A Binary Map as a Representation System

Julean Simon, Wien, Hochschule der Kuenste, Berlin, 1992

Introduction

A sequence expresses the interrelation of elements of a class as a linear distribution. Then a linear sequence - say the prime number series - is obviously best represented in a linear representation system and can be graphed e.g. as a bar-code:

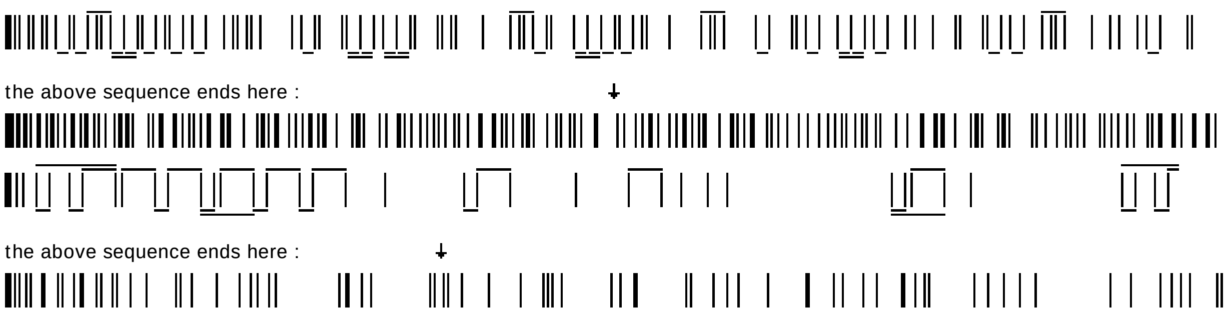


Fig 1. The first panel represents the distribution of prime numbers between 1 and 571 as a sequence of vertical bars patterns in this sequence have been indicated by an underscore; the striking frequency of pairs of bars needs no indicator. The second panel emphasizes these primes-pairs by decreasing the resolution, so that a pair is indicated by a bold bar. The next sequence represents only the distribution of these bars, as if the ‘gray levels’ would have been filtered out. In the last panel the resolution has been changed again, therefore it represents the distribution of prime number pairs between 1 and 3000.

To represent the primes up to, say, 32.000 we will at some point decide to interrupt the succession and start a new line, as we do in writing or counting, combining ‘four - teen’. In the following diagram sequential pieces of the prime series up to 32.385 have been arranged as schematized. A prime is marked as a point at a line/column-position corresponding to its value. This distribution looks fairly homogeneous. How are we to explain those missing lines ?

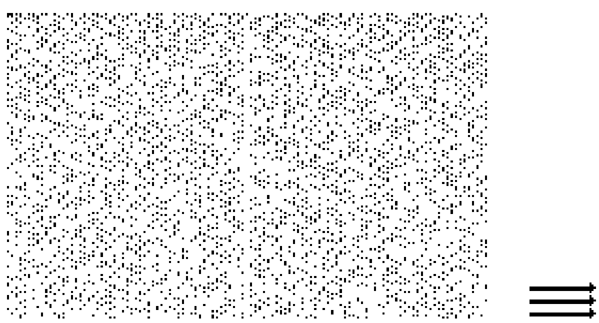


Fig 2. Primes. There are two categories of missing lines: Primes are certainly not even numbers. Since w=226 is even every second column is out of reach. The missing middle line (x=113) is prime : a point is plotted in the (y=0) line. For all other (x=113) positions the answer to the question whether (113 + y* w) is a prime is negative. 113=w/2 therefore : w/2 + y*w is not prime since w/2*(1+2y) is a composite number for all y<> 0.

This is not too sensational; it demonstrates the structural influence of the representation system on the subject to be represented: The missing lines exemplify the ‘reflection’ of the ‘border condition’ (w) on the representation. If value w is odd, the phenomenon shifts from row to row and the missing lines become 45 degree diagonals.
This border condition is a rather arbitrary order principle of the distribution, if we compare the neighborhood of numbers in the horizontal and the vertical dimension.

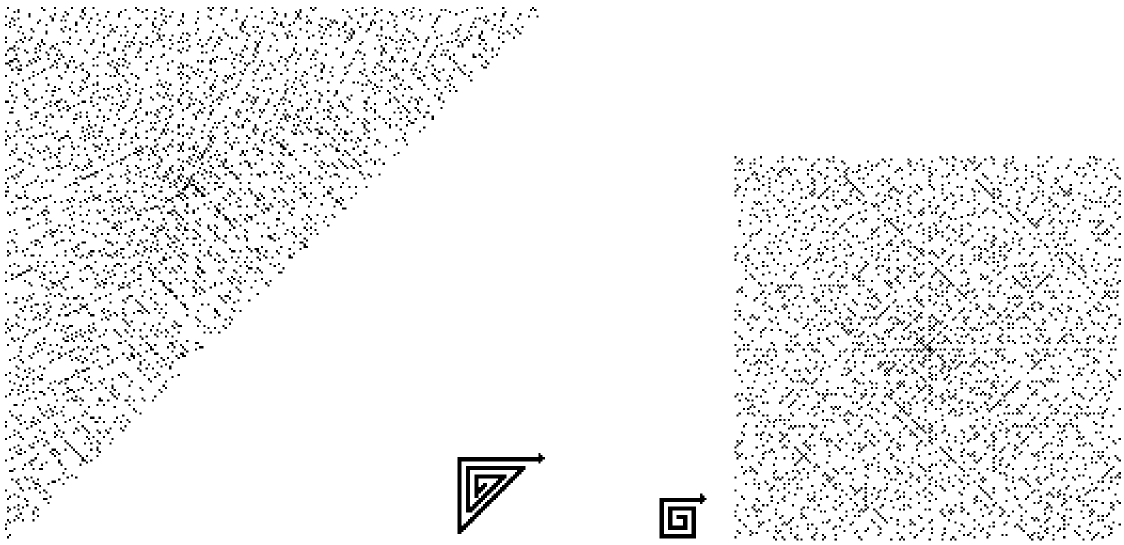


Fig. 3 Two different strategies to represent the prime series up to 32385 and the according order principle. [The vertical and horizontal ‘empty’ rays through the center of the square correspond to series of composite numbers, which can be generated by (n := 7; n :=n + 8 * i + 7) and (n := 10; n :=n + 8 * i + 9) and their parallel rays (n-1). The factors of these composite numbers, t1 and t2, are prime and show the relations: (4*t1-5 = t2), (4*t1+3 = t2), (4*t1+5 = t2) and (4*t1-3 = t2).]

