APPLICATION OF RANDOM CODES
TO THE GATHERING OF STATISTICAL INFORMATION
by
CALVIN NORTHUP MOEHS
B.A. summa cum laude, University of Minnesota
(1941)

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
(1948)

Signature of Author

Certified by

Chairman, Departmental Committee on Graduate Students

Thesis supervisor
APPLICATION OF RANDOM CODES TO THE GATHERING OF STATISTICAL INFORMATION

by Calvin N. Mooers
(January 1948)

I - Introduction

In September 1947 at the 112th Convention of the American Chemical Society, the author presented a paper describing a novel use of random codes, called Zatocoding, which is usable in the mechanical selection of index cards for bibliographic and other uses. Since this paper has not yet been published, it is included here in its entirety as an appendix, and will be referred to as Z. For the selection of index cards, Zatocoding was found to have unusual advantages in power and versatility over previous systems of coding information. This led to the speculation that essentially the same random code methods could contribute similar advantages to the coding of information where the counting of cards or other records is used for the gathering of statistical information. To assess the speculation, the validating conditions for the use of the new coding method in statistical counting were set up. The conclusion follows that Zatocoding has much to offer, especially in instances where more conventional methods have broken down.

The United States Census is a notable example of the use of punched cards for the gathering of statistical information. For each person enumerated in the census, there is made up a set of four standard IBM punched cards. Four cards are necessary to hold the variety of information such as age, sex, occupation, marital status, etc. that may be applicable to each person. This information is recorded on the cards
by means of patterns of meaningful punches. Upon running a pack of such cards through a suitable machine, the punches are sensed, and counts and correlations of the information on the cards are gathered. In my terminology, the number of doctors in the pack is counted, and the number of female doctors is a correlation or count on multiple coincident facts. An example of the significant practical results to be developed in this paper is the possibility of using perhaps one or two cards per person in the Census application, instead of the four cards now needed.

Aside from the application to the Census, the present method presents a new and potentially powerful tool for the recording and analysis of scientific data having a great inherent complexity. In sociology, a complete case history can be recorded on the punchings in an IBM card. Collected case histories can then be analyzed for the occurrence and significance of correlations between factors. This is in general impossible with conventional coding systems due to the limitations described in Z. In the field of physiology and medicine, such as in cancer research, complete experiments can be summarized on a card, and these cards, as assembled in a collection, can be used to establish trends and correlations that would otherwise escape detection. Limited work in this direction by a medical scientist is already going on.

It is evident, in any system based upon the recording of intelligence into a pattern of punches on a card, that the translation from intelligence to punches -- the coding -- is most important. Therefore it is surprising how little systematic study has been given to the matter. The Zatocoding method of random codes described in Z is a departure from previous thought on coding. In that paper, in order to
establish the worth and utility of Zatocoding for the punched card
selective process, a mathematical treatment of conventional coding
systems and of Zatocoding was developed. In brief the conclusions of
Z are that Zatocoding, on a card of the same size, provides an enormously
greater effective vocabulary of coded ideas, and allows these coded ideas
to be used with much greater freedom than with conventional codes. On
the same card, Zatocoding may increase the effectiveness of coding
billions of times in terms of ability to handle a wide range of
complex information.

These advantages of Zatocoding, namely the greatly enhanced
ability to code information and the freedom of use of coding vocabulary,
are not gained without certain disadvantages or strictures inherent in
the new manner of coding. Aside from the need of acquiring a new
philosophy of description of ideas appropriate to its enlarged scope, there
is the more practical situation that derives from the statistical --
rather than exact -- operation of Zatocoding selection. The relation
between the ideas associated with a card and the final pattern of punches
placed in the card is not one-to-one. Thus, as explained in Z,
selection of a set of cards on the basis of a pattern associated with a
desired set of ideas will produce all those cards carrying this set of
ideas, but it will also in general produce an additional set of cards
carrying the same pattern, but with some other fortuitous and totally
unrelated set of ideas. Fortunately, the appearance of these extra
cards is a statistical phenomenon, and is under control by the original
design of the system and by the design of the manner of selection.
In the application to selection of index cards, as described in \( Z \), the appearance of a reasonable number of extra cards is of little moment, for they can be discarded by the user of the punched card file by inspection. However, in the statistical application of concern to this paper, the appearance of an unknown and variable number of extra counts is at least undesirable. It is impossible to reject the extra cards, since the only criterion of counting is the occurrence of the given pattern of punches, which may not be unique.

In the face of error, there are two avenues that may be developed. Attempt can be made to diminish the magnitude of the error by whatever practicable means there are available, e.g. refinements or compensation. Second, an estimate can be placed on the size of the incurred but unknown error in order that unjustified faith not be placed in a determination. In what follows, it will be shown how refinements can be developed in the Zatocoding method to markedly decrease the range of error, and how confidence limits can be placed upon the magnitude of the error for any given situation.

The results that have here been attained are the first and easy reward in a virgin field. While rough-and-ready, the results are powerful enough to give evidence of the existence of a new realm of investigation: of interest both mathematically as a consequence of their new orientation with respect to a simple set of postulates, and practically as a consequence of their intensely useful material results. This paper states the problem, and gives a rough though complete engineering solution. On this beginning, work in the refinements of the ideas and procedures can be founded.
II - Definitions and Symbols

Definitions are given both in their generality, and with application to census punched cards for definiteness. An event is a single situation or occurrence, such as a census interrogation of one person. An event may also be the circumstances and result of a scientific experiment, or a case history of a person. The important facts surrounding an event may be described, and the description is in terms of a set of intellectual components. We shall assume that it is possible to find these components for an event, and that the number of components is finite. To each event is assigned one unit of the medium, e.g. one punched card. Each unit has an indicia-defining coordinate scheme which specifies the positions at which the indicia or punches may be placed. The set of all positions is called the field. Associated with a type of medium and indicia, there is a selector mechanism able to sense the presence of the indicia, and on the basis of some criterion of the pattern of a unit as sensed, to perform some action such as to count.

To record the intellectual components of an event on a unit of the medium, the intellectual content of the separate components is translated into a pattern of indicia by the process known as coding. The different modes of coding are described in detail in Z. By the conventional methods of coding, the field is divided into subfields, and the code of only one component is placed in any one subfield. The unit may carry as many intellectual components as there are subfields. These subfields may range in size from a single position to the whole field of the unit, and the codes may be any pattern of indicia chosen from the positions of its associated subfield. The particular division of subfields, once chosen, is maintained throughout the whole collection of
units in the conventional systems of coding. In the most general form of Zatocoding, the field is not subdivided, and the codes consist of randomly-generated patterns taken from over the whole field. The several patterns of the intellectual components delineating an event are superimposed upon the same field of a unit. The statistical properties of the random codes are utilized to prevent confusion.

**Selection** of a unit according to conventional coding systems occurs when the pattern of indicia in a subfield matches the pattern of the selector. Selection may be on more than one subfield at the same time, but that is a simple extension of the above. Selection according to Zatocoding occurs when the pattern of indicia in the field of a unit includes the pattern of the selector. The same criterion prevails when the selector pattern is the sum of two patterns: then the selection is on the basis of the logical product of the two intellectual components. (Note here how the point-sum of the indicia patterns leads to a logical product of ideas in selection. This interesting complementary relation can be developed into a system involving the several possibilities of the indicia-treatment and the mechanics of selection. Such investigation has shown that the system employed in Zatocoding is the richest and most meaningful.)

