2.3) it follows from the properness of the schemes that $\gcd(d, M) = \gcd(e, M)$, or $d = e$, a contradiction.

Next we show that a proper (d, c) scheme and a proper (d, c') scheme are equivalent if and only if $c \equiv c' \pmod{\frac{M}{d}}$. For by Theorem 2.3 the necessary and sufficient condition for equivalence is

$$\begin{vmatrix} d & c \\ d & c' \end{vmatrix} = d \cdot (c - c') \equiv 0 \pmod{M}$$

and (hence) that c and c' differ by a multiple of $\frac{M}{d}$. \square

Let $\sigma(M)$ denote the sum of the positive divisors of M . Proposition 2.5 can now be generalized in the following way.

Theorem 2.7 There are at most $\sigma(M) - 2$ essentially different (i.e. non-equivalent) proper linear skewing schemes for a given number of memory banks M .

Proof.

By lemma 2.6 there are at most M/d non-equivalent $(d, *)$ schemes for every d with $1 \leq d < M$, $d | M$. In fact for $d=1$ there are only $M-1$ schemes. Thus the total number of non-equivalent proper schemes is bounded by

$$\sum_{\substack{d|M \\ d \neq M}} M/d - 1 = \sum_{d|M} M/d - 2 = \sum_{d|M} d - 2 = \sigma(M) - 2$$

\square

As $\sigma(M) = O(M \log \log M)$, with an average of $O(M)$ ([5.1]), theorem 2.7 proves the earlier claim that the number of non-equivalent proper linear skewing schemes really is substantially less than M^2 for $M \rightarrow \infty$.

3. d-Ordered vectors.

The subject of d -ordered vectors naturally arises in relation to linear skewing schemes $s(i, j) = a \cdot i + b \cdot j \pmod{M}$. For if we consider a matrix row (i fixed), then we see that its elements are "ordered" into successive memory banks that are b apart mod M . Like-

wise, the elements of any matrix column (j fixed) appear "ordered" in banks that are a apart mod M .

Definition. A d -ordered k -vector is a vector of k elements whose i^{th} logical element ($0 \leq i < k$) is stored in memory bank $c + d \cdot i \pmod{M}$, for some arbitrary constant c .

d -Ordered vectors are first discussed in Budnik and Kuck [2] and in Lawrie [10]. In general one would like to have storage schemes such that all vectors of interest can be retrieved as d -ordered k -vectors without conflict, for some suitably chosen d and k . Essentially only the following result is known (stated in Lawrie [10] only as a sufficient condition).

Theorem 3.1. A d -ordered k -vector can be accessed conflict-free if and only if $M \geq k \cdot \gcd(d, M)$.

Proof

Consider any d -ordered k -vector and assume it can be accessed conflict-free. It means that for all $0 \leq i_1, i_2 < k$, $i_1 \neq i_2$ we have

$$c + d \cdot i_1 \not\equiv c + d \cdot i_2 \pmod{M} \Rightarrow$$

$$\Rightarrow d \cdot i \not\equiv 0 \pmod{M} \text{ for } 0 < i < k \Rightarrow$$

$$\Rightarrow \frac{M}{\gcd(d, M)} \text{ is not among } 0 \dots k-1 \Rightarrow$$

$$\Rightarrow \frac{M}{\gcd(d, M)} \geq k \Rightarrow M \geq k \cdot \gcd(d, M)$$

Conversely, assume that $M \geq k \cdot \gcd(d, M)$. Observe that all implications \Rightarrow above can be reversed. Thus no conflict will arise in access-

any d -ordered k -vector. \square

Theorem 3.1 provided a uniform bound of $\lceil \frac{M}{\text{gcd}(d, M)} \rceil$ on the maximum size of "conflict-free" d -ordered vectors. It is advantageous to have $\text{gcd}(d, M) = 1$, which is guaranteed e.g. when M is prime. Larger d -ordered vectors will have to be retrieved in r "parallel fetches", some $r > 1$. Note that we make no assumption about the way an ordered vector is actually split up for this. Theorem 3.1 can now be generalised in the following way.

Theorem 3.2 A d -ordered k -vector can be accessed in r conflict-free fetch operations if and only if $M \geq (\lfloor \frac{k-1}{r} \rfloor + 1) \text{gcd}(d, M)$.

Proof

(We may assume that $r > 1$.) From the proof of Theorem 3.1 it follows in particular that the largest set of elements from a d -ordered vector that can be accessed conflict-free has size $\lceil \frac{M}{\text{gcd}(d, M)} \rceil$ and that this amounts to taking exactly one datum from each memory bank holding elements from the d -ordered vector. (Thus every maximum set can in fact be retrieved as a d -ordered subvector!) This implies that we may assume without loss of generality that d -ordered k -vectors are retrieved in full batches of $\lceil \frac{M}{\text{gcd}(d, M)} \rceil$ elements and a "remainder".

