~~CONFIDENTIAL~~

The following copyright notice is not part of the original document:

Courtesy of Alcatel-Lucent ©[1945] Alcatel-Lucent. All Rights Reserved.

The content of the original document begins on the next page.

COVER SHEET FOR TECHNICAL MEMORANDA

RESEARCH DEPARTMENT

SUBJECT: A Mathematical Theory of Cryptography - Case 20878 (U)

ROUTING:

1	-	HWB-HF-Case Files	MM-	45-110-92
2	-	CASE FILES	DATE	September 1, 1945
3	-	J. W. McRae	AUTHOR	C. E. Shannon
4	-	L. Espenschied	INDEX NO.	P 0.4
5	-	H. S. Black		
6	-	F. B. Llewellyn		
7	-	H. Nyquist		
8	-	D. M. Oliver		
9	-	R. K. Potter		
10	-	C. B. H. Feldman	(“SECRET” heavily crossed out)	
11	-	R. C. Mathes		
12	-	R. V. L. Hartley	(“ABSTRACT” crossed out with Xs)	
13	-	J. R. Pierce		
14	-	H. W. Bode		
15	-	R. L. Dietzo1d		
16	-	L. A. MacCall	DOWNGRADED AT 3 YEAR INTERVALS	
17	-	W. A. Shewhart	DECLASSIFIED AFTER 12 YEARS	
18	-	S. A. Schelkunoff	DOD DIR 5200.10	
19	-	C. E. Shannon		
20	-	Dept. 1000 Files		

ABSTRACT

A mathematical theory of secrecy systems is developed. Three main problems are considered. (1) A logical formulation of the problem and a study of the mathematical structure of secrecy systems. (2) The problem of “theoretical secrecy,” i.e., can a system be solved given unlimited time and how much material must be intercepted to obtain a unique solution to cryptograms. A secrecy measure called tho “equivocation” is defined and its properties developed. (3) The problem of “practical secrecy.” How can systems be made difficult to solve, even though a solution is theoretically possible.

THIS DOCUMENT CONTAINS INFORMATION AFFECTING THE NATIONAL DEFENSE OF THE UNITED STATES WITHIN THE MEANING OF THE ESPIONAGE LAWS TITLE 18 U.S.C. sections 793 and 794.
ITS TRANSMISSION OR THE REVELATION OF ITS CONTENTS IN ANY MANNER TO AN UNAUTHORIZED PERSON IS PROHIBITED BY LAW.

BEST COPY AVAILABLE

~~CONFIDENTIAL~~

A Mathematical Theory of Cryptography - Case 20878

MM-45-110-92
September 1, 1945
Index P0.4

MEMORANDUM FOR FILE

Introduction and Summary

In the present paper a mathematical theory of cryptography and secrecy systems is developed. The entire approach is on a theoretical level and is intended to complement the treatment found in standard works on cryptography.* There, a detailed study is made of the many standard types of codes and ciphers, and of the ways of breaking them. We will be more concerned with the general mathematical structure and properties of secrecy systems.

The presentation is mathematical in character. We first define the pertinent terms abstractly and then develop our results as lemmas and theorems. Proofs which do not contribute to an understanding of the theorems have been placed in the appendix.

The mathematics required is drawn chiefly from probability theory and from abstract algebra. The reader is assumed to have some familiarity with these two fields. A knowledge of the elements of cryptography will also be helpful although not required.

The treatment is limited in certain ways. First, there are two general types of secrecy system; (1) concealment systems, including such methods as invisible ink, concealing a message in an innocent text, or in a fake covering cryptogram, or other methods in which the existence of the message is concealed from the enemy; (2) "true" secrecy systems where the meaning of the message is concealed by cipher, code, etc., although its existence is not hidden. We consider only the second type—concealment systems are more of a psychological than a mathematical problem. Secondly, the treatment is limited to the case of discrete information, where the information to be enciphered consists of a sequence of discrete symbols, each chosen from a finite set. These symbols may be letters in a language, words of a language,

* See, for example, H.F.Gaines, "Elementary Cryptanalysis," or M. Givierge, "Cours de Cryptographie."

amplitude levels of a “quantized” speech or video signal, etc., but the main emphasis and thinking has been concerned with the case of letters. A preliminary survey indicates that the methods and analysis can be generalized to study continuous cases, and to take into account the special characteristics of speech secrecy systems.

The paper is divided into three parts. The main results of those sections will now be briefly summarized. The first part deals with the basic mathematical structure of language and of secrecy systems. A language is considered for cryptographic purposes to be a stochastic process which produces a discrete sequence of symbols in accordance with some systems of probabilities. Associated with a language there is a certain parameter D which we call the redundancy of the language. D measures, in a sense, how much a text in the language can be reduced in length without losing any information. As a simple example, if each word in a text is repeated a reduction of 50 per cent is immediately possible. Further reductions may be possible due to the statistical structure of the language, the high frequencies of certain letters or words, etc. The redundancy is of considerable importance in the study of secrecy systems.

A secrecy system is defined abstractly as a set of transformations of one space (the set of possible messages) into a second space (the set of possible cryptograms). Each transformation of the set corresponds to enciphering with a particular key and the transformations are supposed reversible (non-singular) so that unique deciphering is possible when the key is known.

Each key and therefore each transformation is assumed to have an *a priori* probability associated with it—the probability of choosing that key. The set of messages or message space is also assumed to have *a priori* probabilities for the various messages, i.e., to be a probability or measure space.

In the usual cases the “messages” consist of sequences of “letters.” In this case as noted above the message space is represented by a stochastic process which generates sequences of letters according to some probability structure.

These probabilities for various keys and messages are actually the enemy cryptanalyst’s *a priori* probabilities for the choices in question, and represent his *a priori* knowledge of the situation. To use the system a key is first selected and sent to the receiving point. The choice of a key determines a particular transformation in the set forming the system. Then a message is selected and the particular transformation applied to this message to produce a cryptogram. This cryptogram is transmitted to the receiving point by a channel that may be intercepted by the enemy. At the receiving end the inverse of the particular transformation is applied to the cryptogram to recover the original message.

If the enemy intercepts the cryptogram he can calculate from it the *a posteriori* probabilities of the various possible messages and keys which might have produced this cryptogram. This set of *a posteriori* probabilities constitutes his knowledge of the key and message after the interception.* The calculation of these *a posteriori* probabilities is the generalized problem of cryptanalysis.

As an example of these notions, in a simple substitution cipher with random key there are 26! transformations, corresponding to the 26! ways we can substitute for 26 different letters. These are all equally likely and each therefore has an *a priori* probability 1/26!. If this is applied to "normal English" the cryptanalyst being assumed to have no knowledge of the message source other than that it is English, the *a priori* probabilities of various messages of N letters are merely their frequency in normal English text.

If the enemy intercepts N letters of cryptogram in this system his probabilities change. If N is large enough (say 50 letters) there is usually a single message of *a posteriori* probability nearly unity, while all others have a total probability nearly zero. Thus there is an essentially unique "solution" to the cryptogram. For N smaller (say $N = 15$) there will be many messages and keys of comparable probability, with no single one nearly unity. In this case there are multiple "solutions" to the cryptogram.

Considering a secrecy system to be a set of transformations of one space into another with definite probabilities associated with each transformation, there are two natural combining operations which produce a third system from two given systems. The first combining operation is called the product operation and corresponds to enciphering the message with the first system R and enciphering the resulting cryptogram with system S; the keys for R and S being chosen independently. This total operation is a secrecy system whose transformations consist of all the products (in the usual sense of products of transformations) of transformations in S with transformations in R. The probabilities are the products of the probabilities for the two transformations.

The second combining operation is "weighted addition."

$$T = pR + qS \qquad p + q = 1$$

* "Knowledge" is thus identified with a set of propositions having associated probabilities. We are here at variance with the doctrine often assumed in philosophical studies which considers knowledge to be a set of propositions which are either true or false.

It corresponds to making a preliminary choice as to whether system R or S is to be used with probabilities p and q, respectively. When this is done R or S is used as originally defined.

It is shown that secrecy systems with these two combining operations form essentially a "linear associative algebra" with a unit element, an algebraic variety that has been extensively, studied by mathematicians. Some of the properties of this algebra are developed.

Among the many possible secrecy systems there is one type with many special properties. This type we call a "pure" system. A system is pure if for any three transformations T_i, T_j, T_k in the set the product

$$T_i T_j^{-1} T_k$$

is also a transformation in the set, and all keys are equally likely. That is enciphering, deciphering, and enciphering with any three keys must be equivalent to enciphering with some key.

With a pure cipher it is shown that all keys are essentially equivalent—they all lead to the same set of *a posteriori* probabilities. Furthermore, when a given cryptogram is intercepted there is a set of messages that might have produced this cryptogram (a "residue class") and the *a posteriori* probabilities of messages in this class are proportional to the *a priori* probabilities. All the information the enemy has obtained by intercepting the cryptogram is a specification of the residue class. Many of the common ciphers are pure systems, including simple substitution with random key. In this case the residue class consists of all messages with the same pattern of letter repetitions as the intercepted cryptogram.

Two systems R and S are defined to be "similar" if there exists a fixed transformation A with an inverse, A^{-1} such that

$$R = ASA^{-1}$$

If R and S are similar, a one-to-one correspondence between the resulting cryptograms can be set up leading to the same *a posteriori* probabilities. The two systems are cryptanalytically the same.

The second main part of the paper deals with the problem of "theoretical security". How secure is a system against cryptanalysis when the enemy has unlimited time and manpower available for the analysis or intercepted cryptograms?

"Perfect Secrecy" is defined by requiring of a system that after a cryptogram is intercepted by the enemy the *a posteriori* probabilities of this cryptogram representing various messages be identically the same as the *a priori* probabilities of the same messages before the interception. It is shown that perfect secrecy is possible but requires, if the number of messages is finite, the same number of possible keys—if the message is thought of as being constantly generated at a given "rate" R, (to be defined later), key must be generated at the same or a greater rate.

If a secrecy system with a finite key is used, and N letters of cryptogram intercepted, there will be, for the enemy, a certain set of messages with certain probabilities, that this cryptogram could represent. As N increases the field usually narrows down until eventually there is a unique "solution" to the cryptogram—one message with probability essentially unity while all others are practically zero. A quantity $Q(N)$ is defined, called the equivocation, which measures in a statistical way how near the average cryptogram of N letters is to a unique solution; that is, how uncertain the enemy is of the original message after intercepting a cryptogram of N letters. Various properties of the equivocation are deduced—for example the equivocation of the key never increases with increasing N. This quantity Q is a theoretical secrecy index—theoretical in that it allows the enemy unlimited time to analyse the cryptogram.

The function $Q(N)$ for a certain idealized type of cipher called the random cipher is determined. With certain corrections this function can be applied to many cases of practical interest. This gives a way of calculating approximately how much intercepted material is required to obtain a solution to a secrecy system. It appears from this analysis that with ordinary languages and the usual types of ciphers (not codes) this "unicity distance" is approximately $|K|/D$. Here $|K|$ is a number measuring the "size" of the key space. If all keys are *a priori* equally likely $|K|$ is the logarithm of the number of possible keys. D is the redundancy of the language and measures the excess information content of the language. In simple substitution with random key on English $|K|$ is $\log_{10} 26!$ or about 20 and D is about .7 for English. Thus unicity occurs at about 30 letters.