The point here is to demonstrate the relational character of both a sequence and its interaction with a system of representation (structure principle of representation in general). These relations are reflected in their particular distribution. The more compatible a system of representation is with the features of a sequence to be represented, the more of course it reveals its characteristics through regularities and significant forms. Such patterns may be the basis to establish new (contextual) rules and to perform transformations. Meaning and significance is assigned to an image (a chart) according to a context within which the particular form of the image expresses a consistent sentence.
Thus every image inheres the system of its representation as a structure. The structure of a TV.image incorporates that of the TV.set, the TV.station..., into the graphical and contextual structure of the image as such. A picture by means of x-ray, an electron microscope or by a Telescope is also a metaphor of the technology used. This structure affinity between ‘transmitter and message’ is especially applied to systems which are supposed to extract a particular class of data from the stream of ‘world noise’: The logic of a record reflects the logic of the recording system.

In the following we will develop a two dimensional representation system on the basis of the binary notation.

Binary notation uses two symbols, '0' and '1' [1] to represent a number. It follows, that successive digits of a number are place holders for increasing powers of two: The binary number '101101' is the sum of $1*2^0+ 0*2^1+ 1*2^2+ 1*2^3+ 0*2^4 +1*2^5$ (45 in decimal notation). When a digit is a place holder for

[1] It could of course use any other pair, e.g. true and false.

either '0' or '1', then a number can also be represented as a path through a 'binary tree', where a node has one parent node and two child nodes, and a level of nodes corresponds to equal powers of two:

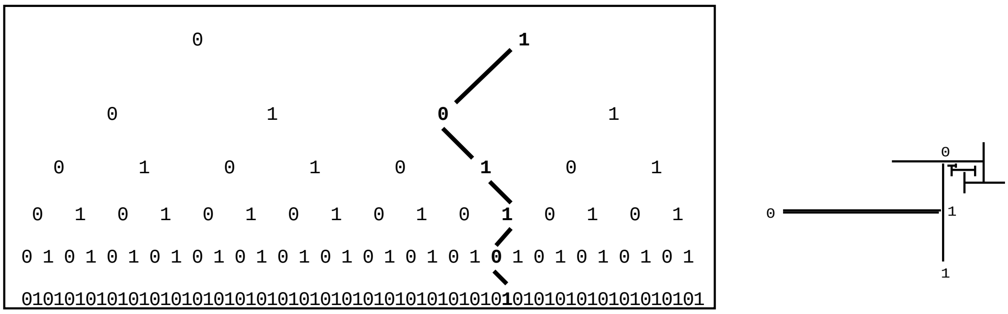


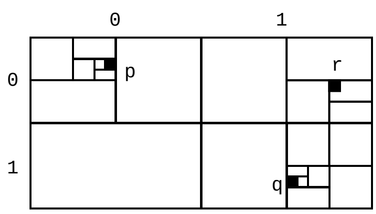
Fig 4. left: '101101' represented as the path through a 'binary tree' .
right: '101100001' represented as a binary branching in a plain: The tree (left) is seen from the top and branches on every second level are turned 90 degrees. The plain thereby is arranged as the organization of (an infinite dense distribution of) discrete positions. The same system can be explained by the following procedure:

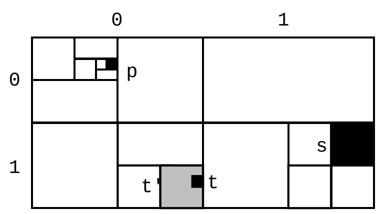
Binary Mapping

Let me pursue the idea to implement this tree-structure as the organization of a two-dimensional[2] space. A dot on a sheet of paper can be expressed by a single number, performing the following procedure:

Fold the sheet and, if the upper half contains the dot, keep a '0' in memory, otherwise a '1'. Turn the folded sheet 90 degrees, fold it again and store the figure indicating the location of the dot. Turn it -90 degrees and repeat the whole procedure until the size of the folded paper matches the dot.

The binary number which is generated by this algorithm, describes the position of the dot up to an arbitrarily high precision, limited only by your skill[3]. The number of foldings (the number of digits, respectively) indicate the 'resolution'. A digit corresponds to the size of a partition of the sheet. The iterated folding-procedure creates an increasingly dense packed space of positions.
Instead of crumpling paper, let us now investigate the system a little more theoretically.
We will speak of the sheet of paper as 'field' or 'frame', and call the smallest resolvable picture element 'pixel', as usually.
With regard to the resolution (res) this system can represent 2^{res} pixels within a given frame.
The field equals the pixel when (res=0), for (res=8) an image is differentiated in 256 pixels.
If [res = ∞] we are dealing with ∞ pixels.
The number of digits determines the size of a pixel. The sequence of figures hierarchically establishes its position. The complementary digit-sequence causes a 180 dgr rotation of the position around the center of the field.





[2]general : x-dimensional system of representation
[3]comp.: accuracy of a measurement

Fig. 5 left: The position of the squares, p and q, is sufficiently determined by the binary numbers 000011101, 111100010 respectively. The complementary digit-sequence causes a 180 dgr rotation. The highest digit corresponds to the whole field; r shows the position of a square representing the number 000011101, where the lowest digit corresponds to the whole field (inverted p).

right: Numbers with lesser digits than the resolution, are not zero-extended. The binary value of p without preceding 0's, e.g. 11101, corresponds to a square s. If lower digits of a number, say 011110111 denoting position t, are ignored, we get a pixel e.g. t' for 01111, where pixel t' contains pixel t at sub-position 0111.

