An Application of Number Theory to the Organization of Raster-Graphics Memory

(Extended Abstract)

Benny Chor
Charles E. Leiserson
Ronald L. Rivest

Laboratory for Computer Science
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

Abstract—A high-resolution raster-graphics display is usually combined with processing power and a memory organization that facilitates basic graphics operations. For many applications, including interactive text processing, the ability to quickly move or copy small rectangles of pixels is essential. This paper proposes a novel organization of raster-graphics memory that permits all small rectangles to be moved efficiently. The memory organization is based on a doubly periodic assignment of pixels to M memory chips according to a "Fibonacci" lattice. The memory organization guarantees that if a rectilinearly oriented rectangle contains fewer than $M/\sqrt{5}$ pixels, then all pixels will reside in different memory chips, and thus can be accessed simultaneously.

We also define a continuous analogue of the problem which can be posed as, *"What is the maximum density of a set of points in the plane such that no two points are contained in the interior of a rectilinearly oriented rectangle of area N."* We give a lower bound of $1/2N$ on the density of such a set, and show that $1/\sqrt{5}N$ can be achieved.

1. Introduction

With the development of high-resolution raster-graphics displays, the length of one memory cycle introduces a bound on how quickly the screen can be updated, a bound that may be unacceptable for many real-time or interactive environments. A natural way to avoid this bound is to access more than a single pixel (picture element) at a time. Since the memory is typically partitioned among M random-access memory chips, up to M pixels can be accessed with a single memory cycle, provided that no two pixels reside in the same memory chip.

Figure 1 illustrates a common organization of raster-graphics memory. Each pixel on the screen is assigned to one of M memory chips in row-major order. Thus in every row, the pixels in column m, $M+m$, $2M+m$, and so forth are stored in the the same memory chip m. This organization made a good deal of sense when raster-graphic displays were new and the interface between the raster memory and the CRT was considered complicated. When the screen is refreshed from memory, the line-by-line horizontal scan accesses M pixels in a row and converts them into an analog video signal. But although the memory system achieves maximal parallelism for the screen refresh operation, it can be remarkably inefficient for other operations. Updating a vertical line of pixels, for example, requires a separate memory access for each pixel.

```
1 2 3 4 . . . M 1 2 3 4 . . . M 1 2 3 4 . . . M
1 2 3 4 . . . M 1 2 3 4 . . . M 1 2 3 4 . . . M
1 2 3 4 . . . M 1 2 3 4 . . . M 1 2 3 4 . . . M
1 2 3 4 . . . M 1 2 3 4 . . . M 1 2 3 4 . . . M
1 2 3 4 . . . M 1 2 3 4 . . . M 1 2 3 4 . . . M
1 2 3 4 . . . M 1 2 3 4 . . . M 1 2 3 4 . . . M
1 2 3 4 . . . M 1 2 3 4 . . . M 1 2 3 4 . . . M
1 2 3 4 . . . M 1 2 3 4 . . . M 1 2 3 4 . . . M
```

Figure 1. *A common organization for raster-graphics memory which is efficient for raster scan operations, but poor for vertical updates.*

For arbitrary patterns of access there is no hope of maximal parallelism since whatever the organization, an adversary can choose to access all the bits in a single memory chip. The best we can hope for is to achieve high concurrency for a limited set of operations, which should be large enough to include frequent patterns of usage. And today, since hardware support for screen refresh is relatively well-understood, attention focuses on those operations which make the graphics system easier to program.

Most raster-graphics applications rely on the copying or moving of a rectangle of pixels as a basic operation, which is demonstrated by the fact that this operation is implemented in the microcode of most graphics processors. The ability to move small rectangles quickly is especially important in text-oriented applications.

Recently, a display was developed at Carnegie-Mellon University [2,5] that is designed to move small squares quickly. Figure 2 shows how pixels are assigned to memory chips in the case of $M = 16$ memory chips. The screen is tiled with $\sqrt{M}$-by-$\sqrt{M}$ squares, each of which contains a pixel assigned to a different memory. The attraction of this scheme is that *any* $\sqrt{M}$-by-$\sqrt{M}$ rectilinearly oriented square, whether aligned on tile boundaries or not, contains pixels assigned to different memories. Thus any square of area M can be accessed in one memory cycle.

Unfortunately, the efficiency of the raster-scan operation is reduced in this scheme compared with the one of Figure 1. The line-by-line scan will only be able to access $\sqrt{M}$ pixels in parallel

This research was supported in part by the Defense Advanced Research Projects Agency under Contract N00014-80-C-0622 and by the National Science Foundation under Grant MCS-8006938.

0272-5428/82/0000/0092$00.75 © 1982 IEEE

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

Figure 2. *The 4-by-4 organization for raster-graphics memory. Every 4-by-4 square contains pixels from distinct memory chips.*

because every $\sqrt{M}+1$-by-one horizontal rectangle contains two pixels in the same memory chip. A possible solution to this problem is to stagger the tiles so that the second column of tiles is shifted vertically by one raster, the third by two rasters, and so on. This *ad hoc* solution allows simultaneous access of all pixels in any M-by-one rectangle as well as simultaneous access of all pixels in any $\sqrt{M}$-by-$\sqrt{M}$ square, but it suffers from asymmetry of horizontal and vertical dimensions and introduces a variety of other complications.

This paper asks the question, *"How many memory chips M are required to guarantee that all pixels can be accesses simultaneously in an arbitrary rectillinearly oriented rectangle of N pixels?"* A naıve organization requires $M = N^2$ memory chips, but we can do much better.

This paper uses techniques from number theory to produce novel memory organization of $M \approx \sqrt{5}N$ chips that allows all pixels in any rectangle of area N to be simultaneously accessed. The scheme is regular—a doubly periodic function in the plane—and the constant $\sqrt{5}$ is approached from below, so that for some values of N, the constant is less than two. Furthermore, for the frequently-used operation of accessing a horizontal line, our scheme allows simultaneous access of all M memory chips.