Assign the following symbols:

- A - the collection of units
- F - the field of a unit
- N - a single pattern associated with an intellectual component
- Y - a pattern of indicia on the field of a unit
- S - the pattern in the selector
E - an event
M - an intellectual component
f - a subfield of F

Assign the following operators:

v - the number of indicia or positions of, counting in the
direction of the field, e.g. vN is the number of
indicia in pattern N.

w - the number of, counting in the direction of the
set A, e.g. wA is the number of units in the set A.

d - the distribution of, e.g. dvY is the distribution of
pattern lengths in all the fields.

r - the number of times some pattern is recorded, e.g.
rN is the number of times pattern N is recorded in
collection A.

s - the number of selections according to a pattern, e.g.
sN is the number of times a selection on pattern
N is made in collection A. In Zatocoding, sN may
be larger than rN.

i - the intersection of, e.g. the indicia common to patterns
N_1 and N_2 is \( N_1 \cap N_2 \)

u - the union of, e.g. \( N_1 \cup N_2 \) as above

b - the space or set of, e.g. bM is the set of all
intellectual components M.

m - a mapping of, e.g. mE into M is a mapping of events into
intellectual components.

In making use of this symbolism in what follows, an attempt will be made
to remind the reader of the meaning of the symbol whenever it is again
newly introduced.
III - Postulational or Abstract Development

It is possible to pursue the consequences of the Zatocoding method by an abstract or postulational development. While this course will not be followed in this paper, a short discussion of some of the abstract mathematical features of conventional coding and Zatocoding will be presented in order to give an insight into their operation as abstracted from their physical embodiments. The postulates are given for their suggestive value, the system has yet to be made precise.

Postulate 1. Any point $E$ in the event space $bE$ can be mapped into a vector $(M_1 \ M_2 \ \ldots \ M_n)$ of a finite number of components which are taken from the intellectual-component-space $bM$, and in such a way that this mapping is useful.

The "useful" here is not further defined. What is meant is that in a practical situation, a finite number of components delineates sufficiently the content of an event.

Postulate 2. Associated with each component $M$ there is a pattern $N$ contained in the field $F$, that is $N$ is given by $mM$ and $N$ is included in $F$.

Postulate 3. If the coding is by the conventional system, $m$ is one-to-one, and therefore has a unique inverse.

Postulate 4. With Zatocoding the inverse of $m$ may not be unique.

Thus with a finite size of field $vF$, with conventional coding there are a finite number of patterns, and thus a finite and limited number of components. With Zatocoding there is no such limitation imposed by one-to-oneness, and the number of descriptive intellectual components $M$ may be unlimited.
Postulate 5. (Selection by conventional coding) Selection of a unit occurs when a vector such as \((N_1, 0, N_3, \ldots, N_n)\) of the selector matches the pattern of the unit, except for any vector components marked 0, which components are inactive in the selection.

Note that the field \(F\) is divided into the subfields \(f_1, f_2, \ldots, f_n\) to correspond to the components of the vector.

Postulate 6. (Selection by Zatocoding) Selection of a unit occurs when the pattern \(S = N_1 u N_2 u \ldots u N_k\) in the selector is included in the pattern \(Y\) of a unit.

This selection corresponds to finding the logical product \(M_1 i M_2 i \ldots i M_n\) in the event of a unit.

Postulate 7. (Conventional coding) The number of points in the \(N\)-vector space is limited to \(2^vF\), where \(vF\) is the size of the field of a unit. Also, the components \(M\) are partitioned among the subsets \(f\) in such a way that \(M_{ij}\) is associated with subset \(f_i\) and \(j\) runs from 1 to \(2^vF\).

The details of this are given in reference 2. This postulate places a stringent limitation on the descriptive power of a conventional system of coding. The restriction does not exist with Zatocoding.

Postulate 8. (Zatocoding) For each intellectual component or point \(M\), a pattern length \(n\) is decided upon, and then by some random process a pattern \(N\) is assigned such that \(vN = n\). This assignment is thereafter retained. This is done for all points \(M\) as they are used.

Thus any two patterns \(N_i\) and \(N_j\), corresponding to the points \(M_i\) and \(M_j\) which may be meaningfully related, bear no more than a statistical relationship with respect
to any of their pattern properties, e.g. the distribution of the number of identical indicia or \( dv(N_i, N_j) \) is statistically determined.

Consider now the most interesting process, namely that of coding. **First, with conventional coding:** Given a point \( E \) in event space, a finite set of points \( M \) is chosen to delineate the intelligence of \( E \) by an \( M \)-vector. The set \( M \) has many restrictions. No two \( M \) in the set can be associated with the same subfield on one unit. The number of \( M \) in the descriptive delineation of \( E \), i.e. the components of the \( M \)-vector, is limited by the number of subfields \( f_i \) in \( F \). The vector space of \( (M_1, \ldots, M_n) \) has only a finite number of possible points, namely \( 2^vF \). From this vector space there is a one-to-one transformation to the final pattern \( Y \) on the field of a unit. Notice that the weak link is in the transformation from the point \( E \) in the event space to the point \( (M_1, \ldots, M_n) \) in the vector space. If the descriptive nature of the possible \( M \)'s is unsuitable for the meaning of \( E \), then this transformation is misleading if attempted at all, and it may be entirely impossible to effect such a transformation. This difficulty in coding is matched by other logical difficulties in selection which are discussed in \( Z \) and will not be covered here.

**Second, with Zatocoding:** Given a point in event space, a finite set of points \( M \) is chosen to delineate the intelligence of \( E \). The space of \( M \) is unlimited in size. If there is intelligence in \( E \) that is not suitably covered by any \( M \) in the space, an additional point may be added to the space to represent this meaning (and at the same time generating a random code \( N \) to go with it). Thus no relevant meaning can elude the transformation from \( E \) into the \( M \)-vector space. The transformation from
the $M$-vector space to the pattern placed on $F$ is by the superposition of the patterns of the respective $N$'s, i.e. the pattern in $F$ is $Y = N_1 \cup N_2 \cup \ldots \cup N_n$. While the number of different $Y$'s is finite, the number of different vectors representable is unlimited. This leads to lack of one-to-oneness, and is the weak spot in Zatocoding. However, the weakness is under statistical control and there is no impossible situation to correspond with the unsatisfactory initial transformation from $E$ to $M$-vector of the conventional coding.

It is clear that the content of the paper $Z$ could be more concisely described in this language. Also, in the same manner, the consequences of the postulates can be set forth in a formal fashion. However, because this paper does not represent a finished investigation of the material, and because of the assistance in thinking in terms of concrete applications, an abstract and formal development will not be followed. Wherever the above symbolism is useful, it will be used, but the thinking will proceed in whatever manner it is best handled and described.

IV - Formulation of the Problem

In a collection $A$ of size $wA$, assume for simplicity that the distribution of lengths of patterns $Y$ in the fields of all units of $A$, i.e. $d_Y$, is 1 when $\gamma_Y = \frac{1}{2}v_F$ and 0 otherwise. Computations on this basis will then provide an upper limit for the actual situation in which $d_Y$ has an appreciable value for $\gamma_Y$ less than $\frac{1}{2}v_F$.

Then as shown in $Z$, a selection and count on a nonsense pattern $S$ will provide an average count of magnitude $\left(wA(1/2)^{VS} \right)$. This count is equivalent to the 'extra' cards of $Z$. Here we refer to this count as the
noise level, and study the properties of this noise in greater detail. Certainly then, in any count of the collection A according to some meaningful pattern S, the legitimate count is going to be increased by this noise level. No legitimate count can be missed, but spurious counts will be added. Since the counting and selection is only on the basis of the pattern Y in the field, these extra counts cannot be thrown out by the kind of inspection possible in the bibliographic use of Zatocoding as described in Z. The noise must be accepted as part of the price to be paid for such advantages of Zatocoding, such as that of there always being a meaningful transformation from a point E to the final pattern Y. The matter of importance is whether we can control this noise level in such a manner that sufficiently accurate statistical information about the intellectual content of the set A may be derived by a simple counting process.