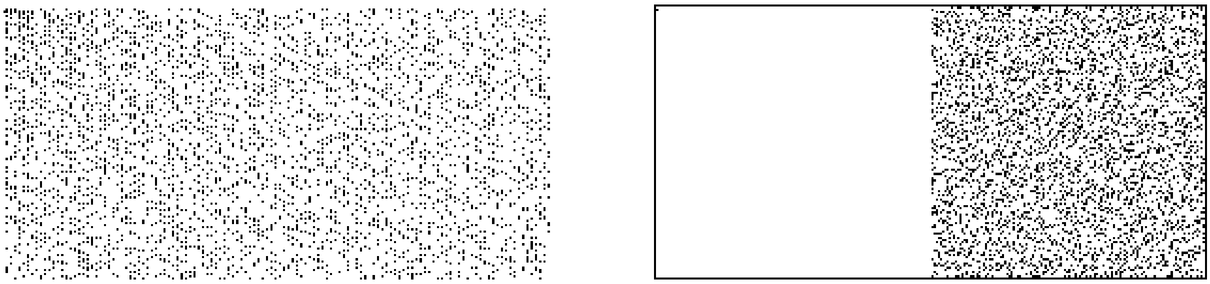


Fig. 6 Binary mapped distribution of primes between 0 and 32767. Left: lowest digit corresponds to the pixel, right: lowest digit corresponds to the frame. Here, the left half is empty (except 2) since the lowest digit of a prime is 1.

prime	binary notation	plotposition x	y
1	(000000000000001)	1	0
2	(000000000000010)	0	1
3	(000000000000011)	1	1
5	(000000000000101)	3	0
7	(000000000000111)	3	1
11	(000000000001011)	1	3
13	(000000000001101)	3	2
17	(000000000010001)	5	0

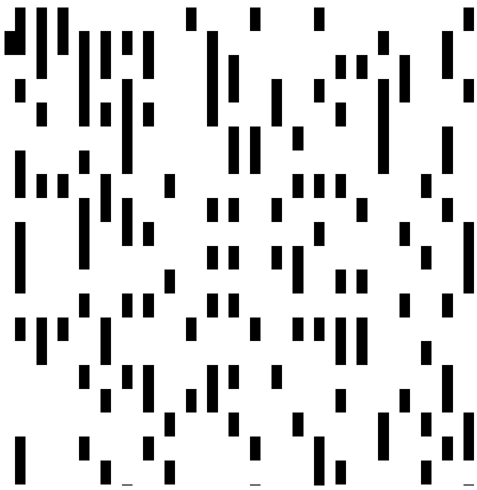


Fig. 7. Left: binary notation of primes and x/y coodinates. Right: Primes distribution (zoom) shows vertical lines which consist of primes whose difference corresponds to (what we will call later) a ‘line-difference’ of 2 (prime pairs), or 6, 22, 86, etc.

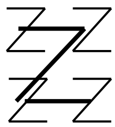
It is our decision, whether the lowest digit should relate to the frame or to the pixel, which is equivalent to the inversion of the digit-sequence and produces, as we saw in Fig. 6, different graphical output. In the following we will use the system, where the lowest digit corresponds to the pixel-level.

Another decision concerns the relation between frame and resolution: In order to represent arbitrary large numbers, we may either increase the resolution (let pixel size -> 0), or think the frame to be open to the right and downwards.

We also could treat the field as a <u>unit-space, in order to represent numbers between 0 and 1 with an arbitrary high precision</u>.

Features of the binary map

Description of some of the properties of this two-dimensional arrangement of numbers.

res	numberfield	decimal notation	scheme
2	00 01 10 01	0 1 2 3	
4	0000 0001 0100 0101 0010 0011 0110 0111 1000 1001 1100 1101 1010 1011 1110 1111	0 1 4 5 2 3 6 7 8 9 12 13 10 11 14 15	

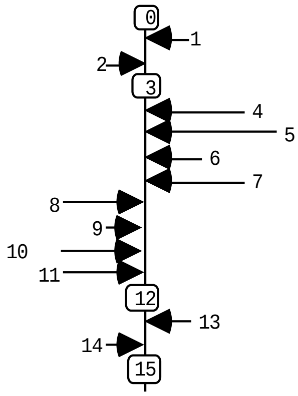


Fig. 8. left: Number fields for (res=2) and (res=4). right: the two dimensional distribution of numbers in relation to the linear number scale, which can be seen in the main diagonal through 0.

<u>The building rule of the number-arrangement has the form of a 'Z', and shows, with respect to the resolution, scaling symmetry</u>.

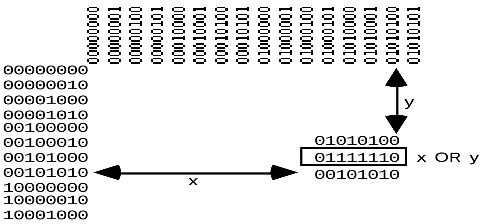
		2^0	2^2	2^2+2^0	2^4	2^4+2^0	2^4+2^2	$2^4+2^2+2^0$	2^6	2^6+2^0	2^6+2^2	$2^6+2^2+2^0$	2^6+2^4	$2^6+2^4+2^0$	$2^6+2^4+2^2$	$2^6+2^4+2^2+2^0$
	0	1	4	5	16	17	20	21	64	65	68	69	80	81	84	85
2^1	2	3	6	7	18	19	22	23	66	67	70	71	82	83	86	87
2^3	8	9	12	13	24	25	28	29	72	73	76	77	88	89	92	93
2^3+2^1	10	11	14	15	26	27	30	31	74	75	78	79	90	91	94	95
2^5	32	33	36	37	48	49	52	53	96	97	100	101	112	113	116	117
2^5+2^1	34	35	38	39	50	51	54	55	98	99	102	103	114	115	118	119
2^5+2^3	40	41	44	45	56	57	60	61	104	105	108	109	120	121	124	125
$2^5+2^3+2^1$	42	43	46	47	58	59	62	63	106	107	110	111	122	123	126	127
2^7	128	129	132	133	144	145	148	149	192	193	196	197	208	209	212	213
2^7+2^1	130	131	134	135	146	147	150	151	194	195	198	199	210	211	214	215
2^7+2^3	136	137	140	141	152	153	156	157	200	201	204	205	216	217	220	221
$2^7+2^3+2^1$	138	139	142	143	154	155	158	159	202	203	206	207	218	219	222	223
2^7+2^5	160	161	164	165	176	177	180	181	224	225	228	229	240	241	244	245
$2^7+2^5+2^1$	162	163	166	167	178	179	182	183	226	227	230	231	242	243	246	247
$2^7+2^5+2^3$	168	169	172	173	184	185	188	189	232	233	236	237	248	249	252	253
$2^7+2^5+2^3+2^1$	170	171	174	175	186	187	190	191	234	235	238	239	250	251	254	255