Now assume that a d -ordered k -vector can be accessed in r conflict-free fetches. It follows that $k \leq r \cdot \lceil \frac{M}{\text{gcd}(d, M)} \rceil$. Write $k = \alpha \cdot r + \beta$ for $0 \leq \beta < r$ and distinguish two cases. If $r \mid k$ (hence $\beta = 0$) then $\alpha = \frac{k}{r} = \lfloor \frac{k-1}{r} \rfloor + 1$ and $M \geq \alpha \cdot \text{gcd}(d, M)$. If $r \nmid k$ then we obtain $\alpha = \lfloor \frac{k-1}{r} \rfloor$ and

$$\alpha \cdot r + \beta \leq r \cdot \lceil \frac{M}{\text{gcd}(d, M)} \rceil - 1 \Rightarrow$$

$$\Rightarrow \alpha + \frac{\beta+1}{r} \leq \lceil \frac{M}{\text{gcd}(d, M)} \rceil \Rightarrow$$

$$\Rightarrow \alpha + 1 \leq \frac{M}{\gcd(d, M)}$$

whence $M \geq (\alpha + 1) \gcd(d, M)$. This proves the bound on M for both cases.

To prove the converse, we rephrase the earlier interpretation of Theorem 3.2 once again and observe that a d -ordered k -vector can be retrieved in r conflict-free accesses if and only if no memory bank holds more than r elements of the d -ordered vector. If $M \geq (\lfloor \frac{k-1}{r} \rfloor + 1) \gcd(d, M)$ then this is exactly what happens because it forces a bound on k (basically $k \leq r \cdot \frac{M}{\gcd(d, M)}$) so the elements of the vector that are stored in banks in the order

$$\underbrace{c \pmod{M}, c+d \pmod{M}, \dots, c + \left(\frac{M}{\gcd(d, M)} - 1\right) d \pmod{M}}_{\text{first batch}}, \underbrace{c \pmod{M}, \dots}_{\text{second batch}}, \dots$$

are not "wrapped" over the same bank more than r times. \square

Corollary 3.3 A d -ordered k -vector can be accessed in precisely $\lfloor \frac{(k-1) \cdot \gcd(d, M)}{M} \rfloor + 1$ conflict-free fetch operations (where each fetch operation retrieves a d -ordered subvector), and this is best possible.

Proof.

By the analysis in Theorem 3.2 the (optimal) method for accessing a d -ordered k -vector is in full batches of $\frac{M}{\gcd(d, M)}$ elements and a "remainder". Writing $k = \alpha \cdot \frac{M}{\gcd(d, M)} + \beta$ for $\alpha \geq 0$ and $0 \leq \beta < \frac{M}{\gcd(d, M)}$ it follows that α fetches are necessary and sufficient for $\beta = 0$, and $\alpha + 1$ fetches are necessary and sufficient for $\beta > 0$. One easily verifies that the precise formula for α and $\alpha + 1$, respectively, is as given above. \square

Finally note that two d -ordered vectors are either stored in the same set of $\frac{M}{\gcd(d, M)}$ memory banks or in fully disjoint ones, depen-

ding on the value of c (the "offset") mod $\gcd(d, M)$. Thus up to $\gcd(d, M)$ different d -ordered vectors can be retrieved conflict-free in one cycle.

The results for d -ordered vectors as presented, simple as they are, have been frequently used to estimate the smallest number of memory banks (M) needed for conflict-free access of certain vectors of interest. Two fallacies can be recognized: (i) "vectors of interest" cannot always be molded into d -ordered vectors for a proper value of d (although the literature is at least suggestive at this point), and (ii) theorem 3.1 does not imply that M is necessarily a multiple of k (although this is often concluded).

4. Conflict-free access to rows, columns and (anti-) diagonals.

We shall now consider storing an $N \times N$ matrix into M memory banks, some M , such that a choice of "vectors of interest" can be accessed conflict-free. We shall make use of a suitable (proper) linear skewing scheme s :

$$s(i, j) = a \cdot i + b \cdot j \pmod{M} \quad 0 \leq i, j < N$$

where a and b (like M) are yet to be determined. The following observations are immediate (see Lawrie [10]):

- each row is a b -ordered N -vector,
- each column is an a -ordered N -vector,
- the main diagonal is an $(a+b)$ -ordered N -vector,
- the main anti-diagonal is an $(a-b)$ -ordered N -vector,

the latter provided that $a \neq b$. Note that all other (anti-) diagonals are likewise "ordered" k -vectors for suitable $k < N$. We conclude:

Proposition 4.1 A linear skewing scheme provides conflict-free access to all [non-circulant] (anti-) diagonals of a matrix if and only if it provides conflict-free access to the main (anti-) diagonal.