It is possible to construct secrecy systems with a finite key for certain "languages" in which the function $Q(N)$ does not approach zero as $N \longrightarrow \infty$. In this case, no matter how much material is intercepted, the enemy still does not get a unique solution to the cipher but is left with many alternatives, all of reasonable probability. Such systems we call *ideal* systems. It is possible in any language to approximate such behavior—i.e., to make the approach to zero of $Q(N)$ recede out to arbitrarily large N. However, such systems have a number of drawbacks, such as complexity and sensitivity to errors in transmission of the cryptogram.

The third part of the paper is concerned with “practical secrecy.” Two systems with the same key size may both be uniquely solvable when N letters have been intercepted, but differ greatly in the amount of labor required to effect this solution. An analysis of the basic weaknesses of secrecy systems is made. This leads to methods for constructing systems which will require a large amount of work to solve. A certain incompatability among the various desirable qualities of secrecy systems is discussed.

PART I

FOUNDATIONS AND ALGEBRAIC STRUCTURE OF SECRECY SYSTEMS

1. Choice, Information and Uncertainty

Suppose we have a set of possible events whose probabilities of occurrence are $p_1, p_2, \ldots, p_n$. These probabilities are known, but that is all we know concerning which event will occur. Can we define a quantity which will measure in some sense how "uncertain" we are of the outcome? How much "choice" is involved in the selection of the event by the chance element that operates with these probabilities? We propose as a numerical measure of this rather vague notion the quantity

$$H = -\sum_{i=1}^{n} p_i \log p_i.$$

There are many reasons for this particular formula. Quantities of this kind appear continually in the present paper and in the study of the transmission of information.

To justify this definition we will state a number of properties that follow from it. These properties will not be proved here,* but are easily deduced from the definition. Properties of $H = -\sum p_i \log p_i$

1. $H = 0$ if and only if all the p_i but one are zero, this one having the value unity. Thus only when we are certain of the outcome does H vanish.

2. For a given n, H is a maximum and equal to $\log n$ if and only if all the p_i. are equal (i.e. $1/n$). This is also intuitively the most uncertain. situation.

3. Suppose there are two events in question, with m possibilities for the first and n for the second. Let p_{ij} be the probability of the joint occurrence of i for the first and j tor the second. The uncertainty of the joint event is

$$H = -\sum_{i,j} p_{ij} \log p_{ij}.$$

For given probabilities $p_i = \sum_j p_{ij}$ for the first and

* It is intended to develop these results in coherent fashion in a forthcoming memorandum on the transmission of information.

$q_j = \sum_i p_{ij}$ for the second, the quantity H is maximized if and only if the events are independent, i.e., $p_{ij} = p_i q_j$. This maximum value, is the sum of the individual uncertainties

$$\begin{aligned} \text{H} \quad &= H_1 + H_2 \\ &= -\sum p_i \log p_i - \sum q_i \log q_i. \end{aligned}$$

These facts can be generalized to any number of different events.

4. Suppose there are two chance events A and B as in 3, not necessarily independent. We define the mean conditional uncertainty of B, knowing A as

$$\overline{H}_A = \sum_A p(A) H_A(B)$$

where $H(B)$ is the uncertainty of B when A has a definite value A
Thus $\overline{H}_A(B)$ is the average uncertainty of B for all different events A, weighted according to their different probabilities of occurrence. The uncertainty of the joint event is the sum of the uncertainty of the first and the mean conditional uncertainty of the second. In symbols

$$H(A,B) = H(A) + \overline{H}_A(B)$$

This is true whether or not there are any casual connections or correlations between the two events.

5. In the same situation the uncertainty of B is not greater than the joint uncertainty $H(A,B)$.

$$H(B) \leq H(A,B)$$

The equality holds if and only if every B (of probability greater than zero) is consistent with only one A. That is, if A is uniquely determined by B.

6. From properties 3 and 4 we have

$$\begin{aligned} H(A) \quad &+ \quad && H(B) \geq H(A,B) \\ H(B) \quad &\geq && H(A,B) - H(A) \\ &= && H(A) + \overline{H}_A(B) - H(A) \\ H(B) \quad &\geq \overline{H}_A(B) && \end{aligned}$$

Thus the uncertainty of B is not greater than its average value when we know A. Additional information never increases average uncertainty. The equality holds if and only if A and B are independent.

7. Suppose we have a set of probabilities $p_1, p_2, \dots, p_n$
   Any change toward equalization of these (supposing them unequal) increases H. Thus if $p_1 \leq p_2$ and we increase p_1, decreasing p_2 an equal amount (to keep the sum $\sum p_i$ constant at unity) so that p_1 and p_2 are more nearly equal, then H increases. More generally if we perform any "averaging" operation on the p_i of the form

$$p_i' = \sum a_{ij} p_j$$

   where $\sum_i a_{ij} = 1$ and all $a_{ij} \geq 0$ then H increases (except in the special case where this transformation amounts to no more than a permutation of the p_j with H of course remaining the same).

8. H measures in a certain sense how much "information is generated" when the choice is made. Suppose such a chance event occurs and we wish to describe which of the n possible events took place. The average amount of paper required to write it down in a properly chosen notation is in the cases of interest to us, about proportional to H. Thus there might be $10^{30} + 10^{50}$ possible events, with 10^{30} of them having a probability $\frac{1}{2}10^{-30}$ and 10^{-50} a probability of $\frac{1}{2}10^{-50}$. We could set up a notational system to describe which event occurs as follows. We number the events from 1 up to $10^{30} + 10^{50}$ and when one occurs write down the corresponding number. The average amount of paper required will be proportional to the average number of digits we need. This will be nearly 30 if the event is in the first group of 10^{30} and about 50 if in the second group. Thus the average number of digits is about 40. We also have

$$H = -10^{30}\frac{1}{2}\log\frac{1}{2}10^{30} - 10^{50}\frac{1}{2}\log\frac{1}{2}10^{50} \doteq 40$$

9. Although the last result is only approximately true when the number of choices is finite it becomes exactly true when an unlimited sequence of choices is made. Thus if a sequence of N independent choices is taken each choice being from n possibilities with probabilities $p_1, \ldots, p_n$ then the total amount of information generated is

$$H = -N \sum p_i \log p_i$$

If N is sufficiently large, the expected number of digits required to register the particular choice made is arbitrarily close to H, providing the correspondence between sequences of digits and sets of choices is correctly made. If incorrectly made it will be greater than H. Moreover, if N is sufficiently large the probability of needing much more than H digits is very small.

10. It can be shown that if we require certain reasonable properties of a measure of choice or uncertainty then the formula $-\sum p_i \log p_i$ necessarily follows. These required properties and the proof of this statement are given in Appendix I. The chief property is that the measure be in a sense additive—if a choice be decomposed into a series of choices the total choice is the sum (properly weighted) of the individual choices.

11. Finally we note that quantities of the type $\sum p_i \log p_i$ have appeared previously as measures of randomness, particularly in statistical mechanics. Indeed the H in Boltmann's H theorem is defined in this way, p_i being the probability of a system being in cell i of its phase space. Most of the entropy formulas contain terms of this type.

The base which is used in taking logarithms in the formula. amounts to a choice of the unit of measure. If the base is 10 we will call the resulting units "digits;" if the base is two the units will be called "alternatives." One digit is about 3.3 alternatives. A choice from 1000 equally likely possibilities is 3 digits or about 10 alternatives.

2. Language as a Stochastic Process

A natural language, such as English, can be studied from many points of view—lexicography, syntax, semantics, history, aesthetics, etc. The only properties of a language of interest in cryptography are statistical properties. What are the frequencies of the various letters, of different digrams (pairs of letters), trigrams, words, phrases, etc.? What is the probability that a given word occurs in a certain message? The "meaning" of a message has significance only in its influence on these probabilities. For our purposes all other properties of language can be omitted. We consider a language, therefore, to be a stochastic (i.e., a statistical) process which generates a sequence of symbols according to some system of probabilities. The symbols will be the letters of the language, together with punctuation, spaces etc., if these occur.

Conversely any stochastic process which produces a discrete sequence of symbols will be said to be a language. This will include such cases as:

1. Natural written languages such as English, German, Chinese.

2. Continuous information sources that have been rendered discrete by some quantizing process. For example, the quantized speech from a PCM transmitter, or a quantized television signal.

3. "Artificial" languages, where we merely define abstractly a stochastic process which generates a sequence of symbols. The following are examples of artificial languages.

(A) Suppose we have 5 letters A,B,C,D,E which are chosen each with probability .2, successive choices being independent. This would lead to a sequence of which the following is a typical example.

B D C B C E C C C A D C B D D A A E C E E A

A B B D A E E C A C E E B A E E C B C E A D

This was constructed with the use of a table of random numbers.*

(B) Using the same 5 letters let the probabilities be .4, .1, .2, .2, .1 respectively, with successive choices independent. A typical "text" in this language is then:

A A A C D C B D C E A A D A D A C E D A

E A D C A B E D A D D C E C A A A A A D

(C) A more complicated structure is obtained if successive letters are not chosen independently but their probabilities depend on preceding letters. In the simplest case of this

* Kendall and Smith, "Tables of Random Sampling Numbers" Cambridge, 1939.

type a choice depends only on the preceding letter and not on ones before that. The statistical structure can then be described by a set of transition probabilities $p_i(j)$, the probability that letter i is followed by letter j. The indices i and j range over all the letters in the language. A second equivalent way of specifying the structure is to give the digram probabilities $p(i,j)$, the relative frequency of the digram ij in the language. The letter frequencies $p(i)$, (the probability of letter i). the transition probabilities $p_i(j)$ and the digram probabilities $p(i,j)$ are re1ated by the following formulas.

$$p(i) = \sum_j p(i,j) = \sum_j p(j,i) = \sum_j p(j)p_j(i)$$
$$p(i,j) = p(i)p_i(j)$$
$$\sum_j p_i(j) = \sum_i p(i) = \sum_{i,j} p(i,j) = 1$$

As a specific example suppose there are three letters A, B, C with the probability tables:

$p_i(j)$		j		
		A	B	C
	A	0	.8	.2
i	B	.5	.5	0
	C	.5	.4	.1

$p(i)$	
A	$\frac{9}{27}$
B	$\frac{16}{27}$
C	$\frac{2}{27}$

$p(i,j)$		j		
		A	B	C
	A	0	$\frac{4}{15}$	$\frac{1}{15}$
i	B	$\frac{8}{27}$	$\frac{8}{27}$	0
	C	$\frac{1}{27}$	$\frac{4}{135}$	$\frac{1}{135}$

A typical text in this language is the following.