Fig. 9. The order of numbers in the binary map exemplified by the top left corner of an open frame. For simplicity the decimal notation is used.

We will speak of horizontal <u>lines</u> and vertical <u>rows or columns</u>. 45 degree rays we will call <u>diagonals</u>. Those diagonals through 0 (and e.g. 60) will be called <u>main diagonal</u>. And a diagonal perpendicular to a main diagonal, reaching from the first line to the first row, will be called <u>base diagonal or base</u>.

The first line of the number-field contains increasing numbers, which can be written as the sum of powers of two with even exponents and the exponent 0, the first column contains those with odd exponents. These two series of numbers are the dimensions of the number field and assign an x/y position to each number in the field, where x denotes the column number and y the line number. These sums of powers of two (left and above of the number field in Fig. 9) not only equal the numbers in the first line/column, but are significant for the whole column/line they start.

Every number in the field results from adding the coordinated sums of powers of two with even and with odd exponents. Therefore, a number simply is the sum of the corresponding numbers in the first line and first column. (226 = 162+64, i.e. $2^7+2^6+2^5+2^1$).
All numbers in a line have equal partial-sum of powers with odd exponents, for all numbers in a column the sum of powers with even exponents (or 0) is equal.




```
function   val2pos (val: longint): point;
  var      i, x, y: integer;

 begin
  x := 0;
  y := 0;
  for  i := 0 to 15 do
   if  btst(val, i * 2) then
    x := x + bsl(1, i);
  for  i := 1 to 16 do
   if  btst(val, (i * 2) - 1) then
    y := y + bsl(1, i - 1);

  val2pos.h := x;
  val2pos.v := y;
end ;
```

Fig. 10. left: The 'crossing' of the even and odd powers of two in binary notation. To find the position of a number, separate the even and odd powers. In order to get the value of a position add the sums of powers coordinated to the position, or perform the bit-operation [x OR y], where x and y are the coordinated binary numbers.
right and below: The algorithms to convert the value of a number into the corresponding position, vice versa.
The bit operations: bit set (bset), bit test (btst), bit shift left (bsl), bit clear (bclr) have direct access to the binary representation of the numbers. Val2pos : btst(val,i*2) tests every even digit and, if not 0, increases the x-value by the corresponding power of two. The same procedure is performed with odd digits (i*2-1) for y-values.
The transformation of coordinates to the corresponding number:

```
for  i := res downto  0 do begin
    a[i] := x div  bsl(1, i);
    x := x mod bsl(1, i);
   end;
```

$$\xi(x)=\sum_{i=0}^{res(x)} a_i * 4^i$$

$$\psi(y)=2*\sum_{i=0}^{res(y)} a_i * 4^i$$

$$val_{(x,y)}=\xi(x)+\psi(x)$$

```
function   pos2val (x, y, res: integer): longint;
    var     i, ii: integer;     val: longint;
    procedure   digSort (coord, j: integer);
    begin
     if Btst(coord, j) then
       Bset(val, i)
     else
        bclr(val, i);
      i := i + 1;
     end;
begin
  val := 0;
  i := 0;
   for  ii := 0 to res do begin
      digSort(x, ii);
     digSort(y, ii);
   end;
  pos2val := val;
end;
```