Proof

Directly from linearity, or from Theorem 3.1. \square

Applying Theorem 3.1 we obtain simple conditions on a , b and M for conflict-free access to rows, columns and diagonals (cf. Lawrie [10]):

$$(*) \begin{cases} M \geq N \cdot \text{gcd}(b, M) & \text{for rows,} \\ M \geq N \cdot \text{gcd}(a, M) & \text{for columns,} \\ M \geq N \cdot \text{gcd}(a+b, M) & \text{for (all) diagonals,} \\ M \geq N \cdot \text{gcd}(a-b, M) & (a \neq b) \text{ for (all) anti-diagonals.} \end{cases}$$

Proposition 4.2 In order to have conflict-free access to rows, columns, diagonals and anti-diagonals using a linear skewing scheme, it is sufficient to choose M as the smallest prime number $\geq N$ ($N > 3$).

Proof.

This is immediate from (*) taking e.g. $a=1$ and $b=2$. \square

The choice of M implied by proposition 4.2 will normally not be the smallest possible. Discussions in e.g. Lawrie [10] show that there are "non-prime" cases where $M=N$. We show:

Theorem 4.3 In order to have conflict-free access to rows, columns, diagonals and anti-diagonals using a linear skewing scheme, the smallest number of memory banks required is

$$M = \begin{cases} N & \text{if } 2 \nmid N \text{ and } 3 \nmid N, \\ N+1 & \text{if } 2 \mid N \text{ and } N \equiv 0, 1 \pmod{3}, \\ N+2 & \text{if } 2 \nmid N \text{ and } 3 \mid N, \\ N+3 & \text{if } 2 \mid N \text{ and } N \equiv 2 \pmod{3}. \end{cases}$$

Proof

First let $2 \nmid N$. If also $3 \nmid N$ then take $a=1$ and $b=2$ and observe that all gcd's in (*) are 1 when $M=N$. Thus all inequalities are satisfied for this smallest choice of M . If $3 \mid N$ then necessarily $N=6v+3$, some v . Now note that M cannot be equal to N . For otherwise at least one of the gcd's in (*) would be ≥ 3 as with any choice of a and b at least one of $a, b, a+b$ and $a-b$ will have a factor 3. The next best choice $M=N+1$ (= even) will not do either because with any choice of a and b at least two of $a, b, a+b$ and $a-b$ will have a factor 2. Observe that $M=N+2$ ($=6v+5$) is neither even nor divisible by 3 and (thus) taking $a=1$ and $b=2$ satisfies (*) and makes $M=N+2$ the smallest number of memory banks to use in this case.

Next let $2 \mid N$. If $N \equiv 2 \pmod{3}$ then necessarily $N=6v+2$, some v . By the same argument as above we must choose M equal to the first number $\geq N$ that is neither even nor divisible by 3 in order that valid a and b can be chosen, hence $M=N+3=6v+5$ and we can again take $a=1$ and $b=2$. If $N \equiv 0, 1 \pmod{3}$ then N is of the form $6v$ or $6v+4$, some v . In both cases $M=N$ is not possible for satisfying (*) but $M=N+1$ is, again with $a=1$ and $b=2$. \square

Corollary 4.4 If it is possible to have conflict-free access at all to rows, columns, diagonals and anti-diagonals using a linear skewing scheme, then it is possible to achieve this using the scheme $i+2j \pmod{M}$.

Proof

From the preceding analysis it follows that for the required conflict-free accesses M must necessarily be non-divisible by 2 or 3 (otherwise a contradiction occurs for any choice of a and b), or it is much larger than the minima given in theorem 4.3 (but then the smaller number we need is available too). From the proof we conclude that one can always use $i+2j \pmod{M}$ for a skewing scheme,

with a suitable choice of M . \square

Theorem 4.3 shows that only a few extra banks are really needed over the minimum of N to access a large collection of "vectors of interest" conflict-free, and corollary 4.4 states that $(1,2)$ -schemes are virtually universal for it! Clearly other bounds will result if the set of vectors of interest is changed. We shall study the case of conflict-free access to rows, columns and full circulant diagonals and anti-diagonals. Besides being of interest as an exact extension of the theory above, it will show the difficulties when the vectors of interest are no longer "d-ordered". (Note that a linear skewing scheme makes a "jump" when applied to circulant off-diagonals.)

Historically the case $M=N$ has received considerable attention, not just within the context of vector processing. In the statistical analysis of experiments any assignment of bank-numbers 1 to N to the cells of an $N \times N$ matrix such that (in our terminology) conflict-free access is provided to rows, columns and all circulant diagonals and anti-diagonals is called a Knut Vik design (after Vik [14]). In 1973 Hedayat and Federer [7] showed that no Knut Vik designs exist for N even, and in 1975 Hedayat [6] completed the analysis and proved that Knut Vik designs of order N exist if and only if N is not divisible by 2 and 3. In terms of (general) skewing schemes the result was obtained independently by Shapiro [12] (see also [13]). In fact, Hedayat [6] relies on observations of Euler [3] to show (in our terminology) essentially the following:

Theorem 4.5 Let $2 \nmid N$ and $3 \nmid N$. Then there exists a (proper) linear skewing scheme using $M=N$ banks that provides conflict-free access to rows, columns and all circulant diagonals and anti-diagonals.