A B B A B A B A B A B A B A B B B A B B B B B A B
A B A B A B A B B B A C A C A B B A B B B B A B B
A B A C B B B A B A

The next increase in complexity would involve trigram frequencies but no more. The choice of a letter would depend on the preceding two letters but not on the text before that point. A set of trigram frequencies $p(i, j, k)$ or equivalently a set of transition probabilities $p_{ij}(k)$ would be required. Continuing in this way one obtains successively more complicated stochastic processes. In the general n-gram case a set of n-gram probabilities $p(i_1, i_2, \ldots, i_n)$ or of transition probabilities $p_{i_1,i_2,\ldots,i_{n-1}}(i_n)$ is required to specify the statistical structure.

(D) Stochastic processes can also be defined which produce a text consisting of a sequence of "words." Suppose there are 5 letters A, B, C, D, E, and 16 "words" in the language with associated probabilities:

.10	*A*	.16	*BEBE*	.11	*CABED*	.04	*DEB*
.04	*ADEB*	.04	*BED*	.05	*CEED*	.15	*DEED*
.05	*ADEE*	.02	*BEED*	.08	*DAB*	.01	*EAB*
.01	*BADD*	.05	*CA*	.04	*DAD*	.05	*EE*

Suppose successive "words" are chosen independently and are separated by a space. A typical message might be:

DAB EE A BEBE DEED DEB ADEE ADEE EE DEB BEBE BEBE
BEBE ADEE BED DEED DEED CEED ADEE A DEED DEED BEBE
CABED BEBE BED DAB DEED ADEB

If all the words are of finite length this process is equivalent to one of the preceding type, but the description may be simpler in terms of the word structure and probabilities. We may also generalize here and introduce transition probabilities between words, etc.

These artificial languages are useful in constructing simple problems and examples to illustrate various possibilities. We can also approximate to a natural language by means of a series of simple artificial languages. The zero order approximation is obtained by choosing all letters with the same probability and independently. The first order approximation is obtained by choosing successive letters independently but each letter having the same probability that it does in the natural language. Thus in the first order approximation to English E is chosen with probability .12 (its frequency in normal English) and W with probability .02, but there is no influence between adjacent letters and no tendency to form the preferred digrams such as TH, ED, etc. In the second order approximation digram structure is introduced. After a letter is chosen, the next one is chosen in accordance with the frequencies with which the various letters follow the first one. This requires a table of digram frequencies $p_i(j)$, the frequency with which letter j follows letter i. In the third order approximation trigram structure is introduced. Each letter is chosen with probabilities which depend on the preceding two letters.

3. The Series of Approximations to English

To give a visual idea of how this series of processes approaches a 1anguage, typical sequences in the approximations to English have been constructed and are given below. In all cases we have assumed a 27 symbol "alphabet," the 26 letters and a space.

1. Zero order approximation (symbols independent and equiprobable).

   XFOML RXKHRJFFJUJ ZLPWCFWKCYJ FFJEYVKCQSCXYD QPAAMKBZAACIBZLHJQD

2. First order approximation (symbols independent but with frequencies of English text).

   OCRO HLI RGWR NMIELWIS EU LL NBNESEBYA TH EEI ALHENHTTPA OOBTTVA NAH BRL

3. Second order approximation (digram structure as in English).

   ON IE ANTSOUTINYS ARE T INCTORE ST BE S DEAMY ACHIN D ILONASIVE TUCOOWE AT TEASONARE FUSO TIZIN ANDY TOBE SEACE CTISBE

4. Third order approximation (trigram structure as in English).

   IN NO IST LAT WHEY CRATICT FROURE BIRS GROCID PONDENOME
   OF DEMONSTURES OF THE REPTAGIN IS REGOACTIONA OF CRE

5. 1st Order Word Approximation. Rather than continue with tetragram,
   . . ., n-gram structure, it is easier and better to jump at this point to word units. Here words are chosen independently but with their appropriate frequencies.

   REPRESENTING AND SPEEDILY IS AN GOOD APT OR COME CAN DIFFERENT NATURAL HERE HE THE A IN CAME THE TO OF TO EXPERT GRAY COME TO FURNISHES THE LINE MESSAGE HAD BE THESE.

6. 2nd Order Word Approximation. The word transition probabilities are correct but no further structure is included.

   THE HEAD AND IN FRONTAL ATTACK ON AN ENGLISH WRITER THAT THE CHARACTER OF THIS POINT IS THEREFORE ANOTHER METHOD FOR THE LETTERS THAT THE TIME OF WHO EVER TOLD THE PROBLEM FOR AN UNEXPECTED

The resemblance to ordinary English text increases quite noticeably at each of the above steps. Note that these samples have reasonably good structure out to about twice the range that is taken into account in their construction. Thus in (3) the statistical process insures reasonable text for two-letter sequence, but four-letter sequences from the sample can usually be fitted into good sentences. In (6) sequences of 4 or more words can easily be placed in sentences without unusual or strained constructions. The particular sequence of ten words "attack on an English writer that the character of this" is not all unreasonable.

The first two samples were constructed by the use of a book of random numbers in conjunction for (2) with a table of letter frequencies. This method might have been continued for (3), (4), and (5), since digram, trigram, and word frequency tables are available, but a simpler equivalent method was used. To construct (3) for example one opens a book at random and selects a letter at random on the page. This letter is recorded. The book is then opened to another page and one reads until this letter is encountered. Thesucceeding letter is then recorded. Turning to another page this second letter is searched for and the succeeding letter recorded, etc. A similar process was used for (4), (5), and (6). It would be interesting if further approximations could be constructed, but the labor involved becomes enormous at the next stage.

The stochastic process 6 is already sufficiently close to English for many cryptographic purposes since most cryptanalysis is based on "local" structure of not more than two or three words in length.

4. Graphical Representation of a Markoff Process

Stochastic processes of the type described above are known mathematically as discrete Markoff processes and have been extensively studied in the literature.* The general case can be described as follows. There exist a finite number of possible "states" of a system:

* For a detailed treatment see M. Frechet, "Methods des fonctions arbitraires. Theorie des événements en chaine dans le cas d'un nombre fini d'états possibles," Paris, Gauthier-Villars, 1938.

$S_1, S_2, \ldots, S_n$ In addition there is a set of transition probabilities: $q_i(j)$ the probability that if the system is in state S_i it will next go to state S_j. To make this Markoff process into a language generator we need only assume that a letter is produced for each transition from one state to another. The states will correspond to the "residue of influence" from preceding letters.

The situation can be represented graphically as shown in Figs. 1, 2, 3 and 4. The "states" are the junction points in the graph and the probabilities and letters produced for a transition are given beside the corresponding line. Fig. 1 is for the example B in Section 2, while Fig. 2, corresponds to the example C. In Fig. 1 there is only one state since successive letters are independent. In Fig. 2 there are as many states as letters. If a trigram example were constructed there would be at most n 2 states corresponding to the possible pairs of letters preceding the one being chosen. Figs. 3 and 4 show two graphs for the case of word structure in example D. In these S corresponds to the "space" symbol. In Fig. 3 each word has a separate chain of branches from the left to the right junction point, while in Fig. 4 the branches have been combined, simplifying the graph.

5. Pure and Mixed Languages

As we have indicated above a "language" for our purposes can be considered to be generated by a Markoff process. Among the possible discrete Markoff processes there is a group with special properties of significance in cryptographic work. This special class consists of the "ergodic" processes and we shall call the corresponding languages "pure languages." Although a rigorous definition of an ergodic process is somewhat involved, the general idea is simple. In an ergodic process every sequence produced by the process is the same in statistical properties. Thus the letter frequencies, digram frequencies, etc., obtained from particular sequences will, as the lengths of the sequences increases, approach definite limits independent of the particular sequence. Actually this is not true of every sequence but the set for which it is false has probability zero. Roughly the ergodic property means statistical homogeneity.

All the examples of artificial languages given above are pure, the corresponding Markoff process being ergodic. This property is related to the structure of the corresponding graph. If the graph has two properties the language it generates will be pure. These properties are:

1. The graph cannot be divided into two parts A and B such that it is impossible to go from junction points in part A to junction points in part B along lines of the graph in the direction of arrows and also impossible to go from nodes in part B to nodes in part A.

2. A closed series of lines in the graph with all arrows on the lines pointing in the same orientation will be called a "circuit." The "length" of a circuit is the number of lines in it. Thus in Fig. 4 the series BEBES is a circuit of length 4. The second property required is that the greatest common divisor of the lengths of all circuits in the graph be one.

If the first condition is satisfied but the second one violated by having the greatest common divisor equal to $d > 1$, the sequences have a certain type of periodic structure. The various sequences fall into d different classes which are statistically the same apart from a shift of the origin (i.e. which letter in the sequence is called letter 1). By a shift of from 0 up to $d - 1$ any sequence can be made statistically equivalent to any other. A simple example with $d = 2$ is the following. There are three possible letters a,b,c. Letter a is followed with either b or c with probabilities $\frac{1}{3}$ and $\frac{2}{3}$ respectively. Either b or c is always followed by letter a. Thus a typical sequence is

$$a\ b\ a\ c\ a\ c\ a\ c\ a\ b\ a\ c\ a\ b\ a\ b\ a\ c\ a\ c.$$

This type of situation is not of much importance for our work.

If the first condition is violated the graph may be "separated" into a set of subgraphs each of which satisfies the first condition. We will assume that the second condition is also satisfied for each subgraph. We have in this case what may be called a "mixed" language made up of a number of pure components. The components correspond to the various subgraphs. If $L_1, L_2, L_3, \ldots$ are the component languages we may write

$$L = p_1L_1 + p_2L_2 + p_3L_3 + \cdots$$

where p_i is the *a priori* probability of the component language L_i.

Physically the situation represented is this. There: are several different languages $L_1, L_2, L_3, \ldots$ which are each of homogeneous statistical structure (i.e., they are pure languages). We do not know *a priori* which is to be used, but once the sequence starts in a given pure component L_i it continues indefinitely according to the statistical structure of that component. We do have, however, a set of *a priori* probabilities for the various components, $p_1, , p_2, \ldots.$

As an example one may take two of the artificial languages defined above and assume $p_1 = .2$ and $p_2 = .8$. A sequence from the mixed langunge

$$L = .2L_1 + .8L_2$$

would be obtained by choosing first L_1 or L_2 with probabilities .2 and ,8 and after this choice generating a sequence from whichever was chosen.

A natural language, such as English or German, is not, of course, pure. Different kinds of text, literary, newspaper, technical or military, display consistently different types of structure. These differences are small, however, in comparison with the differences between different natural languages. If only local structure—letter, digram and trigram frequencies, for instance—is of much importance, it is reasonable to consider "normal English" to be nearly pure.