The remainder of this extended abstract is organized as follows. Section 2 introduces a continuous model of the problem that prompted our (discrete) solution. Section 3 presents the doubly periodic organization, and Section 4 provides the number theoretic analysis necessary to prove that the scheme works. Section 5 contains some concluding remarks.

2. A continuous analogue

In this section we introduce a continuous analogue to the discrete problem. *What is the maximum density of a set of points in the plane such that no two points are contained in the interior of a rectilinearly oriented rectangle of area N?* The set of points in this problem corresponds in the discrete problem to the set of pixels which reside in the same memory chip, and the density of points corresponds to the reciprocal of the number of memory chips. The principal difference in formulation is that in the continuous model, we no longer require that the "pixels" fall on grid points.

We shall prove that the density of a set of N-*compatible* points (no two points are contained in the interior of a rectangle of area N) is bounded from above by $1/2N$, and we give a set of N-compatible points that achieves a density of $1/\sqrt{5}N$. This solution motivates the solution to the discrete problem that is given in Section 3.

Let S be a set of points in $\mathbf{R}^2$. We formally define S as being an N-*compatible* set of points if for any pair of points (x_1, y_1) and (x_2, y_2) in the set, we have

$$|(x_1 - x_2)(y_1 - y_2)| \geq N .$$

Around every point P drawn from a set S of N-compatible points, there is an infinite-area *forbidden region* bounded by two hyperbolae inside which no other point of S may lie. Figure 3 shows the forbidden region for a point at the origin. The points in that forbidden region satisfy $|xy| < N$.

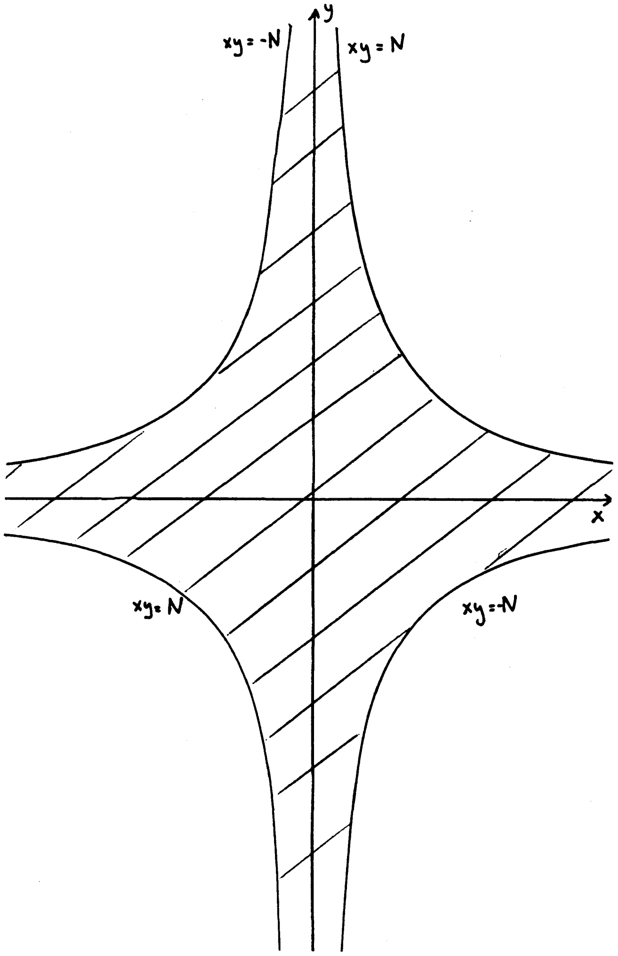


Figure 3. *The forbidden region around the origin. Any set of N-compatible points that contains the origin cannot contain any other point in within the bounds of the hyperbolae.*

We shall find it convenient to adopt some standard terminology from geometry of numbers. A *lattice* is a set of points that can be expressed as an integral, linear combination of linearly independent (over $\mathbf{R}$) *basis vectors.* If there are only two basis vectors, we define the parallelogram with the two basis vectors as sides the *basic region* of the lattice. The *fundamental lattice* is the lattice generated by the basis vectors $(0,1)$ and $(1,0)$, and we call its points *grid points.* Simple properties of lattices can be found in [3].

In the discrete model, the problem is to minimize the number M of memory chips required to allow simultaneous access of any rectangle of N pixels. For an arbitrary scheme of assigning pixels to memory chips, in a square region of A pixels, there will be some memory chip with the largest number k of pixels in the area. Therefore, the number of memory chips M is at least A/k, or $1/d$ where d is the maximum *density* of pixels from a single memory chip in the square region.

The analogue to minimizing the number of memory chips is, in the continuous model, to maximize the density of points in a set of N-compatible points. Formally, we define the *density* of an arbitrary set of points S as

$$d(S) = \limsup_{r\to\infty} \frac{|\{p \in S \mid p \in \mathcal{B}(r)\}|}{\text{Area}(\mathcal{B}(r))} ,$$

where $\mathcal{B}(r)$ is a ball centered at the origin with radius r. The next theorem shows that any set of N-compatible points has bounded density, and in fact, that the density is bounded from above by $1/2N$.

Theorem 1. *Any set S of N-compatible points has density $d(S)$ which is no more than $1/2N$.*

Proof. Consider the lattice that is spanned by the basis vectors $(\sqrt{N}, \sqrt{N})$ and $(-\sqrt{N}, \sqrt{N})$ and shown in Figure 4. We first show that in the interior of any square region of the lattice, there cannot be two points of S.