For all other values of N we will need a larger number of memory banks

M to have the same effect. Consider the following collection of conditions for a linear skewing scheme:

$$(**) \begin{cases} ax \not\equiv 0 & \text{for all } x \in [1..N-1], \\ bx \not\equiv 0 & \text{" " " "}, \\ (b+a)x \not\equiv 0 & \text{" " " "}, \\ (b-a)x \not\equiv 0 & \text{" " " "}, \\ (b+a)x \not\equiv bN & \text{" " " "}, \\ (b-a)x \not\equiv bN & \text{" " " "}, \end{cases}$$

where equivalences \equiv are taken modulo M .

Lemma 4.6 Conditions $(**)$ are necessary and sufficient for the existence of a linear skewing scheme using M banks that provides conflict-free access to rows, columns and all circulant diagonals and anti-diagonals.

Proof.

We shall accumulate the necessary and sufficient conditions for each of the sets of vectors

(i) Rows

For the elements of the i^{th} row to be stored in different banks it is required that $ai + bj_1 \not\equiv ai + bj_2 \pmod{M}$, i.e., that $b(j_1 - j_2) \not\equiv 0$ for all $0 \leq j_1, j_2 < N$ with $j_1 \neq j_2$. This translates into the second condition of $(**)$.

(ii) Columns.

By a completely analogous argument this leads to the first condition of $(**)$.

(iii) Circulant diagonals.

The k^{th} circulant diagonal ($0 \leq k < N$) consists of the "vector" of cells $(i, (i+k) \bmod N)$ for $0 \leq i < N$. The i^{th} element is thus mapped to bank $(a+b)i + bk \pmod{M}$ if $0 \leq i < N-1-k$, and to bank

Using lemma 4.6 one can prove several effective upper bounds on the required number of memory banks M for conflict-free linear skewing of the vectors we are considering. We shall consider the case that only conflict-free access to rows, columns and circulant [forward] diagonals is desired separately. From the analysis above is easily derived that necessary and sufficient conditions for this case are:

$$(***) \begin{cases} ax \neq 0 & \text{for all } x \in [1..N-1], \\ bx \neq 0 & \text{" " " "}, \\ (b+a)x \neq 0 & \text{" " " "}, \\ (b+a)x \neq bN & \text{" " " "}. \end{cases}$$

Theorem 4.7 Let M be the smallest prime number $\geq N$ with $M \neq N+1$. Then there exists a linear skewing scheme using M memory banks that provides conflict-free access to rows, columns and all circulant (forward) diagonals. ($N \neq 2$.)

Proof.

If N is prime and > 2 , then the conditions of (***) are easily seen to be satisfied with $M=N$ and $a=b=1$. If N is not prime, then certainly $M > N+1$ and we can reason as follows. By theorem 2.4 we may assume without loss of generality that $a=1$. Conditions (***) thus translate to the requirement that an integer b exists with $1 \leq b \leq M-2$ such that $N^{-1}(1+b^{-1})$ has no inverse in the set $1..N-1 \pmod{M}$. But if we let b range over all $M-2$ distinct values 1 to $M-2$, then we get exactly $M-2$ distinct values for the inverses of $N^{-1}(1+b^{-1})$. As $M > N+1$ (hence $M-2 > N-1$.) it now follows by a simple application of the pigeon hole principle that at least one of these inverses must lie outside of the forbidden range $1..N-1$, and the corresponding b will do for our purposes. (Observe that indeed no b can exist if $M=N+1$, M prime.) \square

Another interesting conclusion can be drawn from this.

Corollary 4.8 Any linear skewing scheme that provides conflict-free access to rows, columns and all circulant (forward) diagonals using a minimum number of memory banks M is equivalent to a $(1, c)$ scheme, for some c with $\gcd(c, M) = 1$ ($N \neq 2$).

Proof.

By Bertrand's postulate (cf. [5]) the smallest prime $\gg x$ is $\leq 2x-2$. Thus it follows from theorem 4.7 that the smallest number of memory banks M needed for the desired type of linear skewing satisfies $M < 2N$, and from (***) we see that the choice of a is constrained to $\gcd(a, M) = 1$. By theorem 2.4 we conclude that any linear skewing scheme with this property is equivalent to a $(1, c)$ scheme. The condition on c follows again from (***) \square

Theorem 4.9 Let M be the smallest prime number $> 2N+1$. Then there exists a linear skewing scheme using M memory banks that provides conflict-free access to rows, columns and all circulant diagonals and anti-diagonals.