6. Information Rate and Redundancy of a Language

Suppose we have a pure language L produced by a given Markoff process. Associated with the language there are certain parameters which are of significance in questions of transforming the language and in cryptography. The most important of these is what we will call the "information rate" R for the language. It measures the rate at which the Markoff process "generates information," as determined by the measurement of the amount of choice available on the average per letter of text that is produced. In Section 1 we defined the amount of choice when there are various possibilities with probabilities $p_1, p_2, \ldots, p_n$ as

$$H = -\sum p_i \log p_i$$

In a Markoff process with a number of different "states" there will be a choice value H_i for each of these states and a probability of being in each of the states (or a frequency with which this state occurs). If this relative frequency for state i is p_i, the average amount of choice is

$$R = \sum p_i H_i$$

summed over all the states. This is the definition of the information rate for the language. If $p_i(j)$ is the probability of producing letter j when in state i we have

$$H_i = \sum p_i(j) \log p_i(j)$$

the sum being over all the letters in the language. Thus

$$R = \sum_{ij} p_i p_i(j) \log p_i(j)$$

The information rate R has the units of alternatives (or digits) per letter since it measures the average amount of choice per letter of text that is produced.

A second parameter of importance is the "maximum rate" R_0 for the source. This is defined simply as the logarithm of the number of different letters in the language. R is also measured in alternatives or digits per letter. If successive letters are chosen independently and each letter is equally likely $R_0 = R$. Otherwise we have $R < R_0$.

R and R_0 are actually two limiting cases of information rates for the language. R_0 may be said to be the rate when no statistical structure is taken into consideration and R is the rate when all the structure is taken into account. Between these there is an infinite series of rates $R_1, R_2, \ldots, R_n, \ldots$ which take some of the statistical structure into account. R_1 takes the letter frequencies into account and is defined by

$$R_i = \sum p(i) \log p(i)$$

where $p(i)$ is the probability of letter i.
R_2 takes digram structure into account and is defined by

$$R_2 = \sum p(i) p_i(j) \log p_i(j)$$

where the $p(i)$ are letter probabilities and $p_i(j)$ the transition probabilities, i.e., the probability of letter i being followed by letter j. In general we define

$$R_n = \sum p(i_1, i_2, \ldots, i_{n-1}) p_{i_1 i_2, \ldots, i_{n-1}} p_{i_1 i_2, \ldots, i_{n-1}}(i_n)$$

where the sum is on all indices $i_1, \ldots, i_n$ and $p_{i_1,\ldots,i_{n-1}}$ is the probability of $(n-1)$- gram $i_1 \ldots i_{n-1}$ with $p_{i_1 i_2,\ldots,i_{n-1}}(i_n)$ the probability of this $(n-1)$-gram being followed by letter i_n. R_n may be called the n-gram information rate for the language. It can be shown that

$$R_0 \geq R_1 \geq R_2 \geq \cdots \geq R_n \geq \cdots R_\infty = R$$

These rates determine how much a language can be "compressed" in length by a suitable encoding process. A language with maximum rate R_0 and rate R can be transformed in such a way that a sequence of letters N letters long is transformed into a sequence of letters only N' letters long where

$$N'R_0 = NR$$

(This is approximate and only exactly true in the limit as $N \longrightarrow \infty$.) Thus the information is "compressed" in the ratio

$$\frac{R}{R_0}$$

This is the greatest compression ratio possible. It makes use of all the statistical structure of the language. If only n-gram structure is made use of, a compression ratio

$$\frac{R_n}{R_0}$$

is the best possible.

The compression obtained in this way is only a statistical gain. Some infrequent sequences are encoded into much longer sequences while the more probable ones go into shorter sequences so that on the average the length is decreased. It is the type of compression obtained in telegraphy by using the shortest telegraph symbol, a single dot, for the most frequent letter E, while the uncommon letters Q, Z, etc., are encoded into longer telegraph symbols. An average reduction in time of transmission is obtained but there are possible sequences, e.g., $QQQ\ldots$, which require much longer.

Performing a transformation on a language L which compresses as much as possible will be called reducing L to a "normal" form. When this has been done it can be shown that all letters in the output are equally likely and independent. Actually to realize this transformation would usually require an infinitely complex machine, but we can always approximate it as closely as desired with a machine of finite complexity.

The quantity

$$D = R_0 = -R$$

will be called the redundancy rate of the language. It measures the excess information that is sent if sequences in the language are transmitted in their original form (without compression or reduction to normal form). Correspondingly there is a whole series of redundancy rates:

$$\begin{aligned} D_0 &= R_0 - R_0 = 0 \\ D_1 &= R_0 - R_1 \\ D_2 &= R_1 - R_2 \end{aligned}$$

$$\begin{aligned} D_n &= R_0 - R_n \\ D &= R_0 - R \end{aligned}$$

D_n is the redundancy rate due to n-gram structure in the language.

The redundancy D can also be said to measure the amount of statistical structure in the language. If the sequence is purely random $D = 0$ while at the other extreme if each letter is completely determined by preceding letters with no freedom of choice, D has its maximum possible value R_0. It is sometimes convenient to use the "relative" redundancy D/R_0 which must lie between 0 and 100%.

If we have a source of rate R, maximum rate R_0 (both in digits per letter) and consider the possible sequences of N letters these fall into two groups for N large. One group of "high probability" sequences contains about

$$10^{RN}$$

sequences (where we have assumed R measured in digits per letter). All of these have substantially the same logarithmic probability. The remainder of the total of $10^{R_0 N}$ possible sequences are of very small probability. In fact their total probability approaches zero as N increases. The logarithm of the probability of an individual sequence in the high probability group is thus about -RN. In a precise statement of these results we must allow a certain fuzziness in R, i.e., replace R by $R \pm \epsilon$ where $\epsilon \longrightarrow 0$ as as $N \longrightarrow \infty$.

Reduction of a language to normal form is performed by properly matching the probabilities of sequences to the length of the corresponding sequences in the normal form. The "high probability" sequences are translated into short sequences and the remainder into longer sequences.

An example will clarify the results we have given. Let the language contain 4 letters A, B, C, D. In a sequence successive letters are chosen independently, the four letters having probabilities 1/2, 1/4, 1/8, 1/8, respectively. We have

$$R_0 = \log_2 4 = 2 \textit{ alternatives / letter}$$

and

$$\begin{aligned} R_1 &= R_2 = R_3 = \cdots = R = -(1/2 \log 1/2 + 1/4 \log 1/4 + 2/8 \log 1/8) \\ &= 1/2 + 1/2 + 6/8 = 7/4 \textit{ alternatives / letter} \end{aligned}$$

By a suitable transformation the average length of sequences can be reduced by the factor $\frac{7/4}{2} = 7/8$. A transformation to do it is the following. First we translate into a sequence of binary digits (0 or 1) by the following table

A	0
B	10
C	110
D	111

After this pairs of the binary digits are translated into the original alphabet as follows

00	A'
01	B'
10	C'
11	D'

For a typical sequence this works out as shown below:
Translation into binary digits:

A	B	C	A	B	A	C	B	B	D	A	A	D	A	D	A
0	10	110	0	10	0	110	10	10	111	0	0	111	0	111	0

Regrouping and translation back into letters:

01	01	10	01	00	11	01	01	01	11	00	11	10	11	10
B'	B'	C'	B'	A'	D'	B'	B'	B'	D'	A'	D'	C'	D'	C'

In this case there are 16 letters in the original and 15 in the final text. Thus due to the small redundancy and the shortness of the text only part of the saving is evident. In a long text however the full reduction of 1/8 would appear. This may be verified directly in this case. In a long text of N letters each letter will appear with about its appropriate frequency. Thus the number of binary digits will be about

$$N[1/2 \cdot 1 + 1/4 \cdot 2 + 1/8 \cdot 3 + 1/8 \cdot 3] = \frac{7}{4}N$$

since each A gives one binary digit, each B gives two, etc. The number of letters in the final text is half this since each pair of binary digits goes into one letter. Thus the reduction is by a factor 7/8.

It is also easy to see in this case that the binary digits are equally likely and independent, and from this that the final text letters are also.

This situation is more complicated for mixed 1anguages and we shall not enter into it here, We may note, however, that if

$$L = p_1 L_1 + p_2 L_2 + \cdots + p_n L_n$$

where L_i is pure with rate $R^{(i)}$, then the 1ong sequences of L fall into $(n + l)$ groups. The first n groups correspond to the n pure components. Those in group i number about

$$10^{R^{(i)}} N$$

and have logarithmic probability about

$$-R^{(i)} N$$

The last group contains all other sequences and has a small total probability.

7. Redundancy Characteristic of a Language

The form of the curve $D(N)$ as a function of N may be called the redundancy characteristic of the language. In a rough way it describes the way in which the redundancy appears. In Fig. 5 several types of characteristics are shown, all with the same final redundancy. The way in which this approach occurs is of importance in cryptography. For languages which reach the final redundancy at one or two letters (Curves 1 and 2) one type of cipher (ideal ciphers) can be used. For those which remain near zero out to fairly large N (like Curve 5) another type is appropriate. Natural languages are apt to show a characteristic more like 3, and this makes them difficult to encipher with security by simple means.

Examples:

1. A language in which successive letters are independent but with different probabilities has a characteristic of Type I.

2. Consider a language constructed as follows. First select 26^8 different sequences of letters, each 16 letters long from the 26^{16} possible sequences of this length. This should be a random selection. The 16-letter sequences chosen are the "words" of the language. Messages are random sequences of these "words." Such langunge has a characteristic like the Curve 5.

3. *A* language with digram structure only, such as Example C in Section 2 above, has a characteristic of the Type 2 in Fig. 5, reaching its final value at $N = 2$.

4. English has the characteristic 3 in Fig. 5.

The redundancy characteristic describes how the structure in the language is spread out. If the structure is localized, the curve rises rapidly to its final value. If there are long range in influences the asymptotic value is approached more slowly. If the structure is "locally" random the curve will remain near zero for small N.

8. Secrecy Systems

Before we can apply any mathematical analysis to secrecy systems, it necessary to idealize the situation suitably, and to define in a mathematically acceptable way what we shall mean by a secrecy system. A "schematic" diagram of a general secrecy system is shown in Fig. 6. At the transmitting end there are two information sources—a message source and a key source. The key source produces a particular key from among those which are possible in the system. This key is transmitted by some means, supposedly not interceptible, e.g., by messenger, to the receiving end, The message source produces a message (the "clear") which is enciphered, and the resulting cryptogram sent to the receiving end by a possibly interceptible means, for example radio. At the receiving end the cryptogram and key are combined in the decipherer to recover the message.

Evidently the encipherer performs a functional operation. If M is the message, K the key, and E the enciphered message, or cryptogram, we have

$$E = f(M, K)$$

i.e., E is a function of M and K. We prefer to think of this, however, not as a function of two variables but as a (one parameter) family of operations or transformations, and we write it

$$E = T_i M$$

The transformation T_i applied to message M produces cryptogram. E. The index i corresponds to the particular key being used. If there are m possible keys there will be m transformations in the family $T_1, T_2, \ldots T_m$.

At the receiving end it must be possible to recover M, knowing E and K. Thus the transformations in the family must have unique inverses

$$M = T_i^{-1} E$$

at any rate this inverse must exist uniquely for every E which can be obtained from an M with key i.

The key source can be thought of as a "probability machine," something which chooses from the possible keys according to a system of probabilities. Mathematically then, the keys (or the parameter of the family of transformations) belong to a probability or measure space.