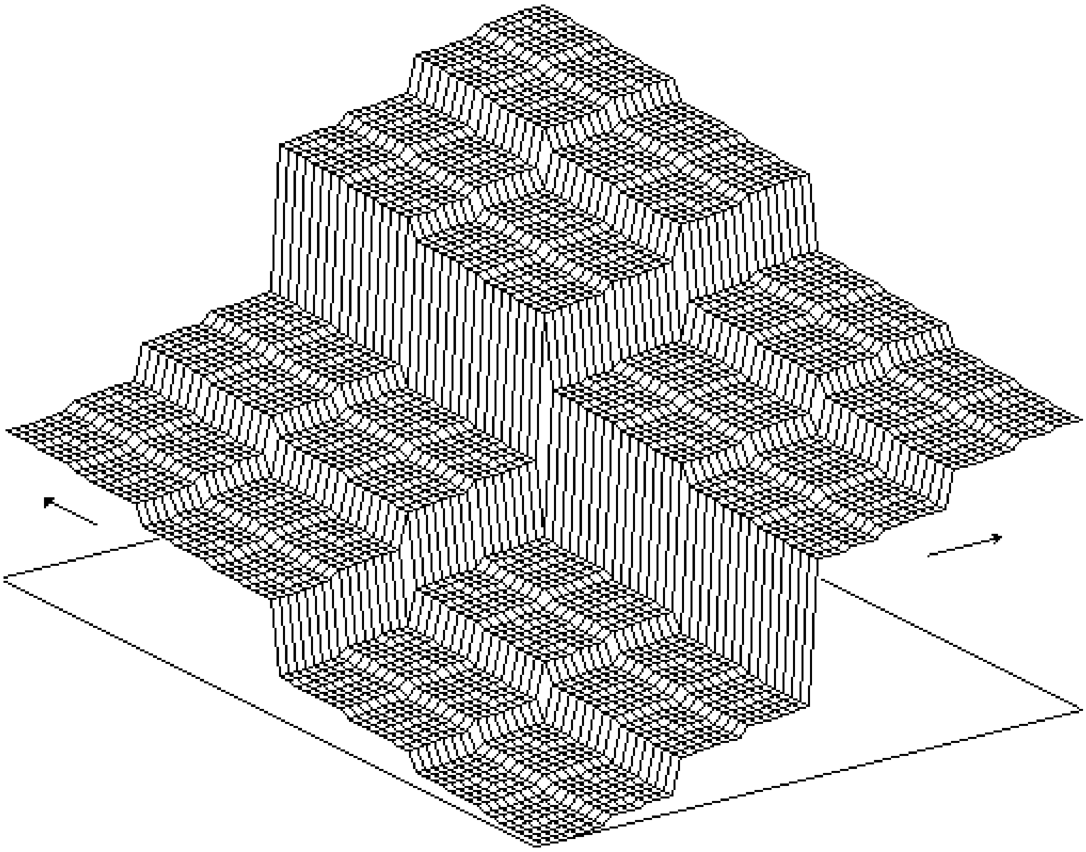


Fig 11. A partition of the number-field, seen from 0, with the values used as heights, demonstrating the scaling symmetry of the number-arrangement. A number is, of course, represented by a horizontal plane, the vertical steps only illustrate the coherence of the numbers-space.

Numbers in horizontal sequences show a 1:2-relation to numbers in vertical sequences: The sequence of numbers in every line corresponds to a sequence of numbers in every second column.
See Fig. 9: 0 corresponds to 0, 126 to 252. So 6, 7, 18,...corresponds to 12, 14, 36 etc., but also in the other direction 246, 244 and 222 corresponds to 123,122 and 111. This relation works the other way round, too: 118,124,126 corresponds to 236, 248,252 (every second number in horizontal direction).

The difference between numbers at the same position in two lines or columns is constant.
(e.g. 63-47=181-165 or 181-63=165-47). e.g. line-difference 2 holds between line zero and line one; column difference 11 between third and forth column. Since this relation holds for lines and columns, we get :

Every two crossing diagonals[4] of numbers, which have equal length and intersect in their center have equal sums, no matter whether the crossing point is a number, or falls between four numbers. This equation holds for all patterns with an x- and y-symmetry.

---

[4]The term 'diagonal' denotes a connecting line between numbers, where the difference of lines and columns is equal: $|x_2-x_1|=|y_2-y_1|$. The 'O-diagonal' or 'main diagonal' is a line through numbers, where x=y. e.g. the origin 0, 3, 12...

32+(35+44+)47 = 42+(41+38+)37
36+(15+(24)+19)+20 = 4+(7+(24)+27)+52
72 = (29+66)-23

(34+40+41)+(13+24+26) = (8+9+10)+(45+50+56)

Fig. 12. Since the field is open, it is here represented as a triangle, that can grow downwards.
left: Crossing diagonals have equal sums. This lets us 'far-fetch' a certain similarity to the Pascal Triangle: a number is the sum of the two numbers above it (but ! - minus the number above these two 29+66 -23 =72).
right : x- and y-symmetrical patterns.

The number-arrangement shows properties akin to magic squares: Given a number-square with arbitrary size, at arbitrary position within the field, with its sides on lines and columns. We already know: the sums of numbers in the two diagonals of the square are equal. But all other parallel lines to these diagonals will yield equal sum, too, provided, two are pieced together, in the way shown in Fig. 13.

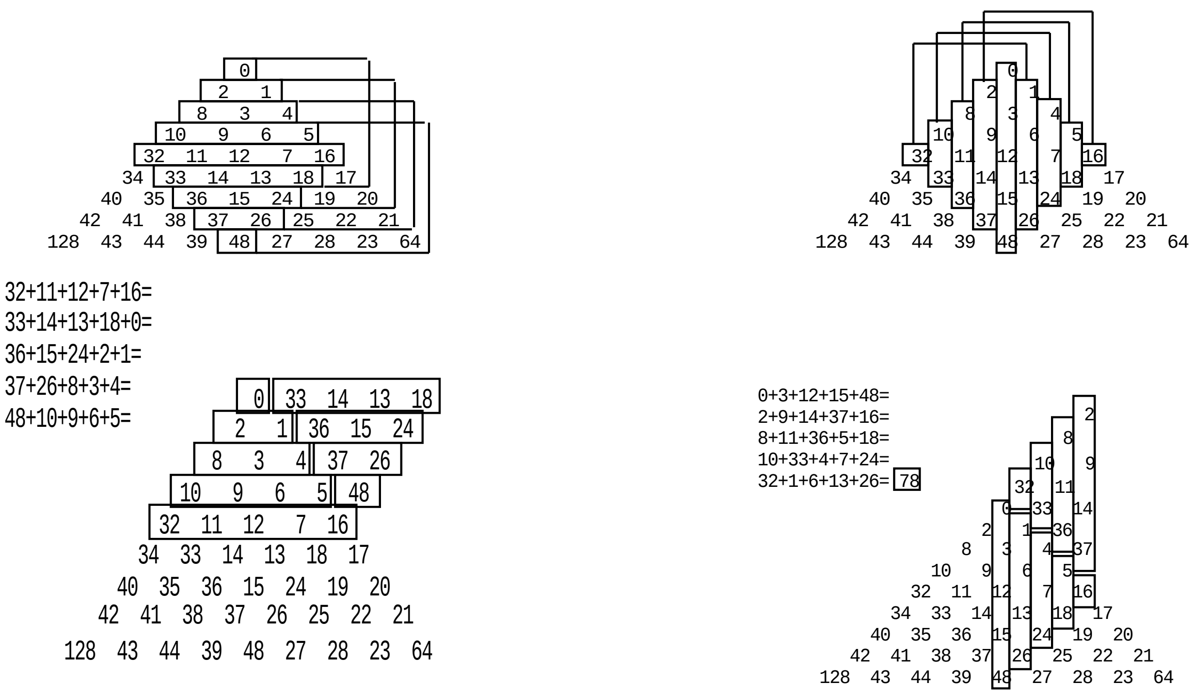


Fig. 13. The sum of diagonals or of 'complemented diagonals' within a square is constant. The bow (above) or the translation (below) indicate which number sequences add up to equal the sum of the main diagonals of the square.