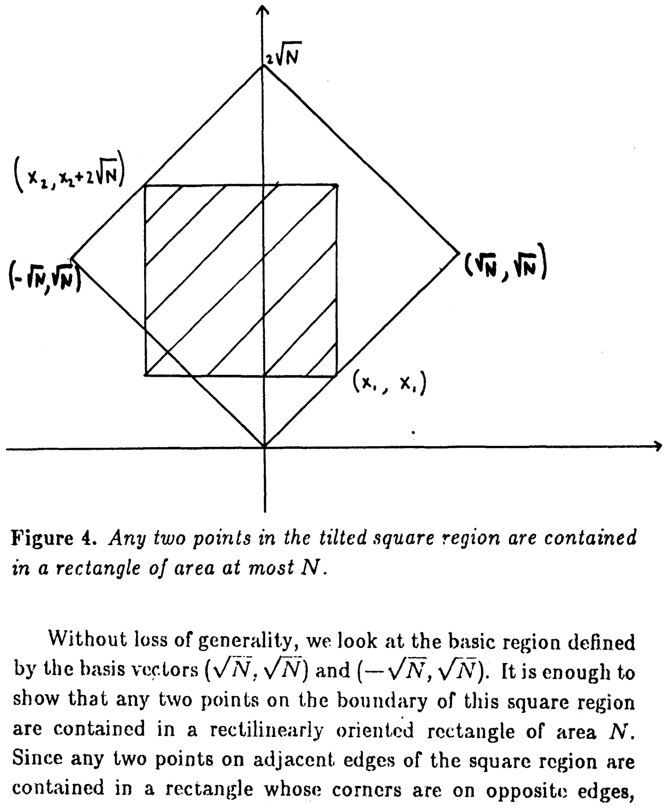


Figure 4. *Any two points in the tilted square region are contained in a rectangle of area at most N.*

Without loss of generality, we look at the basic region defined by the basis vectors $(\sqrt{N}, \sqrt{N})$ and $(-\sqrt{N}, \sqrt{N})$. It is enough to show that any two points on the boundary of this square region are contained in a rectilinearly oriented rectangle of area N. Since any two points on adjacent edges of the square region are contained in a rectangle whose corners are on opposite edges, we may assume the two points are on opposite edges. Suppose the two points are on the edges with positive slope, and are at coordinates (x_1, x_1) and $(x_2, x_2 + 2\sqrt{N})$. The area is

$$\begin{aligned}(x_1 - x_2)(x_2 + 2\sqrt{N} - x_1) &\\ &\le (x_1 - x_2)(2\sqrt{N} - (x_1 - x_2)) \\ &\le \left(\frac{(x_1 - x_2) + 2\sqrt{N} - (x_1 - x_2)}{2}\right)^2 \\ &= N ,\end{aligned}$$

since the geometric mean is less than the arithmetic mean.

Because no two N-compatible points can occupy the same square region of this lattice, and since the area of a square region is $2N$, the density $d(S)$ of a set of N-compatible points cannot be more than $1/2N$, which was to be proved.∎

Whether density as high as $1/2N$ can be achieved is an open question for arbitrary N-compatible sets. We can come close, however, as the following theorem shows.

Theorem 2. *The lattice that is generated by the basis vectors $(\sqrt{N/\phi}, \sqrt{N\phi})$ and $(-\sqrt{N\phi}, \sqrt{N/\phi})$ form an N-compatible set whose density is $1/\sqrt{5}N$, where $\phi = \frac{1}{2}(1 + \sqrt{5})$ is the golden ratio.*

Proof. For simplicity, denote $(\sqrt{N/\phi}, \sqrt{N\phi})$ by (a, b). The lattice points are N-compatible iff for all integers u and v, the lattice point $v(a, b) + u(-b, a) = (av - bu, bv + au)$ is outside the forbidden region around the origin (since the lattice is invariant under translations by its basis vectors). Equivalently, for all pairs $(u, v) \neq (0, 0)$, we must have

$$|(av - bu)(bv + au)| \ge N .$$

We can rewrite the product as

$$\begin{aligned}(av - bu)(bv + au) &= abv^2 + (a^2 - b^2)uv - abu^2 \\ &= N(v^2 - (\phi - 1/\phi)uv - u^2) \\ &= N(v^2 - uv - u^2) .\end{aligned}$$

Since the Diophantine equation $v^2 - uv - u^2 = 0$ has no solution except $u = v = 0$, it follows that

$$|(av - bu)(bv + au)| = N|v^2 - uv - u^2| \ge N ,$$

and thus the lattice points are indeed N-compatible.

The area of the basic region of the lattice is $a^2 + b^2 = N(\phi + 1/\phi)$, which is $\sqrt{5}N$. Since there is a one-to-one correspondence between lattice points and lattice squares, the density is $1/\sqrt{5}N$.∎

Although we have not yet been able to close the gap between the bound of Theorem 1 and that of Theorem 2, we can show that a density of $1/\sqrt{5}N$ is the best possible for any lattice of N-compatible points, no matter how many basis vectors define it.

Theorem 3. *Any lattice of N-compatible points has density at most $1/\sqrt{5}N$.*

Proof. We shall prove the bound on density after we prove that a lattice of N-compatible points can always be generated by two basis vectors. Any three basis vectors in $\mathbf{R}^2$ are linearly dependent over the reals. We show that if the vectors are independent over $\mathbf{Z}$ then the set of points they span will have the origin as an accumulation point, and if they are linearly dependent over $\mathbf{Z}$ then one vector can essentially be omitted.

Suppose three basis vectors are linearly independent over the integers. By a slight variation on Kronecker's Theorem [3, page 382] it can be shown that there is a point in the lattice generated by the three vectors which is arbitrarily close to the origin. But $\sqrt{N}$ is the minimum distance possible between a pair of points in an N-compatible set, so the set of points cannot be N-compatible. Thus we can assume the three basis vectors are linearly dependent over the integers.