Proof

By theorem 2.4 we may assume without loss of generality that $a = 1$. Conditions (***) now translate into the requirement that there exists an integer b with $2 \leq b \leq M-2$ such that both $N^{-1}(1+b^{-1})$ and $N^{-1}(1-b^{-1})$ have inverses outside of the set $1 \dots N-1 \pmod{M}$. Consider the collection of pairs $\{(N^{-1}(1+b^{-1}))^{-1}, (N^{-1}(1-b^{-1}))^{-1}\}$ for b from 2 to $M-2$. This gives precisely $M-3$ distinct pairs, and the values that appear in the first (or second) coordinate as inverses are necessarily all distinct. Thus striking out all pairs that have a "forbidden" first or second coordinate eliminates at most $2(N-1)$ pairs and (hence) leaves at least $M-3 - 2(N-1) = M - (2N+1) \geq 1$ pairs with both coordinates outside of the range $1 \dots N-1$. Choosing for b the integer corresponding to one of these pairs will do for our purposes. \square

In case $3 \nmid N$ the bound on M given in Theorem 4.9 can be improved to the smallest prime number $\geq 2N+1$ (but also recall Theorem 4.5). To show this we only need to consider the situation that $2N+1$ is prime and hence $M = 2N+1$. Necessarily $N = 3v+2$, some v . It can be verified that $2^{-1} \equiv N+1$, $N^{-1} \equiv 2N-1$ and $(2N)^{-1} \equiv 2N$. Consider the linear skewing scheme determined by $a=1$ and $b = M-2 = 2N-1$. Clearly $b+a \neq 0$ and $b-a \neq 0$ (provided $N > 1$), and it follows that we only need to verify the last two conditions of (**).

(i) Suppose $2N \cdot x \equiv (2N-1)N$. Then $x \equiv 2^{-1} (2N-1) \equiv (N+1)(2N-1) \equiv 2N$, and thus $(b+a)x \equiv bN$ has no solution in the range $1 \dots N-1$.

(ii) Suppose $(2N-2)x \equiv (2N-1)N$, or $3x \equiv 2N$. This means that we seek a solution $x \in [1 \dots 2N]$ to the equation $3x = 2N + \lambda(2N+1) = (2+2\lambda)3v + 4 + 5\lambda$. For $\lambda=1$ we get as (necessarily unique) solution $x = 4v+3 = \frac{4}{3}N + \frac{1}{3}$, which is outside the range $1 \dots N-1$. Thus $(b-a)x \equiv bN$ has no solution in the latter range.

It follows that the conditions (**) are satisfied for this choice of a and b . The result is only of interest if a prime number of memory banks is provided, and certainly does not give the best possible bound. For if $3 \nmid N$ then it is easily seen that (**) can be satisfied with $a=1$, $b=2$ and $M=2N$ (i.e., $2N$ memory banks are sufficient).

5. Conflict-free access through multiple fetches.

In section 3 we anticipated that "vectors of interest" can be retrieved by performing (at most) r conflict-free fetches from the given set of M memory banks, some $r \geq 1$. This certainly applies to the case of retrieving circulant diagonals (cf. section 4) which, after all, can be obtained by at most 2 conflict-free fetches using a skewing scheme that is valid for non-circulant (i.e., ordinary) diagonals. Using Theorem 4.3 it follows that no more than $N+3$ memory banks are needed to skew an $N \times N$ matrix and have conflict-free access to rows, columns and all circulant diagonals, if only we allow up to 2 re-

trivial operations per "vector". (Compare this to theorem 4.7.) In this section we shall examine the effect of multiple fetches more closely.

We shall first consider accessing rows, columns and (ordinary) diagonals of an $N \times N$ matrix using a linear skewing scheme and r -fold fetching. Consider the following conditions.

$$(***) \begin{cases} M \geq (\lfloor \frac{N-1}{r} \rfloor + 1) \text{gcd}(a, M) \\ M \geq (\lfloor \frac{N-1}{r} \rfloor + 1) \text{gcd}(b, M) \\ M \geq (\lfloor \frac{N-1}{r} \rfloor + 1) \text{gcd}(b+a, M) \\ M \geq (\lfloor \frac{N-1}{r} \rfloor + 1) \text{gcd}(b-a, M) \\ (a < b) \end{cases}$$

Lemma 5.1 Conditions (***) are necessary and sufficient for the existence of a linear skewing scheme using M banks that provides conflict-free access to rows, columns, diagonals and anti-diagonals in at most r fetches per vector.

Proof.

The comments at the beginning of section 4 apply to characterize the "vectors of interest" as d -ordered vectors of size (at most) N , for the proper values of d . In theorem 3.2 was shown that such vectors can be retrieved in at most r conflict-free fetches if and only if one has $M \geq (\lfloor \frac{N-1}{r} \rfloor + 1) \text{gcd}(d, M)$. Substituting the pertinent values of d here leads to the conditions (***) as claimed. \square

Theorem 5.2 In order that there exists a linear skewing scheme that provides conflict-free access to rows, columns, diagonals and anti-diagonals in at most r fetches per vector the number of memory banks M required need be no larger than $\lfloor \frac{N-1}{r} \rfloor + 4$.

Proof.