Hence we arrive at the definition:

A secrecy system is a family of uniquely reversible transformations T_i of a message space Ω_M into a cryptogram space Ω_E, the parameter i belonging to a probability space Ω_K. Conversely any set of entities of this type will be called a "secrecy system".

The system can be visualized mechanically as a machine with one or more controls on it. A sequence of letters, the message, is fed into the input of the machine and a second series emerges at the output. The particular setting of the controls corresponds to the particular key being used. Some method must be prescribed for choosing the key from all the possible ones.

To make the problem mathematically tractable we shall assume that the enemy knows the system being used. That is, he knows the family of transformations T, and the probabilities of choosing various keys.

One might object to this as being unrealistic, in that the cryptanalyst often does not know what system was used or the probabilities of various keys. There are two answers to this objection.

Examples:

1. The assumption is actually the one ordinarily used in cryptographic studies. It is pessimistic and hence safe, but in the long run realistic (particularly in military work), since one must expect his system to be found out eventually through espionage, captured equipment, prisoners, etc. Thus, even when an entirely new system is devised, so that the enemy cannot assign any *a priori* probability to it without discovering it himself, one must still live with the expection of his eventual knowledge.

2. The restriction is much weaker them appears at first, due to our broad definition of what constitutes the system. Suppose a cryptographer intercepts a message and does not know whether a substitution, transposition, or Vigenère type cipher was used. He can consider this as being enciphered by a system in which part of the key is the specification of which of these types was used, the next part being the particular key for that type. These three different possibilities are assigned probabilities according to his best guess of of the *a priori* probabilities of the encipherer using the respective types of cipher.

A second possible objection to our definition of secrecy systems is that no account is taken of the common practice of inserting nulls in a message and the use of multiple substitutes. Thus there is not a unique $E = T_iM$, but actually the encipherer can choose at will among a number of different E's for the same message and key. This situation could be handled, but would only add complexity at the present stage, without altering any of the basic results. To define the more general secrecy system, one would add a second parameter to the transformations T_i which corresponds to the various choices of cryptograms corresponding to a given message and key. It is possible, but not always desirable, to consider this second parameter as part of the key, since it does not need to be transmitted to the receiving point.

We also assume that the enemy is in possession of a measure in the space Ω_M, the *a priori* probabilities of various messages. The same objection and essentially the same answers might be given to this assumption as to his knowledge of the transformations T_i. This measure, however, we do not consider as part of the secrecy system for reasons which will appear later. The secrecy system whose transformations are T_i will be denoted by T and this concept includes the space Ω_M on which T operates (without its measure), the transformations T_i and the spaces Ω_K and Ω_E, the former with its probability measure.

If the messages are produced by a Markoff process of the type described previously, the probabilities of various messages are determined by the structure of the Markoff process. For the present, however, we wish to take a more general view of the situation and regard the messages as merely an abstract set of entities with associated probabilities, not necessarily composed of a sequence of letters and not necessarily produced by a Markoff process.

It should be emphasized that throughout the paper a secrecy system means not one but a set of many transformations. After the key is chosen only one of these transformations is used and we might be led to define a secrecy system as a single transformation on a language.* The enemy, however, does not know what key was chosen and the "might have been" keys are as important for him as the actual one. Indeed it is only the existence of these other possibilities that gives the system

* A.A Albert in a paper presented at a Manhattan, Kansas, meeting of the American Mathematical Society (Nov. 22, 1941), entitled "Some Mathematical Aspects of Cryptography," has defined a ciphering system in this way. With this limited definition about all one can do is to describe and classify from the mathematical point of view various types of transformations.

any secrecy. Since the secrecy is our primary interest, we are forced to this rather elaborate concept of a secrecy system. This type of situation where possibilities are as important as actualities is almost the rule in games of strategy. The course of a chess game is largely controlled by threats which are *not* carried out. See also the "virtual existence" of unrealized imputations in von Neumann's theory of games.

There are a number of difficult epistemological questions connected with the theory of secrecy, or in fact with any theory which involves questions of probability (particularly *a priori* probabilities, Bayes' theorem, etc.) when applied to a physical situation. Treated abstractly, probability theory can be put on a rigorous logical basis with the modern measure theory approach.* As applied to reality, however, especially when "subjective" probabilities and unrepeatable experiments are concerned, there are many questions of logical validity. For example in the approach to secrecy made here, *a priori* probabilities of various keys are assumed known by the enemy cryptographer—how can one determine operationally if his estimates are correct, on the basis of his knowledge of the situation?

It may happen that the keys are chosen by the encipherer according to one system of probabilities, i.e., one measure in the key space Ω_K and that the enemy cryptanalyst estimates a second different system of probabilities Ω'_K in this space which are entirely reasonable in the light of his knowledge of the situation—which is correct? I believe *both* are correct. The calculation based on Ω_K leads to the solution when the enemy knows just how the keys *are* chosen and the solution based on $\Omega_0 K$ leads to solutions which are correct for a situation agreeing with the enemy's knowledge of the actual situation. It appears intuitively that the enemy's lack of knowledge can only do him harm, and probably this can be proved, but this question has not been investigated. In fact, we assume only one measure Ω_K in the key space. Similar remarks may be made regarding measure in the message space Ω_M.

* See J. L. Doob, "Probability as Measure," Annals of Math. Stat., v. 12, 1941, pp. 206–214.
A. Kolmogoroff, "Grundbegriffe der Wahrscheinlichkeits Rechnung," Ergebnisse der Mathematic, v.2, No. 3 (Berlin 1933).

Actually in practical situations, only extreme errors in *a priori* probabilities of keys and messages cause much error in the important parameters. This is because of the exponential behavior of the number of messages, etc., and the logarithmic measures employed.

With regard to the application of the mathematical theory of probability to physical situations there are two main theories or ways of setting up the correspondence. (1) The frequency theory. Probability is correlated with relative frequency of an event. This is the correspondence used by the practicing statistician, in principle by the physicist, etc. (2) The degree of belief approach. Probability is a subjective phenomena and measures one's degree of belief in the occurrence of an event. This approach is seen often in the work of historians, judges, and in everyday life. Although this latter approach has often been attacked as meaningless we cannot agree with this opinion. In the first place the intuitive approach can be given a rigorous mathematical foundation. This has been done in a very elegant way by B. O. Koopman.[†] Essentially one need only assume that a person be capable of making probability judgments (Event A is more or less probable than event B or they are equiprobable) and that his judgments be self consistent (e.g., if he judges A more probable than B and B more probable than C he should judge A more probable than C). One can even establish numerical values by the use of a "standard gauge," for example a roulette wheel, and thus relate the subjective and the frequency probabilities. In the second place, on progmatic grounds one can hardly ignore the subjective applications, since almost all of our everyday decisions are based on this sort of probability judgment. Cryptographic work involves both types of appliantions. In the use of frequency tables, significance tests etc., the cryptanalyst is following the frequency approach, In the "intuitive" methods of cryptanalysis (probable words etc.) the degree of belief approach is more in evidence.

We may remark that a single operation on a language which is reversible forms a degenerate type of secrecy system under our definition—a system with only one key of unit probability. Such a system has no secrecy—the cryptanalyst finds the message by applying the inverse of this transformation, the only one in the system, to the intercepted cryptogram. The decipherer and cryptanalyst is this case

[†] B. O. Koopman, "The Axioms and Algebra of Intuitive Probability." Annals of Mathematics, v.41, no.2, 1940, p.269.
"Intuitive Probabilities and Sequences," v.42. no.1, 1941, p.169.

possess the same information. In general, the only difference between the decipherer's knowledge and the enemy cryptanalyst's knowledge is that the decipherer knows the particular key being used, while the cryptanalyst only knows the *a priori* probabilities of the various keys in the set. The process of deciphering is that of applying the inverse of the particular transformation used in enciphering to the cryptogram. The process of cryptanalysis is that of attempting to determine the message (or the particular key) given only the cryptogram and the *a priori* probabilities of various keys and messages.

A system will be called "closed" if any possible cryptogram can be deciphered with any possible key. This means that the inverse transformations T_i^{-1} are all defined for every element in the cryptogram space.

We shall use the notation $|M|$ for the "size" of the message space:

$$|M| - \sum P(M) \log P(M)$$

where $P(M)$ is the probability of message M and th e sum is over all messages of just N letters. Thus $|M|$ is a function of N, and measures the amount of "choice" in the selection of an N letter message. For large N, $|M|$ is approximately RN. Similarly $|K|$ is the size of the key space

$$|K| = - \sum P(K) \log P(K)$$

the sum being over all keys.

9. Representation of Systems

A secrecy system can be represented in various ways. One which is convenient for illustrative purposes is a line diagram, as in Figs. 7, 10, 11. The possible messages are represented by points at the left and the possible cryptograms by points at the right. If a certain key, say key 1, transforms message M_2 into cryptogram E_4 then M_2 and E_4 are connected by a line labeled 1, etc. From each possible message there must be exactly one line emerging for each different key.

A second representation is by means of a rectangular array. This may be done in three different ways. For the closed system of Fig. 7, the three arrays are as follows:

K			
M	1	2	3
M_1	E_1	E_4	E_2
M_2	E_3	E_1	E_4
M_3	E_4	E_3	E_1
M_4	E_2	E_2	E_3

E				
M	E_1	E_2	E_3	E_4
M_1	1	3		2
M_2	2		1	3
M_3	3		2	1
M_4		$1, 2$	3	

K			
E	1	2	3
E_1	M_1	M_2	M_3
E_2	M_4	M_4	M_1
E_3	M_2	M_3	M_4
E_4	M_3	M_1	M_2

From the first of these message M_2 with key 3 yields cryptogram E_4. From the second M_1 is transformed into E_2 by key. 3. No key transforms M_1 into E_3 and either 1 or 2 transforms M_4 into E_2. From the third E_3 is deciphered by key 2 to give M_3. All of these arrays and the line diagram contain equivalent information—from any one the others can be derived.

These arrays and diagrams only describe the set of transformation in the system. To specify the system the probabilities of various keys must also be given. This may be done by merely listing the keys with the associated probabilities. Similarly the message source is not completely specified until the probabilities of the various messages are given.

A more common way of describing a system is to describe the set of transformations by telling what operations one performs on the message for an arbitrary key to obtain the cryptogram. Similarly one defines implicitly the probabilities for various keys by describing how a key is chosen, or what we know of the enemy's habits of key choice. The probabilities for messages are implicitly determined by stating our *a priori* knowledge of the enemy's language habits, the tactical situation (which will influence the probable content of the message) and any special information we may have regarding the cryptogram.