If we have three vectors in $\mathbf{R}^2$ that are linearly dependent over $\mathbf{Z}$, then we can find a integer linear transformation with determinant one [1, Lemma 279, p. 311] that maps the three vectors into another three, one of which is the zero vector. The two nonzero vectors span exactly the same set of points because the inverse of an integer linear transformation with determinant one has integer components. By induction, an arbitrary set of basis vectors can be reduced to a set of two.

We may now suppose that we have a set S of lattice points generated by two basis vectors (a, b) and (c, d). The density $d(S)$ is just $1/\Delta$, where $\Delta = |ad - bc|$ is the area of the basic region of the lattice. By a theorem from Minkowski's geometry of numbers [3, Theorem 454, p. 401], there exist two integers x and y, not both zero, such that the rectilinearly oriented rectangle with corners at the origin and the lattice point $x(a, b) + y(c, d)$ has area not exceeding $\Delta/\sqrt{5}$. Since the lattice points are N-compatible, the area of this rectangle is at least N, from which we conclude that $N \leq \Delta/\sqrt{5}$, or $d(S) = 1/\Delta \leq 1/\sqrt{5}N$.▮

The lattice of Theorem 2 achieves this bound, but it is not unique. In fact, Tom Leighton has observed that there are an infinite number of lattices that achieve the bound. For any t the lattice generated by the basis vectors

$$\sqrt{N}\left(t, \frac{1}{t}\right) \quad \text{and} \quad \sqrt{N}\left(\frac{3+\sqrt{5}}{2}t, \frac{3-\sqrt{5}}{2t}\right)$$

also achieves the bound. The lattice of Theorem 2 is a member of this family of lattices (choose $t = \sqrt{1/\phi}$), although the basis vectors given in the theorem are different. The advantage of the basis vectors defined in the theorem is that they define a basic region which is square, and as we shall see in Section 4, this simplifies somewhat the analysis of the discrete solution.

3. A novel organization of raster-graphics memory

This section describes an organization of raster-graphics memory which is based on an approximation of the lattice scheme from Theorem 2. This organization has the property that the pixels in any rectilinearly oriented rectangle that contains no more than N pixels can be accessed simultaneously. The number M of memory chips required is at most $\sqrt{5}N$, but for many practical values it is less than $2N$.

The discrete, real-world problem differs from the continuous problem in that the locations of pixels must be grid points, and this constraint introduces subtle complications. For example, in the continuous problem, a rectangle of area N could be arbitrarily narrow, but in the discrete problem, one-by-N is as far as we can go.

Theorem 2 suggests that we use two basis vectors to generate the locations of all pixels within the same chip of the raster-graphics memory. We assign pixels to chips using the following general scheme. Let a and b be two relatively prime, nonnegative integers. Use the two orthogonal vectors (a, b) and $(-b, a)$ to generate a lattice in the plane, consisting of all points of the form

$$v(a, b) + u(-b, a),$$

where u and v are integers. Except for the corners, no other grid point lies on the boundary of the basic region. Extend the interior of the basic region to include exactly one of the four corner points. Since the region can be used to tile the entire plane, the number of grid points in the basic region is therefore exactly its area, that is, $a^2 + b^2$. The grid points in the basic region are mapped into $M = a^2 + b^2$ distinct memory chips. Since each grid point in the plane has a unique "parent" in the basic region (namely the one in the basic region that differs from it by a unique lattice vector), we assign each grid point to the same chip as its parent.

In the next section, we will show that the choice of successive Fibonacci numbers $a = F_r$ and $b = F_{r+1}$, which yields the number of memories $M = F_{2r+1}$, guarantees that every rectilinearly oriented rectangle containing no more than $M/\sqrt{5}$ pixels can be accessed simultaneously. Figure 5 illustrates the organization for thirteen memory chips ($a = 2$, $b = 3$). Here, the situation is even better than we promised—any rectangle with at most eleven pixels contains no two pixels from the same memory chip. In particular, horizontal and vertical lines of no more than thirteen pixels have no conflicts. This is not mere luck.

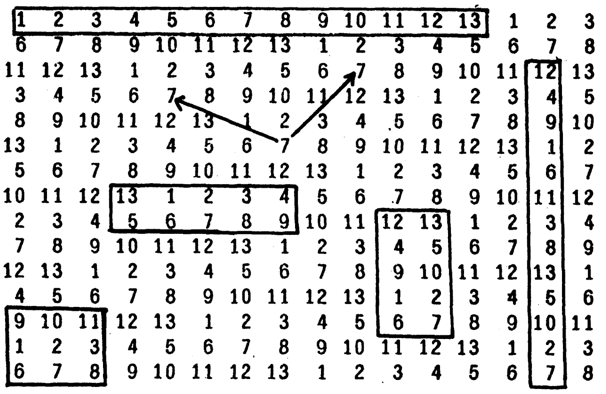
> **Lemma 4:** *A doubly periodic memory organization based on a lattice generated by basis vectors (a, b) and $(-b, a)$, where a and b are positive and relatively prime, has the property that any one-by-M or M-by-one rectilinearly oriented rectangle contains no two pixels from the same chip.*

Proof. Since the organization is doubly periodic, we can consider a horizontal or vertical line that starts at the origin and determine the next lattice point that falls on the line. If the line is vertical, all pixels on it have x-coordinate zero. The general form of lattice points is $v(a, b) + u(-b, a) = (av - bu, bv + au)$, and thus all lattice points on the line will have $av - bu = 0$. It follows that a divides bu, but since a and b are relatively prime, we can conclude that a divides u, and similarly, b divides v. Furthermore, u and v necessarily have the same sign, which means that the magnitude $|bv + au|$ of the v-coordinate is $|bv| + |au|$. Since a divides u, we have $|u| \geq a$, and by the same reasoning, $|v| \geq b$. Therefore, $|bv| + |au| \geq b^2 + a^2 = M$, and the magnitude of any lattice point on the vertical line is at least M. Thus any one-by-M rectangle cannot contain two pixels from the same chip. Horizontal lines are treated the same way.▮

The following table describes the actual values we get for M, N correspondingly for values of M up to 1000.