The analysis in theorem 4.3 (of the very similar conditions in (*))

) shows that the smallest M needed to satisfy (***) need be no larger than $(\lfloor \frac{N-1}{r} \rfloor + 1) + 3 = \lfloor \frac{N-1}{r} \rfloor + 4$. \square

Actually the analysis of Theorem 4.3 shows that the smallest value for M needed in Theorem 5.2 is precisely equal to "the smallest number $\geq \lfloor \frac{N-1}{r} \rfloor + 1$ that is not divisible by 2 and 3". This leads to an interesting observation about the trade-off between the number of memory banks M and r . (We simply take $r=2$.)

Proposition 5.3 Consider the existence of linear skewing schemes for conflict-free access to rows, columns, diagonals and anti-diagonals.

(i) If $2 \mid N$ but $3 \nmid N$ and $4 \nmid N$, then one can "skew" an $N \times N$ matrix in $\frac{N}{2}$ banks and retrieve rows etc. in at most 2 fetches per vector but it is impossible to skew it in N banks and retrieve rows etc. in one single fetch.

(ii) If $2 \nmid N$ and $3 \nmid N$ and $N \not\equiv 1 \pmod{12}$, then one can skew an $N \times N$ matrix in N banks and retrieve rows etc. in one single fetch but it is impossible to skew it in $\lceil \frac{N}{2} \rceil$ banks and retrieve rows etc. in 2 fetches per vector.

Proof.

(i) If $2 \mid N$ but $3 \nmid N$ and $4 \nmid N$, then $\lfloor \frac{N-1}{2} \rfloor + 1 = \frac{N}{2}$ and $2 \nmid \frac{N}{2}$ and $3 \nmid \frac{N}{2}$. Hence $\frac{N}{2}$ memory banks suffice for retrieving rows etc. in at most 2 fetches. On the other hand Theorem 4.3 shows that at least $N+1$ banks are needed if we want to retrieve rows etc. in a single fetch.

(ii) If $2 \nmid N$ and $3 \nmid N$ then N banks suffice for conflict-free access to rows etc. in one fetch (Theorem 4.3). If in addition $N \not\equiv 1 \pmod{12}$ and hence N is of the form $12v+5$, $12v+7$ or $12v+11$ then the smallest $M \geq \lfloor \frac{N-1}{2} \rfloor + 1$ not divisible by 2 and 3 is equal to $6v+5$, $6v+5$ and $6v+7$ respectively. Thus the smallest M required to be able to retrieve rows etc. in 2 fetches is $> \lceil \frac{N}{2} \rceil$ in each case. \square

It is of some interest to consider the effect of multiple fetches if conflict-free access is required to more general "templates" in a matrix than only rows, columns and diagonals. A template is defined as any (finite) configuration of cells that can be put anywhere we like within the "square" array of a matrix (cf. Shapiro [12]). We shall prove that there are linear schemes for skewing an $N \times N$ matrix in $N + O(1)$ memory banks such that every connected template of data can be retrieved conflict-free in $O(\sqrt{N})$ fetches. We shall first consider the case of rookwise connected templates, i.e., templates in which every two cells are connected by a path of consecutive cells that always border by a full side.

Choose M as a perfect square, and consider the linear skewing scheme s defined by

$$s(i, j) = i - \sqrt{M} \cdot j \pmod{M}$$

for $0 \leq i, j < N$. Observe that $s(i, j + \sqrt{M}) = s(i, j)$ and also that $s(i + \sqrt{M}, j + 1) = s(i, j)$. Further it is clear that s maps the cells of any $\sqrt{M} \times \sqrt{M}$ sub-array (viewed as a template) to different banks, i.e., s is conflict-free on $\sqrt{M} \times \sqrt{M}$ blocks. In effect it means that an $N \times N$ matrix is stored by splitting (covering) it into approximately N^2/M sub-arrays of size $\sqrt{M} \times \sqrt{M}$ that are each stored as a full-size vector in the M banks available, with small "cut off" effects along the boundary but an otherwise fully periodic pattern.

Lemma 5.4 Let (i_1, j_1) and (i_2, j_2) be two different cells of the matrix. If $s(i_1, j_1) = s(i_2, j_2)$ then either the i -coordinates or the j -coordinates of the two cells differ by at least \sqrt{M} .

Proof.

This is immediate from the "periodicity" of s , and also from the fact that no two cells that are mapped to the same bank can

lie within one $\sqrt{M} \times \sqrt{M}$ block. \square

Now consider an arbitrary, rookwise connected template T of t cells. If $t \leq \sqrt{M}$ then T necessarily fits in a $\sqrt{M} \times \sqrt{M}$ "box" and can obviously be accessed conflict-free with a single fetch.

Theorem 5.5 Using s to store an $N \times N$ matrix into M memory banks, any rookwise connected template of t cells can be retrieved by means of (at most) $\lfloor \frac{t-1}{\sqrt{M}} \rfloor + 1$ conflict-free fetches of vectors from the M memory banks.

Proof.

(The proof requires some familiarity with the Steiner tree problem in the plane, cf. Melzak [11].) Consider the memory banks that receive elements under s from the given instance of T positioned in the domain of the matrix. Suppose bank α receives the largest number of elements from T , and let this largest number be l .