Besides the simple random noise, we must consider the more difficult problem of the augmentation of the noise due to the interference of code patterns. Let pattern N₁ be on 50% of the units in the collection A. Then if a count is made upon pattern N₂, and if N₂ has several indicia in common with N₁, it is evident that these indicia are not effective in discriminating against the random noise on those units possessing pattern N₁. Thus if there are three indicia in common, the noise will be eight times greater during half the run, due to the ineffectiveness of this part of the selective pattern N₂. Such enhanced error might be serious. Therefore we must determine how often an intersection of patterns such as this may occur, how the intersection can be controlled, how the resulting noise can be estimated and bounded, and whether some limitation on the maximum frequency of patterns such as N₁ can be set.
The simple noise level and the augmentation of noise due to interfering patterns are controllable — if at all — in the design of the system A and its coding. However, if nothing is known about the system A or its content, there should be some procedure by which the expected severity of these two effects may be assessed. That is the final problem.

V - Initial Attempts at Solution

The description of the early attempts at a solution of these problems is included because this field of investigation is very new, and therefore a reformulation of some of the ideas may produce a useful advance at a later stage.

First thought: If S is the selective pattern and if N is the interfering pattern appearing on 100% of the set A, and if S and N were randomly related, the expected effective number of indicia of the selective pattern S would be \( (\nu S)(1 - \nu N/\nu F) \). If N appeared less frequently, this effective length of S can be used for part of A, and the whole of S for the rest of A in determining the noise level. This method is developed in greater detail later, using a different statistical approach.

Second thought: Consider the evaluation of an unknown set A. Perform a selection and count on the basis of all possible patterns of length \( \nu S \), summing all the counts. The first speculation was that the existence of a highly-frequent pattern would cause the total count to increase over the count expected from a totally random set of units. The deviation of the count would then be a measure of the lack of randomness, and thus be indicative of the effects that might be anticipated.
from the interference of codes. It would be possible to give the situation a kind of 'entropy' interpretation, with the ideal or random state providing an extremum in the total count. At first it was thought that the random state count would be the minimum.

Later investigation disclosed that the deviation in total count operated in the other direction, which seems surprising at first when one thinks only of the contribution to the count provided by the repeated pattern. However the effect is due to the depletion of indicia in the rest of the field not covered by the repetitive pattern, and because the random count in this portion is dependent upon an exponential of the indicia density. Thus the count drops off faster in this part of the field than it is picked up in the repetitive pattern. These conclusions apply when \( v_S \) is larger than unity. When \( v_S \) is unity, then the total count is invariant to the repetition of pattern, so long as the average field density is held constant.

**Example 1.** Case \( v_F = 8, v_Y = 4 \) (50% density), and \( v_S = 1 \).

**Random pattern:** The total count for a set of size \( w_A \) is given by

\[
(w_A)(\text{density of indicia})(\text{number of patterns } 3) = w_A(1/2)8 = 4w_A
\]

**Repetitive pattern:** The repetitive pattern consists of indicia at positions 1 and 2 of \( F \) for all \( A \).

A) Counts on positions 1 and 2 = 2\( w_A \)

B) Counts on positions 3 to 8

\[
(\text{\( w_A \)})(\text{density})(\text{patterns}) = w_A(2/6)6 = 2w_A
\]

Total of A) and B) is 4\( w_A \), or the same as with the random distribution.
Example 2. Case \( v_F = 8, \) \( v_Y = 4, \) and \( v_S = 2. \)

Random: total counts \( = wA(1/2)^2(8 \cdot 7/2) = 7wA \)

Repetitive pattern: indicia at positions 1 and 2 for all \( A. \)

A) Counts with \( S = 1,2: \) \( wA(1)^2(1) = wA \)

B) Counts with one element of \( S \) on either 1 or 2:

\( wA(1)(2/6)(2 \cdot 6) = 4wA \)

C) Counts with no element of \( S \) on either 1 or 2:

\( wA(2/6)^2(6 \cdot 5/2) = 15/9 \ wA = 1.67wA \)

Total of A), B), and C) is 6.67wA, and is slightly lower than 7wA for the random case.

With these observations, the method of entropy will be dropped, though at some later date its complexity and richness may provide a useful tool in the evaluation of collections such as the space \( A. \)

Third thought: Random patterns can be selected from the set of all random patterns in such a way that the intersection of any two patterns is held below a certain specified number of indicia. With a large field, the loss in number of possible patterns is of little moment, since an enormous number will remain. Certainly the interference of any two patterns will not then be serious. However, this requirement upon the patterns will upset the statistics governing the appearance of extra selections, and may actually cause an augmented interference in the case of multiple interfering patterns.

While this third speculation was sterile in itself, it did lead to a most important realization, namely that 1) the random noise depends only upon the length of a single pattern, while 2) the possibility that two patterns of the same length do intersect depends inversely upon the
size of the field. Thus intersection can be controlled by increasing the
size of the field, and without placing any restrictions upon the random
nature of the patterns. This possibility, for one, is developed and
assessed in the following paper.

VI - The Noise Level Problem

Now, after this varied introduction to the problem of gathering
statistical information with Zatocoding, and the mathematical reformu-
lation of some of the descriptive content of Z, we are prepared to
consider the results of the present investigation.

The noise level due to the statistical selection of undesired
units is considered first. If a pattern N is recorded in the set A
rN times, it will be found, upon selection, that the number of units
of A that will be selected by the pattern S - N is sN, where sN must be
greater than rN. That is, every unit with N purposely recorded upon it
will be counted, and there are rN of these, but in addition there will
be counts due to the noise level, thus making the actual count sN
greater than rN. The addition due to the noise level is sN - rN, and
the ratio of the additive error to the recorded number rN is (sN - rN)/rN,
and will be called R.

The set A has wA units, and the acquisition of the noise level
counts will be only among those members that do not have the pattern N
intentionally recorded upon them, that is, only upon wA - rN of the
units. Assuming that, except for the single pattern N, the indicia
are placed upon the fields of the units in a completely random fashion
and at a density of 50% as suggested in Z, then the probability of
selection of any one unit not having N recorded thereon is (1/2)^wA x p.
Thus the expected noise level additional count is \((wA - rN)(1/2)^{YN}\).

In the statistical sense, this value is an arithmetic mean, and there will be fluctuations in actual noise level count about this mean, and the extent of these fluctuations must be considered before any reliance can be placed in judgements based upon a value predicted by the mean.

At this point we can fortunately make use of two properties of the normal distribution curve: 1) the area under one tail for deviations greater than 3 sigma is about .00135, or less than one part in 700; and 2) the binomial distribution can be fitted by a normal distribution by setting the mean equal to np and the standard deviation equal to \((npq)^{1/2}\), where \(q = 1 - p\).

In the case at hand, \(n = (wA - rN)\), \(p = (1/2)^{YN}\), and since in most cases \(VN\) is greater than four or five, we can approximate \(q\) by unity. The mean number of noise level counts is then \((wA - rN)(1/2)^{YN}\) as calculated above, and the standard deviation in the number of counts is simply the square root of the mean, since we have set \(q = 1\).