M	5	13	34	89	233	610
N	5	11	23	53	125	307

Notice that for all these values, the size N of rectangles that are guaranteed to have no conflicts is, in fact, larger than $M/2$. Thus for practical values of M, the overhead in allowing fast access to arbitrarily shaped rectangles of pixels is small.



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

Figure 5. *The lattice-based organization for $M = 13$ memory chips. Every rectangle that contains no more than $N = 11$ pixels has all pixels from distinct memory chips.*

4. Mathematical analysis

In this section, we analyze the properties of the lattice organization described in Section 3 and show that in the organization, the number of memory chips M is approximately $\sqrt{5}$ times the size N of the maximum size rectangle that is guaranteed to have no conflicts. Our approach is to answer the question, *"What is the minimal size MIN of any rectilinearly oriented rectangle containing two distinct lattice points?"* This value MIN determines N because no rectilinearly oriented rectangle of size less than MIN contains two lattice points, so all its pixels are necessarily in different memory chips, and thus $N = MIN - 1$.

We now focus our attention on finding the minimum size MIN over all rectangles containing two lattice points. The basis vectors for the raster-graphics memory organization are (F_r, F_{r+1}) and $(-F_{r+1}, F_r)$, where F_r is the rth Fibonacci number. We shall find it convenient, when we do not rely on the Fibonacci properties basis-vector components, to denote the basis vectors by (a, b) and $(-b, a)$. Since the lattice is invariant under translations by its basis vectors, we lose no generality if, instead of discussing all pairs of lattice points, we restrict ourselves to those pairs one of whose elements is the origin. Furthermore, since we are interested in the minimal size, it suffices to consider rectangles that have the two lattice points at opposite corners.

The second lattice point has the form $v(a, b) + u(-b, a)$, and the size of the rectangle, denoted by $S(u, v)$, is its area plus half its perimeter plus one, *i.e.*,

$$S(u,v) = (|au + bv| + 1)(|-bu + av| + 1)\,.$$

Notice that the size is exactly the number of pixels contained in the closed rectangle. The value MIN is the minimum of $S(u, v)$ over all integers u and v not both 0. In order to find MIN, we first translate $S(v, v)$ into a simpler form.

Lemma 5. *Let*

$$S(u,v) = (|F_r u + F_{r+1} v| + 1)(|-F_{r+1} u + F_r v| + 1)\,,$$

and let $\hat{S}(u,v) = (|F_{2r} u - F_{2r+1} v| + 1)(|u| + 1)$. *Then*

$$\begin{aligned} MIN &\overset{def}{=} \min_{(u,v)\neq(0,0)} S(u,v) \\ &= \min_{(u,v)\neq(0,0)} \hat{S}(u,v)\,. \end{aligned}$$

Proof. We shall show that the range of S is the same as the range of $\hat{S}$ by using an intermediate form B. For simplicity, we shall use the notation $a = F_r$ and $b = F_{r+1}$ introduced above.

Define the intermediate form

$$B(u,v) = S(ku - bv, -lu + av)\,,$$

where k and l are integers such that $ak - bl = 1$. (The integers k and l exist because the greatest common divisor of a and b is one.) The linear transformation given by

$$\begin{pmatrix} u \\ v \end{pmatrix} \to \begin{pmatrix} k & -b \\ -l & a \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix}$$

is a bijection since the determinant of the matrix is one. Thus as (u, v) ranges over $\mathbf{Z}^2$, the ordered pair $(ku - bv, -lu + av)$ also takes on all values in $\mathbf{Z}^2$, and hence the range of S is the same as the range of B. Since the linear transformation is a bijection which maps $(0, 0)$ to $(0, 0)$, we have

$$\min_{(u,v)\neq(0,0)} S(u,v) = \min_{(u,v)\neq(0,0)} B(u,v)\,.$$

If we expand $B(u, v)$, we get

$$\begin{aligned} B(u,v) &= S(ku - bv, -lu + av) \\ &= (|a(ku - bv) + b(-lu + av)| + 1) \\ &\quad \cdot (|-b(ku - bv) + a(-lu + av)| + 1) \\ &= (|u| + 1)(|(a^2 + b^2)v - (bk + al)u| + 1)\,, \end{aligned}$$

which has the form $(|u| + 1)(|Mv - Cu| + 1)$. (Note that $M = a^2 + b^2$ is the number of memory chips.)

In order to obtain $\hat{S}(u, v)$, we first determine the explicit coefficients M and C in $B(u, v)$ when the components of the basis vectors are the Fibonacci numbers $a = F_r$ and $b = F_{r+1}$. We use the following two Fibbonaci identities:

$$F_{i+j} = F_j F_{i+1} + F_{j-1} F_i\,,$$
$$F_{i+1} F_{i-1} - F_i^2 = (-1)^i\,.$$

From the first identity, we get that the number of memories M is

$$\begin{aligned} M &= a^2 + b^2 \\ &= F_r^2 + F_{r+1}^2 \\ &= F_{2r+1}\,. \end{aligned}$$

To find C, observe that the k and l such that $ak - bl = 1$ are $k = (-1)^{r+1} F_r$ and $l = (-1)^{r+1} F_{r-1}$. Hence, by using the second identity, we have that

$$\begin{aligned} C &= bk + al \\ &= (-1)^{r+1}(F_{r+1} F_r + F_r F_{r-1}) \\ &= (-1)^{r+1} F_{r+r} \\ &= (-1)^{r+1} F_{2r}\,. \end{aligned}$$