10. Notation

The following notation will generally be followed.
M = the message, also M_i, M_j, particular messages
K = the key E = the enciphered message or cryptogram
Ω_M = the set of all messages with associated probabilities, a probability space
Ω_K = the set of keys with associated probabilities, also a probability space

Ω_E = the cryptogram space, also a probability space, since the probabilities in Ω_M and Ω_K induce probabilities in Ω_E, for each cryptogram.
m_i = the i^th letter of the message
e_i = the i^th letter of the cryptogram
k_i = the i^th letter of the key when it can be so described

Generally P stands for a probability. Conditional probabilities are indicated with subscripts. Thus
$P(M)$ = probability of message M
$P(E)$ = probability of cryptogram E
$P(K)$ = probability of key K
$P_M(E)$ = conditional probability of E if message M is chosen
$P_E(M)$ = conditional probability of M if cryptogram E is intercepted, i.e. the *a posteriori* probability of M if E is observed.
Q = equivocation, a concept to be defined precisely later, which measures the uncertainty of some knowledge defined only by probabilities. We also have conditional equivocations, thus
$Q_M(K)$ is the equivocation of the key knowing the message.
$|K| = -\sum P(K)\log P(K)$ the size of the key space
$|M| = -\sum P(M)\log P(M)$ the size of the message space
$|E| = -\sum P(E)\log P(E)$ the size of the cryptogram space
m = number of different keys
N = number of intercepted letters
R_o = maximum information rate for a language
R = mean rate
$D = R_o - R$ = redundancy of a language
T, R, S, etc, = secrecy systems
T_i, R_i, S_i, etc. = particular transformation of these systems

11. Some Examples of Secrecy Systems

In this section a number of examples of ciphers will be given. These will often be referred to in the remainder of the paper for illustrative purposes.

1. Simple Substitution Cipher.

In this cipher each letter of the message is replaced by a fixed substitute, usually also a letter. Thus the message

$$M = m_1\ m_2\ m_3\ m_4\ \cdots$$

becomes

$$\begin{aligned} E &= e_1\ e_2\ e_3\ e_4 \\ &= f(m_1)\ f(m_2)\ f(m_3)\ f(m_4)\ \cdots \end{aligned}$$

where the function $f(m)$ is function with an inverse. The key is a permutation of the alphabet (when the substitutes are letters) e.g.
X G U A C D T B F H R S L M Q V Y Z W I E J O K N P
The first letter X is the substitute for A, G is the substitute for B, etc.

2. Transposition (Fixed Period d).

The message is divided into groups of length d and a permutation applied to the first group, the same permutation to the second group, etc. The permutation is the key and can be represented by a permutation of the first d integers. Thus for $d = 5$ we might have 2 3 1 5 4 as the permutation. This means that $m_1\ m_2\ m_3\ m_4\ m_5\ m_6\ m_7\ m_8\ m_9\ m_{10} \ldots$ becomes
$m_2\ m_3\ m_1\ m_5\ m_4\ m_7\ m_8\ m_6\ m_{10}\ m_9\ \ldots$. Sequential application of two or more transpositions will be called compound transposition. If the periods are $d_1, d_2, \ldots, d_s$ it is clear that the result is a transposition of period d, where d is the least common multiple of $d_1, d_2, d_3, \ldots, d_s$.

3. Vigenère, and Variations.

In this cipher the key consists of a series of d letters. There are written repeatedly below the message and the two added modulo 26 (considering the alphabet numbered from A = 0 to Z = 25). Thus

$$e_i = m_i + k_i \pmod{26}$$

where k_i is of period d in the index i.

For example with the key G A H we obtain

message	N O W I S T H E . . .
repeated key	G A H G A H G A . . .
cryptogram	T O D O S A N E . . .

The Vigenère of period 1 is called the Caesar cipher. It is a simple substitution in which each letter of M is advanced a fixed amount in the alphabet. This amount is the key, which may be any number from 0 to 25. The so-called Beaufort and Variant Beaufort are similar to the Vigenère, and encipher by the equations

$$e_i = k_i - m_i \pmod{26}$$

and

$$e_i = m_i - k_i \pmod{26}$$

respectively. The Beaufort of period one is called the reversed Caesar cipher.

The application of two or more Vigenères in sequences will be called the compound Vigenère. It has the equation

$$e_i = m_i = k_i + l_i + \ldots + s_i \pmod{26}$$

where $k_i, l_i, \ldots s_i$ in general have different periods. The period of their sum

$$k_i + l_i + \ldots + s_i$$

as in compound transposition, is the least common multiple of the individual periods.

4. Vernam System.[*]

When the Vigenère is used with an unlimited key, never repeating, we have the Vernam system, with

$$e_i = m_i + k_i \pmod{26}$$

the k_i being chosen at random and independently among 0, 1, . . . , 25. If the key is a meaningful text we have the “running key” cipher.

5. Bazeries Cylinder.

In this mechanical system 25 thick disks are used, each having a mixed alphabet stamped around the edge. These disks can be arranged in any order on a spindle, and the particular arrangement used constitutes the key. With the disks in their proper order, a

[*] G. S. Vernam, “Cipher Printing Telegraph Systems for Secret Wire and Radio Telegraphic Communications,” Journal Amer. Inst. of Elect. Eng. V. XLV. pp. 109–115, 1926.

message is enciphered by turning the disks so that the message appears on a line parallel to the axis of the spindle. Any other line of letters may then be chosen for the cryptogram. To decipher, the cryptogram is arranged on a line and the decipherer looks for another line which then makes sense.

6. Digram, Trigram, and N-gram substitution.

Rather than substitute for letters one can substitute for digrams, trigrams, etc. General digram substitution requires a key consisting of a permutation of the 26^2 digrams. It can be represented by a table in which the row corresponds to the first letter of the digram and the column to the second letter, entries in the table being the substitutes (usually also digrams).

7. Interrupted Key Vigenère.

The Vigenère and its variations can be used with an interrupted key. The sequence of key letters is started again at irregularly spaced points. Thus, if the entire key sequence is X P G H F T R S, one can interrupt irregularly to get

X P G H F T X P G X P G H F T R X P X P G . . .

The points of interruption can be determined in various ways. (1). Whenever a certain letter occurs in the clear. (2). Whenever a certain letter occurs in the cryptogram. (3) An interrupting letter, say J, can be reserved as a signal and the encipherer interrupts the key at his discretion. (4). No signal is used and the decipherer locates the interruptions by the meaningless text in the decipherment. In place of starting the key again at each interruption one can omit letters of it or reverse the direction of progression. There are many variations and combinations of these methods.

8. Single Mixed Alphabet Vigenère.

This is a simple substitution followed by a Vigenère.

$$e_i = f(m_i) + k_i$$

$$m_i = f^{-1}(e_i - k_i)$$

The "inverse" of this system is a Vigenère followed by simple substitution

$$e_i = g(m_i + k_i)$$

$$m_i = g^{-1}(e_i) - k_i$$

9. Vigenère with Progressing Key.

The period of a Vigenère can be expanded by adding a fixed number t to the key at each appearance—thus the n^{th} group is enciphered by the equation

$$e_i = m_i + k_i + nt$$

Also this can be varied by adding t and s alternately to the key, etc.

10. Matrix System.*

One method of n-gram substitution is to operate on successive n-grams with a matrix having an inverse. The letters are assumed numbered from 0 to 25, making them elements of an algebraic ring. From the n-gram $m_1, m_2, \ldots m_n$ of message, the matrix a_{ij} gives an n-gram of cryptogram

$$e_i = \sum_{j=1}^{n} a_{ij} m_j \qquad i = 1, \ldots, n$$

The matrix a_{ij} is the key, and deciphering is performed with the inverse matrix. The inverse matrix will exist if and only if the determinant $|a_{ij}|$ has an inverse element in the ring.

11. The Playfair Cipher.

This is a particular type of digram substitution governed by a mixed 25 letter alphabet written in a 5 × 5 square. (The letter J is often dropped in cryptographic work—it is very infrequent, and when it occurs can be replaced by I.) Suppose the key square is as shown below

L	*Z*	*Q*	*C*	*P*
A	*G*	*N*	*O*	*U*
R	*D*	*M*	*I*	*F*
K	*Y*	*H*	*V*	*S*
X	*B*	*T*	*E*	*W*

* See L.S. Hill, "Cryptography in an Algebraic Alphabet," American Math. Monthly, v. 36, No. 6, 1, 1929, pp. 306-312, Also "Concerning Certain Linear Transformation Apparatus of Cryptography," v. 38, No. 3, 1931, pp. 135-154.

The substitute for a digram AC, for example, is the pair of letters at the other corners of the rectangle defined by A and C, i.e., LO, the L taken first since it is above A. If the digram letters are on a horizontal line as RI, one uses the letters to their right DF; RF becomes DR. If the letters are on a vertical line, the letters below them are used. Thus PS becomes UW. If the letters are the same nulls may be used to separate them or one may be omitted, etc.

12. Multiple Mixed Alphabet Substitution.

In this cipher there are a set of d simple substitutions which are used in sequence. If the period d is four

$$m_1\ m_2\ m_3\ m_4\ m_5\ m_6 \cdots$$

becomes

$$f_1(m_1)\ f_2(m_2)\ f_3(m_3)\ f_4(m_4)\ f_1(m_5)\ f_2(m_6)\ \cdots$$

13. Autokey Cipher.

A Vigenère type system in which either the message itself or the resulting cryptogram is used for the "key" is called an autokey cipher. The encipherment is started with a "priming key" (which is the entire key in our sense) and continued with the message or cryptogram displaced by the length of the priming key as indicated below with the priming key COMET. The message used as "key".

MESSAGE	S E N D S U P P L I E S . . .
KEY	C O M E T S E N D S U P . . .
CRYPTOGRAM	U S Z H L M T C O A Y H . . .

The cryptogram used as "key",

MESSAGE	S E N D S U P P L I E S . . .
KEY	C O M E T U S Z H L O H . . .
CRYPTOGRAM	U S Z H L O H O S T S . . .

14. Fractional Ciphers.

In these, each letter is first enciphered into two or more letters or numbers and these symbols are somehow mixed (e.g. by transposition). The result may then be retranslated into the original alphabet. Thus using a mixed 25 letter alphabet for the key we may translate letters into two digit quinary numbers by the table

	0	1	2	3	4
0	L	Z	Q	C	P
1	A	G	N	O	U
2	R	D	M	I	F
3	K	Y	H	V	S
4	X	B	T	E	W

Thus B becomes 41. After the resulting series of numbers is transposed in some way they are taken in pairs and translated back into letters.

15. Codes.

In codes words (or sometimes syllables) are replaced by substitute letter groups. Sometimes a cipher of one kind or another is applied to the result.

12. Valuations of Secrecy Systems

There are a number of different criteria that should be applied in estimating the value of a proposed secrecy system. The more important of these are:

1. Amount of Secrecy.

There are some systems that are perfect—the enemy is no better off after intercepting any amount of material than before. Other systems, although giving him some information, do not yield a unique "solution" to intercepted cryptograms. Among the uniquely solvable systems, there are wide variations in the amount of labor required to effect this solution, and the amount of material that must be intercepted to make the solution unique.

2. Size of Key.

The key must be transmitted by non-interceptible means from transmitting to receiving ends. Sometimes it must be memorized. It is desirable then to have the key as small as possible.

3. Complexity of Enciphering and Deciphering Operations.

These should, of course, be as simple as possible. If they are done manually, complexity leads to loss of time, errors, etc. If done mechanically, complexity leads to large expensive machines.