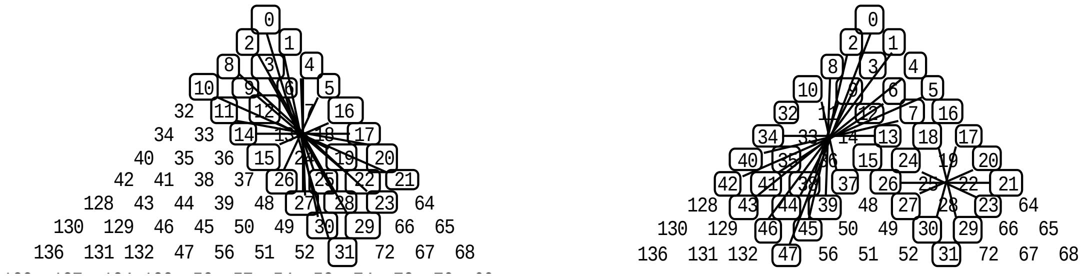


Fig. 14. The connecting lines between all number-pairs, which give the sum 31 intersect in one point (in the square [7+24=13+18]). If the sum is 47 they intersect in two points: [11+36=33+14] and [19+28=25+22]. (Obviously the patterns are related to powers of two minus one: 32-1, 32+16-1.)

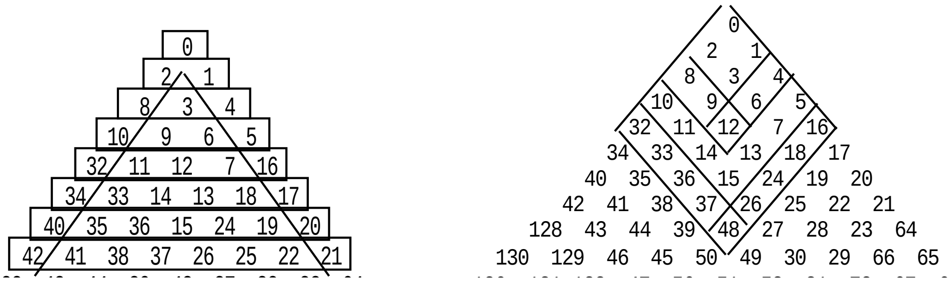


Fig. 15. Diagonals which are perpendicular to the main diagonal and start in the first column/ end in the first line we call 'base diagonal' or 'base' (of the triangle).

<u>The sum of numbers on a base is divisible by 3.</u>
<u>The sum of a base equals the next base's sum, minus the first and the last number.</u>
The first number is the last number times two; together they give three times 'something'. The sum of numbers between them is a multiple of three for the same reason: This sum is a multiple of three, since it is derived from the first base, which only consist of a first and a last number.
The sums of successive bases give the sequence: {3,15,30,78,129,189,252,444... }:
$\{(2^0)*3,(2^2+2^0)*3,(2^3+2^1)*3,(2^4+2^3+2^1)*3,\ldots\}$

The sum of the first and the last number of a 'base' can be found on the 0-diagonal: Since 0 is the top of the triangle to every base, we get the sum as a single number, at a position where it completes the triangle to a square. <u>The 0-diagonal connects only numbers which are divisible by three, because the sum of the first and the last number of a 'base' is divisible by three as well.</u> The numbers of the diagonal are exactly the numbers of the first line times three.

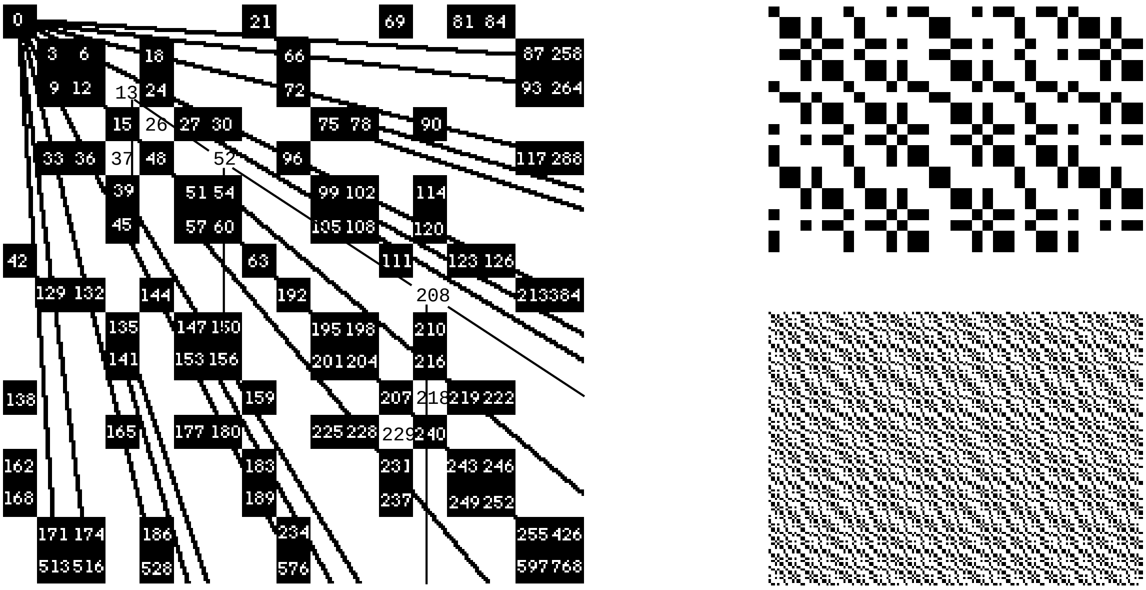


Fig 16. The distribution of numbers divisible by 3 is symmetrical relative to the main diagonal, the pattern is periodical. The two 'eyes' (26/37, 218/229 or 794/805) have always the number-difference of 11. There are several rays through 0 and numbers which are multiples of 3 (0, 9, 36, 45, 144, 153, 180, 189 ...).