With this, we can say that in performing a count, the chances are 700 to 1 that the ratio of error \(R = (sN - rN)/rN\) does not exceed

\[
R = \frac{(wA - rN)(1/2)^{YN}}{3(wA - rN)^{1/2}(1/2)^{YN/2}}
\]

Since we are interested only in the maximum error, we need consider only one tail of the distribution, as shown in the diagram:

![Diagram of normal distribution]

- Frequency
- One tail
- Value of \(R\)
- Mean
- 3 sigma
In many cases, rN is negligible compared to wA, and can be neglected in the numerator of the equation. Thus with a confidence of 700 to 1, assuming a random field, we can set the maximum value of R equal to this expression. With this confidence, we have a relation bounding the error which involves the quantities: error ratio R, pattern length vN, size of collection wA, and the number of units recorded with pattern N i.e. rN.

Example 3. In the U.S. census there are probably \(10^9\) events tabulated, or \(wA = 10^9\). If we assume that some component is tabulated only \(10^3\) times, and that an accuracy on a later count must be \(1\%\) or \(R = .01\), what is the pattern length vN that must be used to insure this?

\[
(10^9)(1/2)^{vN} \neq 3(10^9)(1/2)^{vN/2} = (10^3)(10^{-2}) = 10
\]

With the right hand member greater than unity, the first term dominates, and recalling that \(10^9 = 2^{30}\), we see that vN must be in the order of 27, and refining the calculation, we find vN = 28 just satisfies the equation.

\[
4 \neq 3.2 \times 10
\]

We conclude that when confronted only with random noise level error, a code pattern of 28 indicia is sufficient to carry the intelligence.

Example 4. In the same situation, if we require that the later count be absolutely accurate, we can set \(RrN = 1/2\), i.e. \(R = .0005\), and we arrive at the conclusion vN = 36, since this vN gives for the right-hand member \(1/54 \neq 3(1/8)\) which is less than 1/2. If we had not used the conservative margin of \(RrN = 1/2\), but \(1\) instead, vN = 34 would have sufficed. Observe that when RrN is less than unity, the standard deviation term dominates.
Example 5. Assume now in the example of \( wA = 10^9 \) with a different \( N \) that \( rN = 0.50wa \) and that a maximum \( R \) of \( 10^{-3} \) is tolerable. What is the required \( vN? \)

\[
(2^{30})(1/2)^{vN} \neq 3(2^{15})(1/2)^{vN/2} = (1/2)(2^{30})(2^{-10}) = 2^{19}
\]

The first term is dominant, and we see \( vN = 11 \) is satisfactory. However, for reasons demonstrated in the following example, a longer pattern may be desirable.

Example 6. With \( wA = 10^9 \) and with two code patterns \( K_1 \) and \( N_2 \) of length 11, how large must be the number of units bearing both codes in order that a non-null set of the intersection of these two ideas may be inferred? As a practical local example, how long must the patterns be for 'mayor' and 'convict' in order that a count of a given size becomes significant in indicating that there are mayors in jail. With the two patterns of length 11, the combined pattern is of length 22, and gives a mean noise level count of \( 2^8 \) and a sigma of \( 2^4 \). From these we conclude that the count must exceed 304 to indicate a non-null set with a 700 to 1 confidence. In this instance, the short patterns are not very satisfactory. In anticipation of such need to examine for nearly null sets, it would we wise to increase the minimum pattern length from 11 to 15 or 16. With a 15 indicia pattern, a count is significantly non-null at 4, and with a 16 indicia pattern at 2.

These examples illustrate fully the most important cases and conclusions that may be drawn from a study of the random noise level, as it relates to the length of the pattern.
VII - The Interference of Patterns

Now we shall evaluate and deal with the problem presented by the interference in the selective ability of one pattern by other patterns of high frequency which may have indicia in common with the selective pattern. Call the selective pattern $N_1$ and the interfering pattern $N_2$.

So consider a high frequency of the interfering pattern, for if there is a multitude of interfering patterns each of a low frequency, i.e. with $rN/wA$ much less than unity, then we have what is essentially a random set of patterns in the field, and there is nothing to worry about.

The question then is, how frequent must an interfering pattern be in order to upset the selective process? More particularly, what must be the ratio $rN_2/wA$ for a two-fold increase in noise level in the count when there is an overlap of patterns on $n$ indicia? Over $rN_2$ units of $A$ there is an increase in noise by a factor of $2^N$. Thus we have the equation

$$\left( \frac{2^N rN_2}{wA} \right)^P = \frac{wA - rN_2}{wA}$$

where $P$ is the expected noise without interference, i.e. $P = wA(1/2)^{VN_1}$.

This gives $rN_2/wA = 1/(2^N - 1)$.

A tabulation of the first few values reveals:

<table>
<thead>
<tr>
<th>$n$</th>
<th>$rN_2/wA$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1/3</td>
</tr>
<tr>
<td>3</td>
<td>1/7</td>
</tr>
<tr>
<td>4</td>
<td>1/15</td>
</tr>
</tbody>
</table>

and for larger values, the approximation $1/2^n$ will serve.
If we ask when the expected noise with interference will increase by 3 sigma above the mean noise level without the interfering pattern, we get the relation

\[ rN_2/wA = 3/(2^R - 1)Q^{1/2}, \]

where \( Q = wA(1/2)^{\nu N_1} \) and is the mean noise level without interference. Because in many applications \( Q \) is allowed to exceed unity, this condition is more stringent than the last upon the frequency of the interfering patterns. The condition also depends upon the length of the pattern \( N_1 \). If \( Q \) is held near unity, e.g. to insure an accurate count, then the results in the table apply approximately in the 3-sigma criterion.

From this we conclude that both the pattern overlapping, and the frequency ratio \( rN_2/wA \) must be rather great to upset markedly the noise level in some cases. Upon units of \( A \) that have a significant amount of descriptive dispersion, a ratio of 1/10 would be remarkably high. Viewing the results another way, these formulas indicate when a pattern merges into the background of incoherent random patterns as far as interference in selection is concerned. Here the 3-sigma test is particularly good, and it is worth observing how it depends upon the length of the selecting pattern. This dropping to the random noise level is an important factor in the refined engineering of a statistical system using Zatocoding.

The number of overlapping indicia \( n \) has yet to be determined for any two patterns \( N_1 \) and \( N_2 \). Here, as always, our important starting point is the fact that the two patterns are randomly related to each other. \( N_1 \) has some fixed though unknown distribution over the field \( F \). \( N_2 \) is similarly in some fixed distribution over \( F \). The lengths of the patterns are \( \nu N_1 \) and \( \nu N_2 \) and the size of the field is \( \nu F \). The probability \( p \)
of a coincidence between indicia of the two patterns upon any given position in the field $F$ is

$$\frac{\nu N_1}{vF} \frac{\nu N_2}{vF} = p.$$  

Setting $q = 1 - p$, we can compute the probability of a coincidence of $n$ indicia in the whole field $F$ by taking the $n$th term of the expansion of $(q + p)^{vF}$, and thus the probability of $n$ coincidences is $vF C_n q^{vF - np}$.  