Thus for $a = F_r$ and $b = F_{r+1}$, we have

$$B(u,v) = (|u|+1)(|(-1)^r F_{2r}u + F_{2r+1}v| + 1)\,.$$

The form $\hat{S}$ was defined in the statement of the lemma as

$$\hat{S}(u,v) = (|u|+1)(|F_{2r}u - F_{2r+1}v| + 1)\,.$$

If r is odd, then $(-1)^r = -1$, and therefore $B(u,v) = \hat{S}(u,v)$. If r is even, on the other hand, then $B(u,-v) = \hat{S}(u,v)$. Since we have already shown that

$$\min_{(u,v)\neq(0,0)} S(u,v) = \min_{(u,v)\neq(0,0)} B(u,v)\,,$$

we get

$$\min_{(u,v)\neq(0,0)} S(u,v) = \min_{(u,v)\neq(0,0)} \hat{S}(u,v)\,,$$

which was to be proved.▮

The next lemma gives the exact solution for MIN, which by Lemma 5 is the minimum value of $\hat{S}(u,v)$.

Lemma 6. *Let* $\hat{S}(u,v) = (|u|+1)(|F_{2r}u - F_{2r+1}v| + 1)$. *Then* $\min_{(u,v)\neq(0,0)} \hat{S}(u,v) = (F_r + 1)(F_{r+1} + 1)$.

Proof. We first show that

$$\begin{aligned} MIN &= \min_{(u,v)\neq(0,0)} \hat{S}(u,v) \\ &= \min_{(u,v)\neq(0,0)} (|u|+1)(|F_{2r}u - F_{2r+1}v| + 1) \\ &= \min_{0\leq n\leq 2r+1} (F_n + 1)(F_{2r-n+1} + 1)\,, \end{aligned}$$

and then show that the latter minimum is $(F_r + 1)(F_{r+1} + 1)$.

It suffices to consider nonnegative values of u since $\hat{S}(u,v) = \hat{S}(-u,-v)$. The value MIN cannot exceed $\hat{S}(0,1) = F_{2r+1} + 1$, but because $\hat{S}(u,v) \geq u + 1$ (the right factor is at least one), we need only seek a better value for MIN in the interval $0 < u < F_{2r+1}$.

The key idea is to divide the half-open interval $[1, F_{2r+1})$ into subintervals $[F_n, F_{n+1})$, for $n = 2, 3, \ldots, 2r$. (Notice that $F_1 = F_2 = 1$, and thus n starts from 2.) The integer u lies inside one of these intervals. Consider the fraction F_{2r}/F_{2r+1}. The convergents of its continued fraction expansion are $F_1/F_2, F_2/F_3, \ldots, F_{2r}/F_{2r+1}$. By the continued-fraction approximation theorem (3, Theorem 181, p. 151), if $F_n \leq u < F_{n+1}$, then for every integer v we have

$$\left|\frac{F_{2r}}{F_{2r+1}} - \frac{v}{u}\right| \geq \left|\frac{F_{2r}}{F_{2r+1}} - \frac{F_{n-1}}{F_n}\right|\,.$$

Multiplying through on both sides yields

$$\left|\frac{uF_{2r} - vF_{2r+1}}{uF_{2r+1}}\right| \geq \left|\frac{F_{2r}F_n - F_{n-1}F_{2r+1}}{F_{2r+1}F_n}\right|\,.$$

Using the Fibonacci identity $|F_iF_j - F_{i+1}F_{j-1}| = F_{i-j+1}$, we get

$$\begin{aligned} |uF_{2r} - vF_{2r+1}| &\geq \frac{u}{F_n}|F_{2r}F_n - F_{n-1}F_{2r+1}| \\ &\geq |F_{2r}F_n - F_{n-1}F_{2r+1}| \\ &= F_{2r-n+1}\,. \end{aligned}$$

To summarize, if u falls in the interval $[F_n, F_{n+1})$, then $|uF_{2r} - vF_{2r+1}| \geq F_{2r-n+1}$. Therefore,

$$(|u|+1)(|F_{2r}u - F_{2r+1}v| + 1) \geq (F_n + 1)(F_{2r-n+1} + 1)\,,$$

and equality is achieved when $u = F_n$ and $v = F_{n-1}$. As a result, we have

$$\begin{aligned} &\min_{(u,v)\neq(0,0)} (|u|+1)(|F_{2r}u - F_{2r+1}v| + 1) \\ &\quad = \min_{0\leq n\leq 2r+1} (F_n + 1)(F_{2r-n+1} + 1)\,, \end{aligned}$$

which completes the first part of the proof.

The second part of the proof is to show that indeed

$$\min_{0\leq n\leq 2r+1} (F_n + 1)(F_{2r-n+1} + 1) = (F_r + 1)(F_{r+1} + 1)\,.$$

If we define

$$E(n,r) \overset{def}{=} (F_n + 1)(F_{2r-n+1} + 1)\,,$$

then what we want to show is

$$\min_{0\leq n\leq 2r+1} E(n,r) = E(r,r)\,.$$

Since $E(n,r)$ is invariant when n is replaced by $2r + 1 - n$, it suffices to consider values of n in the interval $[0, r]$.