4. Propagation of Errors.

In certain types of secrecy systems an error of one letter in enciphering or transmission leads to a large amount of error in the deciphered text. The errors are spread out by the deciphering operation, causing the loss of much information and frequent need for repetition of the cryptogram. It is naturally desirable to minimize this error expansion.

5. Expansion of Message.

In some types of secrecy systems the size of the message is increased by the enciphering process. This undesirable effect may be seen in systems where one attempts to swamp out message statistics by the addition of many nulls, or where multiple substitutes are used. It also occurs in many "concealment" types of systems (which are not usually secrecy systems in the sense of our definition).

13. Equivalence Classes in the Key Space

It may happen that in a ciphering system two or more different keys, say keys 1, 2, and 7, are equivalent. By this we mean that for every M

$$T_1M = T_2M = T_7M$$

These keys will not be considered as distinct but will be thrown into an equivalence class. It is clear that the cryptanalyst can never determine which particular one of these was used but only (at best) the class. The probability for the class is of course the sum of the probabilities of the different keys in the class.

As an example, in the Playfair cipher with the system given above, the following are equivalent key squares.

G	H	X	P	Y		E	C	I	Z	F
Z	F	E	C	I		N	R	D	L	O
L	O	N	R	D		V	S	Q	T	A
T	A	V	S	Q		W	B	M	K	U
K	U	W	B	M		X	P	Y	G	H

We can think of the possible equivalence classes in this case as arrangements of a 25 letter alphabet on a 5 × 5 square drawn on an oriented torus. The number of different keys is not 25! but $25!/5^2 = 24!$

When we say that two secrecy systems are the same we mean that they consist of the same set of transformations T_i, with the same message and cryptogram space (range and domain) and the same probabilities for the different keys (after all identical transformations are put in the same equivalence class).

14. The Algebra of Secrecy Systems

If we have two secrecy systems T and R we can often combine them in various ways to form a new secrecy system S. If T and R have the same domain (message space) we may form a kind of "weighted sum,"

$$S = pT + qR$$

where $p + q = 1$. This operation consists of first making a preliminary choice with probabilities p and q determining which of T and R is used. This choice is part of the key of S. After this is determined T or R is used as originally defined. The total key of S must specify which of T and R is used and which key of T (or R) is used.

If T consists of the transformations $T_1, \ldots, T_m$ with probabilities $p_1, \ldots, p_m$ and R consists of $R_1, \ldots, R_k$ with probabilities $q_1, \ldots, q_k$, then $S = pT + qR$ consists of the transformations $T_1, T_2, \ldots, T_m$, $R_1, \ldots, R_k$ with probabilities pp_1, $pp_2, \ldots, pp_m$, $qq_1, qq_2, \ldots, qq_k$ respectively.

More generally we can form the sum of a number of systems.

$$S = p_1 T + p_2 \ R + \cdots + p_m U \ \sum p_i = 1$$

We note that any system T can be written as a sum of fixed operations

$$T = p_1 T_1 + p_2 T_2 + \cdots + p_m T_m$$

T_i being a definite enciphering operation of T corresponding to key choice i, which has probability p_i.

A second way of combining two secrecy systems is by taking the "product", shown schematically in Fig. 8. Suppose T and R are two systems and the domain (language space) of T can be identified with the range (cryptogram space) of R. Then we can apply first R to our language and then T to the result of this enciphering process. This gives a resultant operation S which we write as a product

$$S = TR$$

The key for S consists of both keys of T and R which are assumed chosen according to their original probabilities and independently. Thus if the m keys of T are chosen with probabilities

$$p_1 \ p_2 \cdots p_m$$

and the n keys of R have probabilities

$$p'_1 \ p'_2 \cdots p'_n$$

then S has mn keys (at most; there may and often will be equivalence classes) with probabilities $p_i p'_j$. This type of product encipherment is often used; for example one follows a substitution by a transposition or a transposition by a Vigenère, or applies a code to the text and enciphers the result by substitution, transposition, fractionation, etc.

A more special type of product may be defined in case both T and R have keys of the same size which may be put in one-to-one correspondence with the same probabilities for corresponding keys. This may be called the "inner product", in contrast with the above which may be more completely described as an "outer product" (these names are derived from a rough analogy with the concepts of tensor analysis). In the inner product, written

$$S = T \cdot R$$

and indicated schematically in Fig. 9, the same key (or corresponding keys) are used for both T and R chosen with the common probability.

For example one may construct a transposition cipher whose key is a permutation of the alphabet, each permutation being equally likely, and apply first this and then a substitution based on the same permutation. One also sees this situation in certain geometrical types of tranposition ciphers where the text is written into a square and a permutation based on a key word applied first to the columns and then to the rows of the square.

It may be noted that multiplication (either kind) is not in general commutative, (we do not always have $RS = SR$) although in special cases such as substitution and transposition it is. Since it represents an operation it is definitionally associative. That is $R(ST) = (RS)T = RST$. Furthermore we have the laws

$$p(p'T + q'R) + qS = pp'T + pq'R + qS$$

(weighted associative law for addition)

$$T(pR + qS) = pTR + qTS$$

$$(pR + qS)T = pRT + qST$$

(right and left hand distributive laws)
and

$$p_1T + p_2T + p_3R = (p_1 + p_2)T + p_3R$$

Finally with regard to this algebraic structure of secrecy operations, we note that every closed secrecy system T has an "inverse" T' obtained by interchanging the E and M spaces, with key probabilities the same, and

$$(TRS)' = S'R'T'$$

$$(pT + qR)' = pT' + qR'$$

Note that TT' is not in general the identity (this is the reason we do not write T^{-1}).

A system whose M and E spaces can be identified, a very common case as when letter sequences are transformed into letter sequences, may be termed endomorphic. An endomorphic system T may be raised to a power T^n.

A secrecy system T whose outer product with itself is equal to T, i.e. for which

$$TT = T$$

will be called idempotent. For example simple substitution, transposition of period p, Vigenère of period p (all with each key equally likely) are idempotent.

The set of all endomorphic secrecy systems defined in a fixed message space constitute an "algebraic variety," that is, a kind of algebra, using the operations of addition and multiplication. In fact, the properties of addition and multiplication which we have discussed lead to the following result.

Theorem 1: The set of endomorphic ciphers with the same message space and the two combining operations of weighted addition and outer multiplication form a linear associative algebra with a unit element, apart from the fact that the coefficients in a weighted addition must be non-negative and sum to unity.

It should be emphasized that these combining operations of addition and multiplication apply to secrecy systems as a whole. The product of two systems TR should not be confused with the product of the transformations in the systems T_iR_j, which also appears often in this work. The former TR is a secrecy system, i.e. a set of transformations with associated probabilities; the latter is a particular transformation. Further the sum of two systems $pR + qT$ is a system—the sum of two transformations is not defined. The systems T and R may commute without the individual T_i and R_j commuting , e.g. if R is a Beaufort system of a given period, all keys equally likely,

$$R_iR_j \neq R_jR_i$$

in general, but of course RR does not depend on its order; actually

$$RR = V$$

the Vigenère of the same period with random key. On the other hand, if the individual T_i and R_j of two systems T and R commute, then the systems commute.

It is rather surprising to find an algebraic variety with as much structure as a linear associative algebra in which the elements have the complexity of ciphers. In Hilbert space theory, for example, one has a linear associative algebra, but the elements of the algebra are transformations. Here the elements are *sets* of transformations with a probability space associated with the transformation parameter.

These combining operations give us ways of constructing many new types of secrecy systems from certain ones, such as the examples given. We may also use them to describe the situation facing a cryptanalyst when attempting to solve a cryptogram of unknown type. He is, in fact, solving a secrecy system of the type

$$T = p_1 A + p_2 B + \cdots + p_r S + p' X \sum p = 1$$

where the $A, B, \ldots, S$ are known types of ciphers, with the p_i their *a priori* probabilities in this situation, and $p'X$ corresponds to the possibility of a completely new unknown type of cipher.

In weighted addition the key size of the result is given by

$$\begin{aligned} |K| &= -\textstyle\sum_i p_i p_i' \log p p_i' - \sum q p_i'' \log q p_i'' \\ &= p|K_1| + q|K_2| - (p \log p + q \log q) \\ &= p|K_1| + q|K_2| + |K_3| \end{aligned}$$

i.e. the weighted mean of the two keys plus the size of the p, q key. This is only in case there are no equivalences; if there are it will always be less.

For the outer product the key size is

$$|K| \leq |K_1| + |K_2|$$

with equality only when there are no equivalences. In the inner product

$$|K| \leq |K_1| = |K_2|$$

with equality under the same condition.

15. Pure and Mixed Ciphers

Certain types of ciphers, such as the simple substitution, the transposition of a given period, the Vigenère of a given period, the mixed alphabet Vigenère, etc. (all with each key equally likely) have a certain homogeneity with respect to key. Whatever the key, the enciphering, deciphering and decrypting processes are essentially the same. This may be contrasted with the cipher

$$pS + qT$$

where S is a simple substitution and T a transposition of a given period. In this case the entire system changes for enciphering, deciphering and decryptment, depending on whether the substitution or transposition was used.

The cause of the homogeneity in certain ciphers stems from the group property—we notice that in the above examples of homogeneous ciphers the product of any two transformations in the set $T_i T_j$ is equal to a third transformation T_k in the set, while T_i S_j does not equal any transformation in the cipher

$$pS + qT$$

which contains only substitutions and transpositions, no products.

We might define a "pure" cipher, then, as one whose T_i formed a group. This, however, would be too restrictive since it requires that the E space be the same as the M space, i.e. that the system be endomorphic. The fractional transposition is as homogeneous as the ordinary transposition without being endomorphic. The proper definition is the following. A cipher T is pure if for every T_i, T_j, T_k, there is a T_s such that

$$T_i T_j^{-1} \ \ T_k = T_s$$

and every key is equally likely. Otherwise the cipher is mixed. The systems of Fig. 7 are mixed. Fig. 10 is pure if all keys are equally likely.

Theorem 2: In a pure cipher the operations $T_i^{-1} T_j$ which transform the message space into itself form a group whose order is m, the number of different keys.

For

$$T_j^{-1} T_k T_k^{-1} T_j = I$$

so that each element has an inverse, also the associative law is true since these are operations, and the group property follows from

$$T_i^{-1} T_j T_k^{-1} T_\ell = T_s^{-1} T_k T_k^{-1} T_\ell = T_s^{-1} T_\ell$$

using our assumption that $T_i^{-1} T_j = T_s^{-1} T_k$ for some s.

The operation $T_i^{-1} T_j$ means, of course, enciphering the message with key j and then deciphering with key i which brings us back to the message space. If T is endomorphic, i.e the T_i themselves transform the space Ω_M into itself (as is the case with most ciphers, where both the message space and the cryptogram space consist of sequences of letters), and the T_i are a group and equally likely, then T is pure, since

$$T_i T_j^{-1} T_k = T_i T_r = T_s$$

Theorem 3: The outer product of two pure ciphers which commute is pure.