A more useful specification is the maximum number of coincident indicia to be anticipated. Here we can again use the normal curve approximation, with $p$ as above, and $q = 1 - p$, and with $vF$ as the number of cases. Because in general $p$ is very much smaller than 1, we can approximate $q$ by unity. It is found then that the mean number of coincidences is $vN_1 vN_2 / vF$ and the standard deviation $\sigma$ is the square root of this. As before, the chances are 700 to 1 that for any two patterns the number of coincidences will not exceed the mean plus 3 $\sigma$. We may take this as a working upper bound on $n$, and take

$$n = \frac{vN_1 vN_2}{vF} \neq 3(vN_1 vN_2 / vF)^{1/2}.$$  

Observe the rather remarkable fact that this is the first instance in which the size of the field $vF$ actually enters into the statistics of the selective process. Hitherto, and in $Z$, $vF$ was only of importance in determining the number of patterns that could be placed upon a single unit. A series of examples will exhibit more important conclusions that can now be drawn.

**Example 7.** Consider again the U.S. census application with $WA = 10^9$ with IBM cards on which $vF$ is approximately $10^3$. (Note: As presently constituted IBM equipment cannot select efficiently by Zatocoding principles, however the card is an excellent
medium.) With code lengths of 30, what conclusions can be drawn with respect to interference of codes? Computing the limiting n, we have approximately
\[ n = \frac{10^3}{10^5} \neq 3(\frac{10^3}{10^5})^{1/2} = 1 \neq 3 = 4. \]
The average overlap is only one, and the maximum expected is four. We can now find the ratio \( r N_2/wA \) at which the interfering pattern \( N_2 \) affects the noise level of selection on \( N_1 \) by less than 3 sigma. \( Q = 1. \) Then
\[ r N_2/wA = 3/(2^4 - 1)(1) = 3/7. \]
It is remarkable that the ratio can be this high, especially since a pattern of this length has an inherent accuracy that gives an error of only a few counts in the whole collection.

**Example 8.** The same as the above, but with code lengths of 15. Those shorter patterns will in general be used on intellectual components having a high frequency, and therefore the interference noise is much less critical, because it is always smaller than a very small fraction of the count. For this reason, we can find the point at which the noise level increases by two. We find \( n = 0.225 \neq 3(0.475) = 1.6 \) or 2. Then \( r N_2/wA = 1/(2^2 - 1) = 1/3. \) Because the noise requirements of a very frequent pattern are lax, such a pattern could well appear on all the units with no serious consequences.

While these examples have shown how the accuracy can be determined in a given situation, either of the examples could also be set up with a specification of the pattern length and accuracy, leaving the size of the field \( V/F \) to be determined.
Up to this point the interference of only a single pattern has been considered. To complete the study, the effect of as many interfering patterns as the field can hold should be assessed. From equation \( \text{Z} \) we have the sum of the separate patterns, i.e. the sum of the \( vN_i \), equal to \( 0.69vF \) as limiting the content of any one field. By the same token, this relation specifies the number of patterns that may be placed in the field. We shall assume that the distribution of any two patterns is independent over \( A \) (this is not true in an actual case, but it taken to furnish a basis for computation). We shall simply add the separate noise level counts from the several interfering patterns. (Though the differential equation of \( Z \) applies here, in the present situation the addition is nearly linear due to the infrequency or low probability of the additional counts.) If there are \( k \) patterns allowed in the field, \( k = 0.69vF/vN \). A typical value of \( k \) is 45 for \( vN = 15 \) and \( vF = 1000 \). We use the maximum intersection value \( n \) given above. Then looking back at the computation of p. 21 concerning the noise level increase, we find that \( rN/wA \) of an interfering code is directly proportional to the increment in the noise level. For a set of interfering patterns of the same length, and an expected increase of \( 3 \) sigma, 

\[
\frac{\text{summation of } rN_i/wA}{k} = 3/(2^n - 1)\zeta^{1/2},
\]

but there are \( k \) patterns on a unit, hence the average

\[
rN/wA = \frac{\text{summation}}{k} = \frac{3vN}{0.69vF(2^n - 1)}\zeta^{1/2}.
\]

By setting \( \zeta \) to unity, and dividing by 3, we have the relation for the two-fold increase. Here we can work simply in the average of the frequencies of the patterns \( N \). That is, counting \( rN_i \) over \( A \), and then averaging for any set of \( k \) patterns, the average of \( rN_i/wA \) for this set should not exceed the above specified value. The variations in single \( rN_i/wA \) may be considerable, but no \( rN_i/wA \) should exceed the limit specified for a single pattern.
To recapitulate the new material of this section, we now have formulas that place upper bounds, limits, and probabilities upon all the vital parameters interning into the problems of the interference between patterns. Tied into functional relationship are $rN_1/wA$, $wA$, $v_F$, and $vN_1$. Except for the one problem of what to do when the limit on $rN_1/wA$ must be exceeded because of the nature of the practical problem (e.g. in the census, $rN_1/wA$ for $N_1$ representing 'male' is near 0.5), the problem of the conditions of validity for the use of Zatocoding in the gathering of statistical information is solved. This one exception will now be treated.

VIII - Use of Conventional Coding for High-frequency Components

As noted in one of the sections above, conventional coding relates in a one-to-one manner any of the possible vectors ($M_1$) of the conceptual space to the pattern $X$ placed in the field. When the descriptive abilities of the vectors ($M_1$) are sufficiently limited, the coding is very economical in use of the field. For example, to specify the sex as male or female by Zatocoding would require some 15 positions in the field, and for every census event one of these descriptions would be required. In contrast, conventional coding would devote a subfield of one position to sex, and the two possibilities are indicated by the presence or absence of an indicium in this one position.

Conventional coding has no problems with interference, but Zatocoding is critical in this matter when there are very frequent patterns. Finally we observe that conventional coding is peculiarly adapted to the coding of a few ideas of limited variety that are expected to appear with a high frequency. In exactly this situation Zatocoding is clumsy and has
no advantage. The converse holds as well. Zatocoding is remarkably superior in the coding of ideas having a very large variety, but which are used with a low frequency. The variety of different component ideas M that Zatocoding can absorb is unlimited.

With these observations, the constructive conclusion is obvious. The coding field of a unit is divided into two parts. The larger is devoted to information coded by Zatocoding. The smaller part is retained for the relatively few pieces of information of high frequency that cause embarrassment in the Zatocoding field. Reciprocally, in cases when the conventionally-coded system breaks down for any of the reasons cited in Z, a pattern or patterns in Zatocoding can explain the trouble. As an improbable but illustrative example, if one position is devoted to the specification of sex, the conventional coding would break down in the case of a hermaphrodite. Then a pattern for hermaphrodite would be placed in the Zatocoding field.

The number of components that need to be put into conventional coding is usually very small. The criteria from interference specify the frequencies at which it is advisable to code a component with conventional coding. From the standpoint of economy of use of the field, those components that are of a simple nature and categorize easily and have a limited variety should be put into conventional coding; the problem here is to avoid the difficulties enumerated in Z. In a total field of 1000 positions, a conventionally-coded field of 50 or 100 should be more than enough to contain this content. In this part of the field, subfields of a single position and of multiple position will of course be used, e.g. there will be a subfield for the more common professions of say 10 positions to code some 1000 professions. If a
man is a doctor and lawyer, the conventional coding breaks down, and the situation is described by Zatocoding.

IX - Conclusion

The conditions of validity for the use of Zatocoding for the gathering of statistical information have been established, the limits being set up on a basis of the unlikelihood of their being exceeded in a particular case. These conditions in turn provide the functional relationships between the design parameters of a statistical application. The functional relationships are a complete set in the sense that given an application, and its requirements, the variable parameters of design can be specified. Finally, in the places where Zatocoding becomes inefficient or clumsy, it is shown that conventional coding can take over, and that the two systems of coding are in a sense complementary. Accuracy to any required degree can be attained by Zatocoding, and absolute accuracy of the conventional coding always prevails. Thus it can be concluded that Zatocoding is a new and powerful adjunct to the gathering of statistical information from a collection of records.