We now show that $E(n,r)$ is no larger than $E(n+1,r)$ for $n = 1, \ldots, r-1$, after which we shall complete the proof by demonstrating that $E(0,r) \geq E(r,r)$. We make use of the explicit formula

$$F_r = \frac{\phi^r - \hat{\phi}^r}{\sqrt{5}}$$

for a Fibonacci number in terms of the golden ratio ϕ and its conjugate $\hat{\phi} = \frac{1}{2}(1 - \sqrt{5})$ in order to obtain an alternative expression for the high-order term of $E(n,r)$:

$$\begin{aligned} F_nF_{2r+1-n} &= \left(\frac{\phi^n - \hat{\phi}^n}{\sqrt{5}}\right)\left(\frac{\phi^{2r+1-n} - \hat{\phi}^{2r+1-n}}{\sqrt{5}}\right) \\ &= \frac{\phi^{2r+1} + \hat{\phi}^{2r+1} - \phi^{2r+1-n}\hat{\phi}^n - \hat{\phi}^{2r+1-n}\phi^n}{5} \\ &= C_r + \frac{(-1)^{n+1}}{\sqrt{5}}\left(F_{2r-2n+1} + \frac{2\hat{\phi}^{2r-2n+1}}{\sqrt{5}}\right)\,, \end{aligned}$$

where C_r is a constant depending on r alone. Taking advantage of the fact that $|\hat{\phi}|$ is less than 1 and using the basic recurrence for Fibonacci numbers, we have

$$\begin{aligned} &E(n,r) - E(n+1,r) \\ &= F_{2r+1-n} + F_n + \frac{(-1)^{n+1}}{\sqrt{5}}\left(F_{2r-2n+1} + \frac{2\hat{\phi}^{2r-2n+1}}{\sqrt{5}}\right) \\ &\quad - F_{2r-n} - F_{n+1} - \frac{(-1)^{n+2}}{\sqrt{5}}\left(F_{2r-2n-1} + \frac{2\hat{\phi}^{2r-2n-1}}{\sqrt{5}}\right) \\ &\geq F_{2r-n-1} - F_{n-1} - \frac{F_{2r-2n}}{\sqrt{5}} - 1 \\ &\geq F_{2r-n-1} - F_{n-1} - F_{2r-2n} \\ &\geq 0\,, \end{aligned}$$

and hence $E(n,r)$ is at least as large as $E(r,r)$ for $n = 1,\ldots,r-1$.

As for the remaining inequality $E(0,r) \geq E(r,r)$, it is merely $F_rF_{r+1} + F_{r+2} + 1 \leq F_{2r+1} + 1$, and its truth may be verified by using the identity $F_r^2 + F_{r+1}^2 = F_{2r+1}$.▮

Lemma 7. *The minimum size of a rectilinearly oriented rectangle that contains two points of the lattice generated by the basis vectors (F_r, F_{r+1}) and $(-F_{r+1}, F_r)$ is*

$$MIN = (F_r + 1)(F_{r+1} + 1)\,.$$

Proof. The proof follows directly from Lemmas 5 and 6.▮

Theorem 8. *Let $M = F_{2r+1}$, and let $N = F_rF_{r+1} + F_{r+2}$. Then there is an organization for raster-graphics memory with M memory chips such that every rectilinearly oriented rectangle of size at most N contains pixels from distinct memory chips. Furthermore, N is greater than $M/\sqrt{5}$.*

Proof. From Lemma 7, we have that $MIN = (F_r + 1)(F_{r+1} + 1)$, and since $N = MIN - 1$, we get $N = F_rF_{r+1} + F_{r+2}$. All that is left to be proved is that $N > M/\sqrt{5}$. Using the the explicit formula for Fibonacci numbers, it can be verified that the sequence

$$\left\{\frac{F_rF_{r+1} + F_{r+2}}{F_{2r+1}}\right\}_{r=1}^{\infty}$$

converges to $1/\sqrt{5}$. We now show that this sequence is monotonically decreasing, so each of its elements is at least as large as the $1/\sqrt{5}$ limit, which will complete the proof.

It is enough to show that the difference of consecutive terms in the sequence is positive, or equivalently, by multiplying through that

$$F_{2r+3}(F_rF_{r+1} + F_{r+2}) - F_{2r+1}(F_{r+1}F_{r+2} + F_{r+3}) > 0\,.$$

Using the explicit formula for Fibonacci numbers, we obtain the identity

$$F_{2r+3}F_r - F_{2r+1}F_{r+2} = (-1)^{r+1}F_{r+1}\,,$$

and the identity

$$F_{2r+3}F_{r+2} - F_{2r+1}F_{r+3} = F_{2r+1}F_r + F_{2r}F_{r+2}$$

may be derived by induction.

Multiplying both sides of the first identity by F_{r+1} and adding it to the second yields

$$\begin{aligned} F_{2r+3}(F_rF_{r+1} + F_{r+2}) &- F_{2r+1}(F_{r+1}F_{r+2} + F_{r+3}) \\ &= F_{2r+1}F_r + F_{2r}F_{r+2} + (-1)^{r+1}F_{r+1}^2\,. \end{aligned}$$

The right hand side is positive because F_{r+1} is less than both F_{2r} and F_{r+2}.▮

The next theorem is the discrete analogue of Theorem 1.

Theorem 9. *For any organization of raster-graphics memory with M memory chips such that every rectilinearly oriented rectangle of size N contains no two pixels in the same memory chip, the relation $M \geq 2N - 4\sqrt{N} + 2$ holds.*

Proof. The proof parallels that of Theorem 1. The principal difference is that the size of a rectangle includes not only its area, but also half its perimeter plus 1. We tile the plane with tilted squares generated by the two vectors $(\sqrt{N} - 1, \sqrt{N} - 1)$ and $(-\sqrt{N} + 1, \sqrt{N} - 1)$. (The lattice points do not necessarily fall on grid points.) Consider two points within a tile. By the same argument as in the proof of Theorem 1, the area of the rectilinearly oriented rectangle whose opposite corners are the two points is at most $(\sqrt{N} - 1)^2$. Half the perimeter is at most the diagonal of the square tile, that is, at most $2\sqrt{N} - 2$. Hence the size of the rectangle is at most $(\sqrt{N} - 1)^2 + (2\sqrt{N} - 2) + 1$, which equals N, and the two points must be from different chips.

Since the two points were chosen arbitrarily from within a square tile, all grid points within the tile must be from distinct chips. Although the number of grid points may vary from tile to tile, there is a tile that contains at least as many grid points as its area $2(\sqrt{N} - 1)^2$, which completes the proof.▮

5. Addressing scheme

The organization for raster-graphics memory proposed in Section 3 guarantees that small rectangles contain pixels from distinct memory chips. In order for the entire system performance to benefit from this organization, however, the address calculations must be easily implemented. We do not try to solve all the engineering problems associated with making this memory organization scheme work, but in this section we give indications of how the address calculations can be efficiently computed.