We can speak of classes of numbers divisible by three:

Class one: Numbers divisible by three can be found for each line and column by multiplying the first number in that line or column by an odd power of two plus one.

$(2^{(2i-1)}+1)*n_{(x,0)}$, or $(2^{(2i-1)}+1)*n_{(0,y)}$, i{N}>0

All (and only) these multiples of three lie on particular rays through 0. Multiples of numbers in the first line are responsible for the main diagonal ($3*n_{(x,0)}$) and rays below this diagonal, e.g. $9*n_{(x,0)}$, $33*n_{(x,0)}$,...Multiples of numbers in the first column lie on rays above the main diagonal, e.g. $3*n_{(0,y)}$.

Class two: Other rays through 0 connect numbers, which are multiples of three of numbers in the same column but not in first position (e.g. 39,156,624..). Others are even more complicated. Finally, there are the multiples of three in the first line and column. Class two rays contain also numbers, which are not divisible by three.

Any two numbers, which have the relation $1:4^i$, lie on a ray through 0. i{N}>0

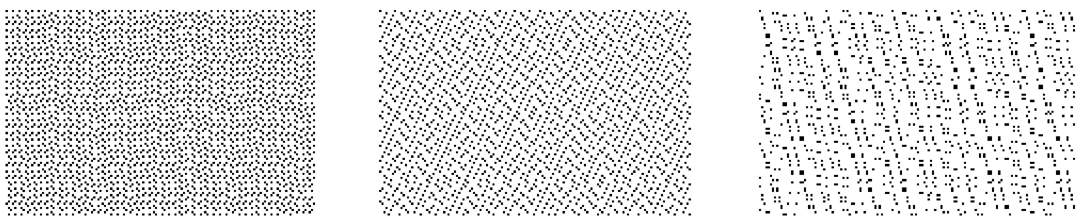


Fig. 17. The distribution of numbers dividable by 5, 7 and 11.

Some significant properties of this order of numbers have been described so far. Most of them are closely related, some just a different description of the same property.

The seed- cell of the system is the square 0 1 / 2 3 which already inheres the basic features.

The binary map can be closed to a torus, and the described features are consistent in this topology, at least, if borders connect each other at x- and y-positions, which are powers of two.

Binary image coding

In the following our binary representation system will be tested for coding images.
A pixel of an image can be represented by one number, according to our system. If there is a black 4 by 4 square in the image, it also can be expressed by one number, if the resolution is changed. The particular resolution, used to represent a pixel, may then be indicated by the number of digits. An image could thereby be resolved in pixels of different resolution, and represented by a list of (binary) numbers, the number of digits indicating the resolution the value the position. Except for images, consisting mainly of squares that fit into the resolution rasters, except for such images there seems to be no significant advantage of this codification.
But the system provides additional capabilities:

The numbers of every square number-field produce a sum, which -in binary notation-features a particular structure.

1. Squares[5] of the same size, say 4 by 4, always produce a sum, which shows the same pattern in the lower binary-digit-section.
2. There is a section, which we will call 'offset-marker', OS.
3. There is a higher binary-digit-section, which indicates the position of the square in the image, with respect to the resolution.

The binary pattern of the sum of a number square holds information of the square’s size and position. This does not only hold for black squares, squares with all positions occupied. If certain positions in the square are black, the bit-pattern of their sum allows to discern some graphical patterns: diagonals, crossed diagonals and lines:

position	OS	resolution, pattern, rastersize	
..000101	0	11	(diagonal / oder \ in square 2*2)
..000101	0	110	(square 2*2)
..000101	0	11110	(diagonal in square 4*4)
..000101	0	111100	(an X in square 4*4)
..000101	0	1111000	(square 4*4)
..000101	0	11111100	(diagonal in square 8*8)
..000101	0	111111000	(an X in square 8*8)
..000101	0	11111100000	(square 8*8)
0101	0	111111111111110000000000000	(square 128*128)
..000101	0	01 / 1 01	(1./2. horiz.line in square 2*2)
..000101	0	10 / 1 00	(1./2. vert.line in square 2*2)
..000101	0	10100 / 0 11000 / 1 00100 / 1 01000	(1./2./3./4. horiz.line in square 4*4)
..000101	0	01010 / 0 10010 / 1 01010 / 1 10010	(1./2./3./4. vert.line in square 4*4)

Fig. 18. Position denotes the field in a raster based on the resolution (the particular position number here is just an example). All possible diagonals within a square (6 in 4*4 squares, 10 in 8*8 squares) have the same bit-pattern. Therefore, they need to be discerned in a different way. Using 32 bit numbers this notation is able to store the 16 possible 128 by 128 squares of a 512 by 512 image.

[5] holds also for 1:2 rectangles

20 21
EXAMPLE: the number-square 22 23 has the sum 86, binary: '101 0 110'. The last three digits '110' indicate a 2 by 2 square, followed by the Offset-marker, and the number '101' indicating position 5 in a 2 by 2 raster. To achieve the position in 0-resolution (top-left of the square), divide the digits, concerning the position by 2^{res+1}, i.e. $(2^6+2^4)/2^2=2^4+2^2=20$.