While the problem is solved in the engineering sense, many mathematical and statistical features remain untouched. A 3 sigma limit was taken throughout to give a single numerical value to characterize the distribution that actually prevailed. Completeness of the theory would require that the whole problem be redeveloped with the use of the distribution. The whole problem of the Zatocoded complexes in fields F over a range A is still unclear. Perhaps the most involved problem of all is the philosophical problem of the nature of knowledge or intelligence, and the set-theoretical problems involved in mapping this
intelligence into a useful recorded form such as writing on a paper or punches on a card. Powerful new insights into these problems of intelligence recording now seem near at hand, having received stimulation from this investigation into the statistical applications of Zatocoding.

* * *
X - Acknowledgement

The author wishes to acknowledge the encouragement and assistance of Professor Henry Wallman in the preparation of this paper.
PUTTING PROBABILITY TO WORK IN CODING PUNCHED CARDS - ZATOCODING\(^1\)

Calvin N. Mooers, Zator Company,  
40 Grozier Road, Cambridge, Mass.

The utility of a punched card in an index system for technical literature depends upon the information that is coded into the punches. A matter of paramount concern is the devising of some system of coding that will enable all the information needed for selection to be entered in the punches on the card. The standard difficulty is that there are too few punching positions on the card relative to the demands of the coding system. Simply stated, the card is usually too small.

This paper treats a new development in the field of punched card coding — a method for which has been coined the name ZATOCODING.\(^2\)

Zatocoding is a method of punched card coding that exploits the statistical properties of random codes superimposed upon a single coding field. Its ability to handle a quantity and variety of coded information in a limited coding field far exceeds the theoretical and practical limits previously accepted for punched card systems. In a given coding situation, it makes the ultimate use of the available coding space. Not only does Zatocoding place a great deal more information upon a coding field of a given size, but it also completely eliminates the serious --- though perhaps often unrecognized --- problems concerning the allocation of certain classes of ideas to definite subfields on the card.

Zatocoding allows the maximum utilization of the coding field of a card, there are no blank areas left when the information to be coded does not come out according to plan. In fact Zatocoding may be said to work without a plan, since it is based on the properties of random code patterns. Two simple principles govern the operation of Zatocoding and within their wide latitude, a coding system is allowed to build up almost as it will. The mathematics of probability theory and of random processes make things come out all right in the end.

We can never get something for nothing, but the advantages of Zatocoding are secured at a relatively small price. By the method, when a card sorting operation is carried out, a small and predictable number of extra cards appear -- cards in addition to those that are desired. The number of these extra cards is under control, and in a typical instance, less than one extra card appears in 256 cards sorted.

Before we enter into the detailed treatment of Zatocoding, it is pertinent first to consider certain features of the more standard coding systems.

\(^1\) Presented before the Division of Chemical Education at the 112th meeting of the American Chemical Society in New York City, September 15 - 19, 1947.

\(^2\) Patent applied for.
The following nomenclature will be used:

F is the number of positions on the card available for punching. This is the size of the coding field.

M is the number of subfields, assumed of equal size, into which the coding field is divided; in the sense that one subfield may contain one coded component idea.

V is the total vocabulary of different ideas that is allowed by a system of coding.

T is the possible number of combinations of ideas that may be coded onto a card by a system of coding.

N is the number of punches in the code pattern associated with a given idea from the vocabulary.

X is the sum of the number of punches in the code patterns taken separately for all the ideas coded on one card.

G is the resulting number of positions in the coding field that are punched out for the set of ideas coded on the card. This is the actual number of punches in the card.

S is the number of positions in the coding field that are tested for punches by the selecting apparatus.

There are two parameters of interest in the discussion of any coding system employed in a field having F positions available for punching. The first of these is the vocabulary V of different single, independent ideas that may be handled by the coding of the system. The second is the total number of combinations T of ideas taken from this vocabulary that may be coded onto the card.

If the coding field of F positions is divided into M equal subfields, we first find the number of different ideas in the vocabulary of a single subfield. One position may be either punched or unpunched, and these two possibilities may express two ideas (though these are often taken as the affirmation or denial of a single question). Two positions may be punched or left unpunched in four different ways, and can thus express four different ideas; three positions allow eight ideas; and so on, the number of ideas that may be recorded increasing as 2 raised to the power of the number of positions. Then since a subfield has F/M positions, its vocabulary is

\[ \left( \frac{F}{M} \right)^2. \]

There are M of these subfields, so the maximum vocabulary V of a coding system according to conventional coding methods is M times this, or

\[ V = M \left( \frac{2^F}{M} \right). \]

The maximum number of different combinations of ideas T that may be coded onto a card is equal to the product of the vocabularies of each of the M subfields:

\[ T = \left( \left( \frac{2^F}{M} \right)^M = (2^F). \right. \]

This expression for T until now has been taken as a theoretical upper limit on the amount of coding that may be placed on a card with a coding field having F positions.
These limitations on both \( V \) and \( T \) may be greatly exceeded according to the method of Zatocoding.

Aside from these formulas that express the limiting capabilities of conventional systems of coding, there is the plain fact that a punched card is usually too small to handle a particular job. Part of the reason may be found in the above restrictions on \( V \) and \( T \), but there is an additional serious difficulty in conventional systems whose existence may not be realized. This difficulty is the necessity to use a different subfield or part of the coding field to record each idea placed on a card.

Multiple subfields have been found necessary whenever it is desired to impress on the coding field of a card several pieces of intelligence which must act in selection independently of each other. An example of this is in the use of punched cards in personnel records of a business. There is a card for each employee. Punches in one subfield indicate the sex of the employee; in another, the age; in another the serial number assigned to the employee; another the department he is employed in; and so on. If a request is received for the selection of the personnel records of all the women in the shipping department, it is a simple matter, using standard sorting techniques, to perform this selection on the basis of the punches in the subfields of sex and department to give the desired set of cards. Here there is no difficulty.

The difficulties inherent in the use of multiple subfields appear when conventional methods are applied to problems of much greater complexity than such a simple example as that cited above. An outstanding example is given by the problem of organizing references to the chemical literature on punched cards. A large number of what may be called characteristics or types of information are important in describing the material of chemistry. One request for literature may ask for all references to chemicals that are of a reddish color, have a melting point near 100 degrees centigrade, and will not burn; thus making use of the characteristics of color, melting point, and flammability. Another may ask for references to methods of production of butadiene using a catalyst and with particular reference to the problems of recovery of waste products. Any attempt to categorize the component ideas of these requests into characteristics or classes of ideas will result in the definition of a very large number of such characteristics. Because the different characteristics are independent selective criteria, e.g. color has no relation to melting point for purposes of selection, the component ideas belonging to each characteristic are coded into separate subfields of the card. This usually results in a desire for more subfields and therefore more coding positions than one card possesses. A punched card has only a limited number of positions, and to increase their number would require a complete redesign of all the equipment, which is usually impractical.

At this point a compromise is conventionally resorted to. A drastic limitation is set upon the number of characteristics that are allowed in the system, and broadly-related ideas are forced into the same broad characteristic or class, and thus into the same coding subfield. For example, in organizing a punched card system for chemistry, with five subfields on the card, subfields 1, 2, and 3 might be assigned to the three most important elements in the composition, subfield 4 to the method of preparation, and subfield 5 to the use of the composition.