The addressing mechanism must be able to take the x- and y-coordinates of a pixel and generate the chip number and address within the chip. Suppose the lattice organization is determined by two basis vectors (a,b) and $(-b,a)$. Two pixels at locations (x_0, y_0) and (x,y) which differ by an integral linear combination of the the basis vectors lie in the same memory chip. That is, they have the same memory number if there exist (unique) integers U and V such that

$$(x,y) - (x_0, y_0) = U(a,b) + V(-b,a)\,.$$

One natural, but inefficient, addressing mechanism is based on the fact that each of the $M = a^2 + b^2$ memory chips contains exactly one representative in the basic region with corners $(0,0)$, (a,b), $(-b,a)$ and $(a-b, a+b)$. The chip number of a pixel (x,y) can be determined by computing which pixel (x_0,y_0) in the basic region is from the same chip, and then using the ordered pair (x_0,y_0) as the chip number. By letting

$$U \doteq \left\lfloor \frac{ax + by}{a^2 + b^2} \right\rfloor, \quad V = \left\lfloor \frac{ay - bx}{a^2 + b^2} \right\rfloor,$$

the chip number (x_0,y_0) of a pixel (x,y) is then $(x_0,y_0) = (x,y) - U(a,b) - V(-b,a)$. Furthermore, the ordered pair (U,V) forms an appropriate address for the pixel (x,y) within the chip.

The addressing mechanism can be simplified substantially if we notice that any arbitrary set of M pixels, no two of which are from the same chip, can be used as a set of representatives. In particular, any pixel differs by an integral linear combination of the basis vectors from a unique pixel in the horizontal line extending from $(0,0)$ to $(0, M-1)$. This scheme corresponds to tiling the plane with one-by-M bricks instead of tilted squares. (Holladay [4] uses a similar tiling scheme for halftone generation.)

To derive an appropriate addressing scheme, we choose an alternative pair of basis vectors that span the same lattice. Since a and b are relatively prime, there exist integers k and l such that $ak-bl = 1$. The two vectors $(bk+al, 1)$ and $(a^2+b^2, 0)$ generate the same lattice as the original basis vectors $(a, b), (-b, a)$. Thus any pixel (x, y) can be mapped to a pixel (x_0, y_0) where $y_0 = 0$ and $x_0 \in [0, M)$, which means x_0 alone can serve as the chip number for the pixel. If we denote $C = bk + al$, and recalling that $M = a^2 + b^2$, the chip number for an arbitrary pixel (x, y) is $x - Cy \pmod{M}$. The address of the pixel the ordered pair $(\lfloor x/M \rfloor, y)$, which is also easy to compute.

An advantage of any doubly periodic organization that should be mentioned concerns the communication among the memory chips. Typically, each chip has a single connection to an M-pixel buffer. To move a rectangle of pixels, three steps are required. The rectangle of pixels is read into the buffer, the pixels in the buffer are permuted, and the pixels are written back to the memory chips at different locations. The advantage of the periodic organization is that the set of permutations encompasses only circular shifts of the buffer. Thus a standard barrel shifter can be used for all permutations.

One issue that we have not faced is the problem of generating addresses for each of the M chips given some standard specification of the rectangle to be accessed. But the strong regularity of any lattice-based organization should make the address calculations possible at reasonable cost.

6. Comments

There is still a discrepancy between the lower bound of $1/2N$ and upper bound of $1/\sqrt{5}N$ on the density of an N-compatible set. It seems more likely that the lower bound can be improved because in the proof of the bound, $\sqrt{2N}$-by-$\sqrt{2N}$ regions that tile the plane account only for interactions between pairs of points.

Another open question is how to extend our results to dimensions higher than two, and whether the linear relation between M and N (or N and the density in the continuous case) still remains true.

There is a practical memory organization which is based on the lattice generated by the basis vectors $(1, s)$ and $(-s, 1)$. This scheme allows three types of rectangles—s-by-s, one-by-$s^2 + 1$, and $s^2 + 1$-by-one—to be accessed efficiently. The number of memories required by this scheme is $M = s^2 + 1$.

The Fibonacci lattice organization can also be used to speed up the access rate in machines with interleaved memories. Matrix and image processing applications could find the organization particularly useful.

Acknowledgements

A preliminary analysis of the continuous model was performed in conjunction with Satish Gupta and Dan Hoey of Carnegie-Mellon University. Tom Leighton of MIT suggested portions of the argument used in the proof of Theorem 3. Ron Pinter of MIT contributed to several programs that originally convinced us that Fibonacci organization really works. We would also like to thank Shai Haran of MIT for helpful discussions and Lyle Ramshaw of Xerox PARC for acquainting us with [4].

References

[1] Gauss, C. F., *Disquisitiones Arithmeticae,* translated by A. A. Clarke, Yale University Press, New Haven, Connecticut, 1966.

[2] Gupta, S., *Architectures and Algorithms for Parallel Updates of Raster Scan Displays,* Ph.D. Thesis, Carnegie-Mellon University, December 1981.

[3] Hardy, G. H., and E. M. Wright, *An Introduction to the Theory of Numbers,* Oxford University Press, London, 1968.

[4] Holladay, T. M., "An optimum algorithm for halftone generation for displays and hard copies," *Proceedings of the Society for Information Display,* Vol. 21, Num. 2, 1980, pp. 185-192.

[5] Sproull, R. F., I. E. Sutherland, A. Thompson, S. Gupta, and C. Minter, "The 8-by-8 display," Tech. Report CMU-CS-82-105, Carnegie-Mellon University, December 1981.