Since we are dealing with squares only, we can change the convention for the resolution to res0= 1*1, res1=2*2,... A number val in resolution res then represents a:

square if val mod $2^{(3+res*4)} = 2^{(3+res*4)} - 2^{(1+res*2)}$,

diagonal if val mod $2^{(3+res*4)} = 2^{(2+res*3)} - 2^{(res)}$,

crossed diagonals if val mod $2^{(3+res*4)} = 2^{(3+res*3)} - 2^{(res+1)}$,

The position P = val div $2^{(3+res*4)}$

Remarks

<u>By means of summation of occupied positions within a square we gain a binary number, that contains information about the pattern in the square, the resolution (size) and its position.</u>

As a matter of fact this information is sometimes ambiguous, for example what the direction of a diagonal concerns. The offset marker indicates the offset of a square with respect to a square-size raster. Such an offset may also lead to ambiguities. To avoid ambiguities, squares which do not feature one of the privileged patterns have to be broken up in smaller squares, otherwise their sum-code may allow several possible interpretations.

It was intended to introduce some general features of this binary representation system. In order to use it for image processing or data compression, it evidently has to be mixed with other strategies, to compensate for several weak points.

A main characteristics of this system and certainly an advantage in many respects is the handling of the resolution variable: an image is described and treated on the level of different resolutions at the same time.

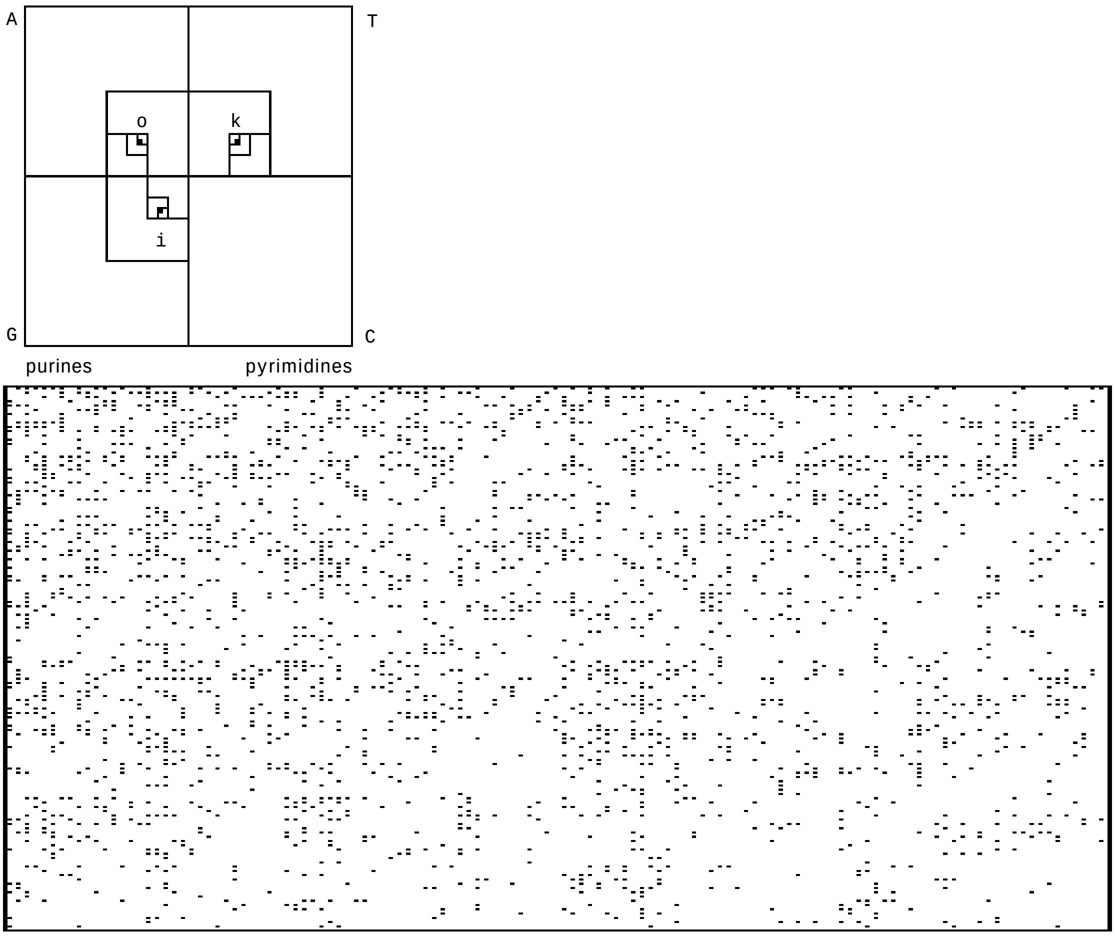
The set of basic image elements are akin to 'primitives' in perception theories. The structure of their codification is scaling invariant and consistent among different features, and still describes them in a simple and explicit way. Therefore the system could be specialized for feature detection tasks. Especially for didactic purposes it could provide a simple model for the automated 'symbolic representations' of images.

Binary Map applied to DNA representation

a brief sketch of an application

DNA sequences can be represented by a variation of this system. Since a DNA sequence is written by the succession of four possible nucleotides (A, T, C, G), for every next symbol the frame is divided into four partitions. Thereby the succession of symbols produces a dot. The complementary sequence (which is equivalent to the second strand of the DNA) corresponds to a symmetrical position relative to the vertical axis.

A position, hit twice, indicates a repeated succession, which may be of interest in the analysis of the sequence. Since the storage of multiple hits would unnecessarily complicate the system, we can produce discrete images of the succession of partly repeated sequences in choosing a resolution higher than the length of repeated strings.



top: The purines A and G occupy the left, the pyrimidines T and C the right half. A sequence 'ACGTTG' produces the dot 'o', the complementary sequence 'TGCAAC' corresponds to position 'k', 'i' to the inverse sequence 'GTTGCA'.

bottom: Distribution of 7 nucleotides long 'words' (Clostridium, SL-gene). To compare longer sequences, an elaborated 'zooming mechanism' will be useful.