Claim 5.5.1 The elements of the given instance of T can be retrieved by means of (exactly) l conflict-free fetches of vectors from the M memory banks.

Proof

Repeatedly select a largest possible vector with one element of T from each bank that (still) has elements of T . This requires exactly l selections to fully assemble T . Clearly each vector so construed can be fetched conflict-free. No smaller number of fetches can do, because l are necessary to get the elements of T from α conflict-free.

We shall proceed by estimating l .

Consider the matrix as an $N \times N$ square of cells on the two-

dimensional grid, and let the cells be labeled with the name of the memory bank they are mapped to under s . Let V be the collection of cells labeled α that are "covered" by the instance of T . Clearly $l = |V|$. Because T is rookwise connected, every two cells of V are connected by a "rectilinear" chain of cells that runs entirely within T . Consider some minimum collection of chains needed to connect all cells this way. It yields a tree-like substructure of T , with the property that "edges" (chains) may intersect or even partly overlap. Thus we have a tree on a super-set of V (the additional cells can appropriately be called Steiner cells, in analogy to the Steiner tree problem [11]), with cells connected by "simple" rectilinear chains. We conclude that T must contain a "rectilinear Steiner tree" on V , and thus have at least as many cells as a Steiner minimal tree on V with rectilinear edges. Let the length of such a Steiner minimal tree be l_S . It follows that $t \geq l_S$.

Using lemma 5.4 it is fairly straightforward to find a lower-bound of about $(l-1) \cdot \sqrt{M}$ on the length l_{\min} of a "classical" minimum spanning tree of V . This translates to a lowerbound for l_S by using a theorem of Hwang [8], who showed that $l_S \geq \frac{2}{3} l_{\min}$. (Note that l_S can be smaller than l_{\min} because edges that partly coincide with other edges do not contribute the overlapped parts to the length twice.)

Claim 5.5.2 $l_S \geq (l-1) \cdot \sqrt{M} + 1$

Proof.

Hanan's theorem [4] asserts that there is a Steiner minimal tree with rectilinear distance that is contained in the (sub-)grid obtained by taking precisely the rows and columns of the cells in V . (Thus Steiner cells necessarily occur at grid-points only, and edges between Steiner cells and/or cells of V either are

straight chains or chains with one "hook".) By the nature of s columns are at least a distance \sqrt{M} apart, and thus horizontal "lines" must contain at least $\sqrt{M} - 1$ cells (only counting the part between the two columns spanned). Also observe that within a column the occurring cells of V must lie a distance of at least \sqrt{M} apart, and thus vertical "lines" that connect cells of V must be at least $\sqrt{M} - 1$ cells in length.

Now "charge" cells of the Steiner minimal tree to V in the following manner. Choose an arbitrary cell of V as the root of the tree, and orient all edges away from the root. (This implies a notion of "distance" from the root, measured by the number of points of V visited on a path.) Label every edge by "h" or "v", depending on whether it is a horizontal or a vertical connection. (Hooks are labelled by "h".) Note that h-edges account for at least $\sqrt{M} - 1$ cells. Begin by charging 1 to the root, to account for the one cell it occupies. Suppose we have completed the charging to cells of V at distance i from the root. Consider any cell p of V at distance i from the root, and all cells q of V at distance $i+1$ reached from p . The cells are connected to p by a (sub-) tree of labeled edges and Steiner cells as "internal nodes". For every leaf q determine the lowest edge in the tree labeled h and mark it by q .

Case (a): there is no edge labeled h on the path from q to the root (p).

It means that all edges on this path are labeled v, and thus we have a straight vertical connection between two points of V in one column. Charge $\sqrt{M} - 1$ (for the length of the vertical line) and 1 (for the cell q occupies), hence a total of \sqrt{M} cells to q . (Note that Steiner points on this vertical line can only have outgoing h-edges beside the v-edges now accounted for, and no

other leaf can be charged the same cells as q .)

Case (b): The h -edge marked by q has no other marks.

It means that this edge can be uniquely assigned to q and we can again charge $\sqrt{M}-1$ (for the horizontal line) and 1 (for the cell of q), hence a total of \sqrt{M} cells to q .

Case (c): The h -edge marked by q has other marks as well.

Note that in this case the h -edge necessarily ends in a Steiner cell, with one outgoing v -edge continuing on to q over a chain of further v -edges. Say (without loss of generality) that the chain leads from the Steiner cell downwards. The only possibility for the h -edge to be marked by another leaf as well is that there is a cell of V in the same column reached from the Steiner cell by going upward. Thus we conclude that the h -edge can only be marked by one other cell of V , that necessarily lies in the same column as q and is vertically connected to it. Now charge the usual \sqrt{M} cells to q (for the h -edge) and \sqrt{M} cells to the other leaf for the vertical lines.