Such an organization of the system can handle a request for finding the references to hydrocarbon compounds prepared by refinery methods which are used in dry cleaning. No trouble appears if the component ideas are properly distributed over the subfields, as they are in this instance. But a request to search for all the references on wetting agents used in dry cleaning puts a double load on the fifth subfield for "use", since both wetting agent and dry cleaning are uses. Yet,
only one idea can be coded in that subfield. The system at this point has broken down. However, such a request is entirely proper. Research should not be limited by the mechanics of a library system.

Besides such overloading of a single subfield, another embarrassment encountered with the above compromise is the difficulty at times of knowing in which subfield to search for a component of a request. With an alloy of 60% iron and 40% nickel, the iron might be coded into the first subfield in the above example, since the rule may be that the most abundant element is coded into the first subfield. A 40–60% alloy of the same two metals would then be coded in the other order. Thus to find all the nickel–iron alloys of all proportions, both these possibilities would have to be tried in a separate searching operation. This is a definite disadvantage in a large system in which all the cards are sorted, since the sorting time is therefore doubled.

Another difficulty arises when important descriptive material eludes a compromise system altogether. With the above elementary example, the information that a hydrocarbon is a ring compound cannot be entered without breaking or straining the system. The idea of ring structure is certainly not one of the chemical elements, so it does not fit in subfields 1, 2 or 3. If the idea were forced into either of the remaining subfields of preparation, or of use, it would possibly displace other necessary information, and it would certainly lead to confusion in deciding in which of these subfields to search for it.

We may conclude that whatever the method of organization of a coding system with the conventional methods of coding using subfields, there will always be non-conforming pieces of information, or of situations that the coding system cannot handle.

We can sum up the situation with respect to the conventional methods of coding:

1. The total vocabulary is limited to \( V = M (2)^{F/N} \), which is often too small for the purposes desired.

2. The total possible number of combinations of this vocabulary on one card, using all the subfields, is never larger than \( T = (2)^{F} \).

3. Because of the inherent necessity for division into subfields, there is not enough room on the coding field of one card to assign a subfield to every characteristic of the subject being coded.

4. There is therefore frequently desired to put more than one component idea into a given subfield, which is impossible.

5. There is a frequent confusion of knowing in which subfield to search when a component idea could be in several alternative places, and a consequent loss of time when more than one subfield must be searched.

6. There are many situations in which necessary components of information may elude the system altogether.

Zatocoding alleviates or overcomes these difficulties to following extent:

In the example of a 40-position coding field, which we will consider in detail later, Zatocoding increases the vocabulary \( V \) by a factor of several hundred.
In the number of possible combinations $T$, the real measure of coding ability, Zatocoding offers an increase from $(10)^{12}$ to $(10)^{31}$.

It eliminates completely all the problems of the use of subfields, since Zatocoding does not make use of subfields.

Finally, as a homely fact, I can assure you that Zatocoding actually works! I have a system of several hundred cards in operation at home, and it encompasses such varied subjects as an index to my patent file, memos to myself, book reviews clipped from the paper, and some of my wife's recipes.

Zatocoding is based upon the statistical properties of code patterns which are randomly -- or nearly randomly -- distributed over the whole coding field. While the field may be divided to produce some inferior versions of the method, the greatest advantages in a small coding field are obtained when the undivided coding field is used. Notice especially that the coding field is not partitioned with respect to the different characteristics or classes of ideas that enter the system. That is, subfields are entirely avoided.

Each component idea is given a random code pattern by which the intelligence conveyed by that idea is punched into a card, and by means of which, in cooperation with other patterns, the card is to be selected. These code patterns are recorded in a coding dictionary, either by an alphabetical listing according to the words they represent; or -- as I've found most convenient -- in a classified listing of ideas. Listing or classifying the ideas is simple, since each idea is a single uncomplicated notion. On a card, however, the ideas build up descriptive complexes, just as words in ordinary language do, to delineate the complex idea written on the card.

The several code patterns from the set of component ideas delineating the complex idea on the card are superimposed upon the same coding field. The punches are made in the card without reference to the punches already in the field. The patterns of punches will intermingle and overlap, but this is expected, and by the Zatocoding method the harmful effects of this loss of identity are controlled and held to a predictably small amount. As we will see, code patterns may be added to the coding field until, on the average, about one half of the coding positions are punched out. This is the condition of optimum or maximum use of the field, and it is this condition that limits the number of code patterns that may be placed in the field.

Selection of cards coded in this manner is performed in the ordinary fashion, just as if the codes were not superimposed. The several ideas, which taken together define the desired selection, are decided upon; the code patterns are found in the coding dictionary; the selector is set up according to all these patterns; and the cards are sorted. All the desired cards appear. The intermingling of the codes has no effect on this. If a code pattern is punched out, it is punched out. But because of the intermingling of the random codes, there may be some patterns — totally unrelated to the desired patterns or their ideas — which simulate the desired patterns. These unrelated cards will be selected too. These are called extra cards.

It is only because the appearance of these extra cards is under the statistical control of the user of the Zatocoding method, and that their number can be held down to a practically small fraction of the total number of cards sorted, that this method of using superimposed random codes is feasible.

---

1 For convenience in use, lists of random patterns have been prepared.
To develop the mathematical theory of Zatocoding, we will first treat the statistics that govern the appearance of the extra cards in the selective operation. Consider a single card. In its field of F positions, there are G punches. Now if the selector is set to select all those cards that have a punch in a given randomly-chosen position in the field, the probability of the chance selection of this one card is \( G/F \), which is less than one. If the selector is set up to select on the punches in two given positions in the field chosen at random, the probability that the one card will be selected is \( (G/F)^2 \); since both punches must be present for the selection, and the probability of the simultaneous occurrence of two independent events is given by the product of the probabilities of the occurrences of the separate events. Notice that because \( G/F \) is less than one, \( (G/F)^2 \) must be less than \( G/F \). Therefore we may conclude that when the selecting apparatus is set up to select on the basis of S randomly-chosen positions, the probability of the card being selected is \( (G/F)^S \), which becomes smaller as S increases.

But every code pattern for each component idea was originally derived from a random process. Thus the pattern of punches in the coding field of a card will be randomly distributed with respect to the selector pattern, unless the card bears the same coded ideas as those set up on the selector. Therefore the above computation of the probability of the chance selection of any one card will apply to the large mass of cards in the collection which are exposed to the selector and which bear no component ideas to match those set up in the selector. Thus the ratio of the number of extra cards to the total number of cards passed through the sorting operation will be \( (G/F)^S \), assuming that all the cards have the same fraction of their field punched. By making S large, this ratio may be made small, and the appearance of the extra cards may be controlled. In the text below we will place a limit on the magnitude of \( G/F \), and then a more satisfactory expression for the ratio of extra cards will result.