It follows (by carrying out this procedure for cells of V at increasing distance from the root) that all $l-1$ cells of V beside the root can be charged a unique set of \sqrt{M} cells. Hence we obtain $l_S \geq (l-1)\sqrt{M} + 1$.

We now complete the proof of Theorem 5.5 as follows. By claim 5.5.1 we need l conflict-free fetches to retrieve T . By claim 5.5.2 we have $l \geq l_S \geq (l-1)\sqrt{M} + 1$, hence $l \leq \lfloor \frac{l-1}{\sqrt{M}} \rfloor + 1$. Thus we can retrieve T by means of at most $\lfloor \frac{l-1}{\sqrt{M}} \rfloor + 1$ fetches. \square

By choosing for M a square close to N , the following result is

immediate. (Take e.g. $M = \lfloor \sqrt{N} \rfloor^2$.)

Corollary 5.6 There is a linear skewing scheme using no more than M memory banks, such that every rookwise connected template of N cells in an $N \times N$ matrix can be retrieved in at most $\sqrt{N} + 2$ conflict-free fetches.

For arbitrary, connected templates T (including e.g. diagonals) a precise analysis as in theorem 5.5 is hard, but the following somewhat weaker bound can be obtained.

Theorem 5.7 Using s to store an $N \times N$ matrix into M memory banks, any connected template of t cells can be retrieved by means of at most $\lfloor 2 \cdot \frac{t}{\sqrt{M}} \rfloor + 1$ conflict-free fetches.

Proof.

Follow the same argument as in theorem 5.5 until after claim 5.5.1. To estimate l we now reason as follows. Enclose every cell of V by a "box" of cells that are at most $\lceil \frac{1}{2} \sqrt{M} \rceil$ away from it, measured in cells along a connected (but not necessarily rookwise connected) chain. Note that the boxes indeed are squares, and that the boxes thus surrounding the cells of V are all disjoint. Assuming $l > 1$, the connectedness of T requires that in every box so distinguished there is a chain of cells leading from the middle cell to the boundary. This accounts for at least $\lceil \frac{1}{2} \sqrt{M} \rceil$ cells of T per box, hence $t \geq l \cdot \lceil \frac{1}{2} \sqrt{M} \rceil$ and $l \leq \lfloor 2 \cdot \frac{t}{\sqrt{M}} \rfloor$. The bound stated in the theorem is thus correct, including the case that t is small yet $l = 1$. \square

Choosing again $M = \lfloor \sqrt{N} \rfloor^2$ ($\approx N$) it follows that every connected template of N cells in an $N \times N$ matrix can be retrieved in at most $2\sqrt{N} + O(1)$ conflict-free fetches, using the linear skewing

scheme s.

6. References.

- [1] Barnes, G., et al., The ILLIAC IV computer, IEEE Trans. Comput. C-17 (1968) 746-757.
- [2] Budnik, P., and D. J. Kuck, The organisation and use of parallel memories, IEEE Trans. Comput. C-20 (1971) 1566-1569.
- [3] Euler, L., Recherches sur une nouvelle espèce des carrés magiques, Verh. Zeeuwsch Gen. Wetensch. Vlissingen 9 (1782) 85-239.
- [4] Hanan, M. On Steiner's problem with rectilinear distance, SIAM J. Appl. Math. 14 (1966) 255-265.
- [5] Hardy, G. H., and E. M. Wright, An introduction to the theory of numbers, 5th ed., Clarendon Press, Oxford, 1979 (1st ed. 1938).
- [5] Hedayat, A., A complete solution to the existence and non-existence of Knut Vik designs and orthogonal Knut Vik designs, J. Combin. Th., Ser. A, 22 (1977) 331-337.
- [7] Hedayat, A., and W. T. Federer, On the non-existence of Knut Vik designs for all even orders, Ann. Stat. 3 (1975) 445-447.
- [8] Hwang, F. K., On Steiner minimal trees with rectilinear distance, SIAM J. Appl. Math. 30 (1976) 104-114.
- [9] Kuck, D. J., ILLIAC IV software and application programming,

IEEE Trans. Comput. C-17 (1968) 758-770.

[10] Lawrie, D.H., Access and alignment of data in an array processor, IEEE Trans. Comput. C-24 (1975) 1145-1155.

[11] Melzak, Z.A., On the problem of Steiner, Canad. Math. Bull. 4 (1961) 143-148.

[12] Shapiro, H.D., Theoretical limitations on the efficient use of parallel memories, IEEE Trans. Comput. C-27 (1978) 421-428.

[13] Shapiro, H.D., Generalized Latin squares on the torus, Discr. Math 24 (1978) 63-77.

[14] Vik, K., Bedømmelse av feilen på forsøksfeltet med og uten målestokk, Meldinger fra Norges Landbrukshøgskole 4 (1924) 129-181.

[15] Wijshoff, H.A.G., and J. van Leeuwen, Periodic storage schemes with a minimum number of memory banks, Techn. Rep. RUU-CS-83-4, Dept of Computer Science, University of Utrecht, Utrecht, 1983.