If the code pattern for one component idea having N punches is placed upon the empty field of one card, N positions in the field will be punched. If the code for a second component idea with a code pattern of the same length is now superimposed on the field, there is a small probability that some of the punches from the two codes may overlap. The result is that on the average slightly less than 2N positions in the field will be punched. As the codes from more and more component ideas are added to the one coding field, this overlapping will become more frequent. Therefore \( G \), the number of punched positions in the field is not simply the sum of all the punches of the separate code patterns. The relation governing the average value of \( G \) may be found. Let \( X \) represent the sum of the punches of the separate code patterns placed in the field. Then regarding \( G \) as a function of \( X \), the probability that a new punch in the field will not overlap any already there is \( (1 - G/F) \). From this we obtain the differential equation

\[
\frac{dG}{dX} = (1 - G/F)
\]

with the initial condition \( G = 0 \) when \( X = 0 \). The solution is

\[
G = F (1 - e^{-X/F}).
\]

Notice that for small values of \( X \) this is approximately \( G = X \). With this solution for \( G \), the expression for the ratio of extra cards to all sorted cards becomes

\[
(G/F)^S = (1 - e^{-X/F})^S.
\]

The condition for the maximum utilization of the coding field may now be found. Inspection of the above expression for the ratio of extra cards reveals that the ratio increases with an increase in \( X \), but decreases with an increase of \( S \) since
the quantity in the parenthesis is always less than one. In other words, as the amount of coded intelligence impressed on the coding field increases, the extra cards appear more frequently, while an increase in the number of positions inspected by the selector apparatus will cut down the number of extra cards. Moreover, the above expression reveals for a small X/F, that X may undergo a large percentage increase and yet have no more affect on the ratio than a small percentage increase in S. But this disparity in the relative effect of X and S decreases as the fraction G/F of punched positions in the coding field increases, and in fact when G/F is near unity, or X/F is very large, the relative strength of the effect is reversed. This leads us to ask at what value of G/F are the two effects equal? Or when will, say, a 1% increase in X have an effect that is just compensated by a 1% increase in S? Stated mathematically, when will

\[ \frac{dx}{x} = \frac{ds}{s} = 1 \]

We will take this condition as defining the optimum or maximum amount of coding that we may place in the field. If we place more coding than this in the field, the difficulties with extra selections will increase more rapidly than advantage is gained by the coding of a greater amount of information. On the other hand, if we set a lower limit, we will throw away coding ability to an extent not compensated by the decreased ratio of extra selections.

An explicit formulation of the condition of optimum coding may be found by differentiating the expression

\[ (1 - e^{-X/F}) S = \text{constant}, \]

regarding X and S as the two variables. Then by inserting \( \frac{dx}{x} = 1 \), \( \frac{ds}{s} \) we obtain the relation

\[ G/F = \left(1 - e^{-X/F}\right) = \frac{1}{2}. \]

Therefore when one-half of the field on the average is punched, we have reached the optimum or maximum desirable utilization of the coding field in the sense defined above. The equation may be solved for X to find the sum of the code punches that may be expected to yield this 50% average coverage of the field. Solving, we find

\[ X = 0.69 F \]
Therefore the sum of the code punches in the separate patterns may become as great as 69% of the number of positions in the coding field, before the limiting 50% average coverage of the field will be exceeded. We will take this 69% limit on $X$ as our basic limit on the amount of coding that we place in the field. Many fields will contain less coding.

At this point we may observe that if precisely the same reasoning is followed, but if overlapping of codes is not taken into account, i.e. $X = \frac{G}{F}$, then

$$X = \frac{1}{e} F = 0.37 F.$$ 

This approximation leads to a decidedly inferior result.

With the 69% limit, the average value of $G/F$ for all the cards will be no larger than $\frac{1}{2}$, and it may be significantly less. Placing $\frac{1}{2}$ as a maximum value for $G/F$ in the expression for the ratio of extra cards, we have that the ratio of extra cards is less than

$$\left(\frac{1}{2}\right)^S.$$

This will be true so long as $X$ does not exceed the 69% limit. The development of the theory of Zatocoding is now completed.

The conclusions arrived at may be restated as basic principles governing all elementary applications of Zatocoding:

**First Principle:** The sum of the separate punches of the code patterns impressed on the coding field of one card shall not exceed 69% of the total number of positions in the field, and in that case with random codes, the number of punched positions in the field will average 50%.

**Second Principle:** When the first principle is obeyed, in a selection on the basis of $S$ positions in the field, the ratio of the number of extra cards to the total number of cards sorted will be less than $\left(\frac{1}{2}\right)^S$ in the average.

These two principles contain all that one needs to know to apply the Zatocoding method. The one tells us when a card is full of coding, and the other tells that we may diminish the extra cards by using a longer pattern in the selection.

Indirectly, these same two principles also guide building of a Zatocoded punched card system. They tell us that if we vary the length of the code patterns between different ideas, we should use the longer patterns on those ideas which when used alone may define a useful selection.

In spite of the importance of these two principles, and of their consequences, we will leave them at this point and go on to a numerical comparison of Zatocoding with conventional methods of coding information on a punched card. In the following example we will consider a card having the relatively small coding field of 40 positions, though Zatocoding is applicable to a coding field of any size.

According to the method of Zatocoding we will first compute the vocabulary $V$, and then the number of combinations $T$. In a field of $F$ positions, in which $N$ punches per code pattern are placed without restriction upon their position, the number of different patterns, and thus the possible coding vocabulary, is $pCN$ or the number of combinations of $F$ things taken $N$ at a time. Thus the coding vocabulary $V$ in a 40 position field with a four-punch code is 91,390 different codes. Other code lengths may of course be used. The capacity of this field of 40 positions is obtained by finding 69% of 40, which is 27.5, or nearly 28. Therefore the field can
contain seven four-punch codes (because four times seven equals twenty-eight). By the Zatocoding statistics we know that on the average 20 positions on the field will be punched, due to expected overlapping, when these 28 code punches are placed on the field. Seven codes out of a vocabulary of 91,390 different ones may be placed in the field. Then the number of combinations possible on the field of a card is the number of combinations of 91,390 things taken seven at a time, or something in the order of \((10)^{31}\).

By means of the earlier formulae for the conventional system of coding, comparative figures can be derived for the same 40-position coding field. The total number of combinations \(T\) is simply 2 to the 40th power, or only \((10)^{12}\). To approximate the vocabulary, we will give the conventional system an advantage and assume it has for the moment 42 positions (in order to get a number divisible by seven to correspond to the seven codes in the preceding example). Then the total vocabulary for all seven of the equal sized subfields in a field of 42 positions is only

\[
V = N \frac{2^F}{M} = 7 \cdot (2)^{42}/7 = 448.
\]

Of course the value for 40 positions would be slightly smaller.

These are enormous deficiencies in vocabulary \(V\) and number of combinations \(T\) for the conventional coding systems as compared to Zatocoding. And in addition, there are also the very serious difficulties and inconveniences attached to the use of the separate coding subfields that are eliminated completely by Zatocoding.

As another comparison of conventional methods with Zatocoding: to duplicate Zatocoding performance in a 40-position field, conventional methods would require about 103 holes, with still no abatement of the problems of subfields.

Zatocoding is flexible. It operates in a coding field of any size. Different code lengths may be used within the same coding system in order to secure the maximum utilization of a limited coding field or to get more ideas on the same card. Ideas may be used in any combination from a large coding vocabulary. The same card system can absorb simultaneously any type of subject matter: technical information, literature references, and even gardening notes. A new classification of material can be entered into the system at any time by merely coding the cards and putting them in with the rest. There is no need to plan where the coding system will be going a year from now: the card is not committed to an inflexible allocation of subfields; and Zatocoding can absorb whatever comes up.

New means lead to new methods — and Zatocoding is definitely a new means. The new methods that can be anticipated will be perhaps in a different use of language as it applies to the selective process in punched card systems. By this I mean that punched card systems now have certain typical ways of breaking down a complex idea into attributes suitable for coding. Zatocoding may lead to different ways of describing the same complex ideas with codes. As an instance, it seems that a liberal use of coded broad general ideas, taken in their conjunction, may be a preferred way of defining particular ideas with Zatocoding. On the other hand, Zatocoding can absorb and use any existing punched card classification and coding scheme for what it is worth, and it may do so while at the same time using general ideas.

Zatocoding holds the way open to new language and intellectual techniques in the problems of finding information, and these possibilities may now be exploited.