For if T and R commute $T_i R_j = R_\ell T_m$ for every i, j with suitable ℓ, m, and

$$\begin{aligned} T_i R_j (T_k R_\ell)^{-1} &= T_i R_j R_\ell^{-1} T_k^{-1} T_m R_n \\ &= R_u R_v^{-1} R_w T_r T_s^{-1} T_t \\ &= R_h T_g \end{aligned}$$

The commutation condition is not necessary, however, for the product to be a pure cipher.

A system with only one key, i.e. a single definite operation T_1, is pure since the only choice of indices is

$$T_1 T_1^{-1} T_1 = T_1$$

Thus the expansion of a general cipher into a sum of such simple transformations also exhibits it as a sum of pure ciphers.

An examination of the example of a pure cipher shown in Fig. 5 discloses certain properties. The messages fall into certain subsets which we will call residue classes, and the possible cryptograms are divided into corresponding residue classes. There is at least one line from each message in a class to each cryptogram in the corresponding class, and no line between classes which do not correspond. The number of messages in a class is a divisor of the total number of keys. The number of lines “in parallel” from a message M to a cryptogram in the corresponding class is equal to the number of keys divided by the number of messages in the class containing the message (or cryptogram). It is shown in the appendix that these hold in general for pure ciphers. Summarized in a more formal statement we have

Theorem 4: In a pure system the messages can be divided into a set of “residue classes” $C_1, C_2, \ldots, C_3$ and the cryptograms into a corresponding set of residue classes $C_1, C_2, \ldots C_s$ with the following properties

(1) The message residue classes are mutually exclusive and collectively contain all possible messages. Similarly for the cryptogram residue classes.

(2) Enciphering any message in C_i with any k produces a cryptogram in C'_i. Deciphering any cryptogram in C'_i with any key leads to a message in C_i.

(3) The number of messages in C'_i, say φ, is equal to the number of cryptograms in C'_i and is a divisor of k the number of keys.

(4) Each message in C_i can be enciphered into each cryptogram in C'_i by exactly $\frac{k}{\varphi i}$ different keys. Conversely for decipherment.

The importance of the concept of a pure cipher (and the reason for the name) lies in the fact that for them all keys are essentially the same. Whatever key is used for a particular message, the *a posteriori* probabilities of all messages are identical. To see this, note that two different keys applied to the same message lead to two cryptograms in the same residue class, say C_i. The two cryptograms therefore could each be deciphered by $\frac{k}{\varphi_i}$ keys into each message in C_i and into no other possible messages. All keys being equally likely the *a posteriori* probabilities of various messages are thus

$$P_E(M) = \frac{P(M)P_M(E)}{\sum P_M P_M(E)} = \frac{P(M)}{P(C_i)}$$

where M is in C_i, E is in C_i' and the sum is over all messages in C_i. If E and M are not in corresponding residue classes $P_E(M) = 0$. Similarly it can be shown that the *a posteriori* probabilities of the different keys are the same in value but these values are associated with different keys when a different key is used. The same set of values of $P_E(K)$ have undergone a permutation among the keys. Thus we have the result

Theorem 5: In a pure system the *a posteriori* probabilities of various messages $P_E(M)$ are independent of the key that is chosen. The *a posteriori* probabilities of the keys $P_E(K)$ are the same in value but undergo a permutation with a different key choice.

Roughly we may say that any key choice leads to the same cryptanalytic problem in a pure cipher. Since the different keys all result in cryptograms in the same residue class this means that all cryptograms in the same residue class are cryptanalytically equivalent—they lead to the same *a posteriori* probabilities of messages and, apart from a permutation, the same probabilities of keys.

As an example of this, simple substitution with all keys equally likely is a pure cipher. The residue class corresponding to a given cryptogram E is the set of all cryptograms that may be obtained from E operations $T_j T_k E$. In this case $T_j T_k^{-1}$ is itself a substitution and hence any substitution on E gives another member of the same residue class. Thus if the cryptogram is

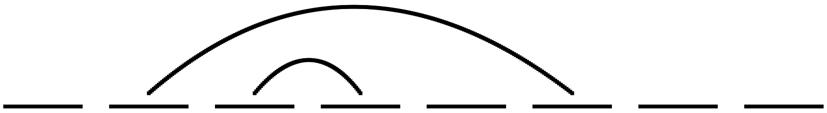
$$E = XCPPGCFQ$$

then

$$E_1 = RDHHGDSN$$

$$E_2 = ABCCDBEF$$

etc. are in the same residue class. It is obvious in this case that these cryptograms are essentially equivalent. All that is of importance in a simple substitution with random key is the *pattern* of letter repetitions, the actual letters being dummy variables. Indeed we might dispense with them entirely indicating the pattern of repetitions in E as follows:*



This notation describes the residue class but eliminates all information as to the specific member of the class. Thus it leaves precisely that information which is cryptanalytically pertinent. This is related to one method of attacking simple substitution ciphers—the method of pattern words.

In the Caesar type cipher only the first difference mod 26 of the cryptogram are significant. Two cryptograms with the same Δe_j are in the same residue class. One breaks this cipher by the simple process of writing down the 26 members of the message residue class and picking out the one which makes sense.

The Vigenère of period d with random key is another example of a pure cipher. Here the message residue class consists of all sequences with the same first differences for letters separated by distance d as the cryptogram. For $d = 3$ the residue class is defined by

$$\begin{gathered} m_1 - m_4 = e_1 - e_4 \\ m_2 - m_5 = e_2 - e_5 \\ m_3 - m_6 = e_3 - e_6 \\ m_4 - m_7 = e_4 - e_7 \\ \cdot \\ \cdot \\ \cdot \end{gathered}$$

* Suggested by a notation used by Quine in Symbolic Logic.

where $E = e_1, e_2, \ldots$ is the cryptogram and $m_1, m_2, \ldots$ is any M in the corresponding residue class.

In the transposition cipher of period d with random key, the residue class consists of all arrangements of the e_i in which no e_i is moved out of its block of length d, and any two e_i at a distance d remain at this distance. This is used in breaking these ciphers as follows. The cryptogram is written in successive blocks of length d, one under another as below ($d = 5$):

$$\begin{array}{ccccc} e_1 & e_2 & e_3 & e_4 & e_5 \\ e_6 & e_7 & e_8 & e_9 & e_{10} \\ e_{11} & e_{12} & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \end{array}$$

The columns are then cut apart and rearranged to make sense. When the columns are cut apart, the only information remaining is the residue class of the cryptogram.

Theorem 6: If T is pure then $T_i T_j^{-1} T = T$ where $T_i T_j$ are any two transformations of T. Conversely if this is true for any $T_i T_j$ in a system T then T is pure.

The first part of this theorem is obvious from the definition of a pure system. To prove the second part we note first that if $T_i T_j^{-1} T = T$ then $T_i T_j^{-1} T_s$ is a transformation of T.
It remains to show that all keys are equiprobable. We have $T = \sum_s p_s T_s$ and

$$\sum_s p_s T_i T_j^{-1} T_s = \sum_s p_s T_s$$

the term in the left hand sum with $s = j$ yields $p_j T_i$. The only term in T_i on the right is $p_i T_i$. Since all coefficients are non negative it follows that

$$p_j \leq p_i.$$

The same argument holds with i and j interchanged and consequently

$$p_j = p_i$$

and T is pure. Thus the condition that $T_i T_j^{-1} T = T$ might be used as an alternative definition of a pure system.

The property of purity in a system is connected with idempotence. Thus consider the system $S = TT'$ where T is pure. We have

$$T_i T_j^{-1} T_s T_r^{-1} = T_i T_\ell^{-1} T_r T_r^{-1} = T_i T_\ell^{-1}$$

so that the transformations of S^2 are the same as those of S, and since both S and S^2 are pure we have

$$S = S^2$$

Theorem 7: If T is pure $S = TT'$ is pure and $S^2 = S$.

An endomorphic system T which satisfies the condition $T_i T_j = T_s$ (but not necessarily with all key probabilities equal) can be shown to approach a pure cipher on raising to a high power, namely the one with the same transformations, but with all probabilities equalized. In fact the probabilities for T^{n+1} are derived from those for T^n by a Markoff process of a special type due to the group property. This special type always approaches the limit of equalized probabilities. This same argument applies more generally We have

Theorem 8: Let T be any endomorphic cipher. If T^n approaches any limit at all, which will necessarily occur if all the transformations of T^n lie in a finite set (no matter how large n) and the transformations of T include the identity then this limit will be a pure cipher.

As an example consider the cipher

$$R = pT + qS$$

where T is transposition with random key and S substitution with random key. We have

$$S^2 = S$$

$$T^2 = T$$

$$ST = TS$$

and hence any product of T's and S's such as $T\ S\ T\ T\ T\ S\ S$ reduces to ST. Thus

$$R^n = p^nT + q^nS + (1 - p^n - q^n)ST$$

As $n \to \infty$ the first two terms approach zero and

$$\lim_{n \to \infty} R^n = ST$$

The concepts of pure and mixed languages and pure and mixed ciphers have an application in practical cryptanalysis, if we interpret them somewhat loosely. When a cryptographer starts work on a cryptogram, his first job is to determine the original language. Approximately then he is determining the pure component of the general language space

$$L = p_1L_1 + p_2L_2 + \cdots + p_nL_n$$

where L_1 say is English, L_2 German, etc. Of course these are not pure but the different components of them are fairly close together in statistical structure.

The second thing a cryptographer does is to determine the "type" of cipher that was used—usually this is about the same as finding the pure component in the general cipher system

$$R = p_1S + p_2T + p_3V + ...$$

where S say is simple substitution, T is transposition, etc. A Vigenère V of unknown period is not a pure cipher but the decomposition

$$V = p_1V_1 + p_2V_2 + p_3V_3 + ...$$

where V_i is of period i, is into pure components (if all keys are equally likely for any period). In solving a Vigenère the first problem is to determine the period. The same is true in transposition.

The reason for this initial isolation of pure or nearly pure language and cipher is that only then can a simple meaningful statistical analysis be carried out.

16. Involutory Systems

If every transformation in a system T is its own inverse, i.e., if

$$T_iT_i = I$$

for every i, the system will be called involutory. Such systems are important practically since the enciphering and deciphering operations are then identical. This leads to simplified instructions to cryptographic clerks in manual operation, or in mechanical cases the same machine with the same key setting may be used for both operations.

Examples: In simple substitution we may limit our transformations to those in which when letter Θ is the substitute for φ, φ is the substitute for Θ. Another example is the Beaufort cipher.

If T is involutory, so is the system whose operations are

$$S_j T_i S_j^{-1}$$

since

$$S_j T_i S_j^{-1} (S_j T_i S_j^{-1}) = S_j T_i S_j^{-1} S_j T_i S_j^{-1} = I$$

17. Similar and Weakly Similar Systems

Two secrecy systems V and S will be said to be *similar* if there exists a transformation A having an inverse A^{-1} such that

$$R = AS$$

This means that enciphering with R is the same as enciphering with S and then operating on the result with the transformation A. If we write $R{\approx}S$ to mean R is similar to S then it is clear that $R{\approx}S$ implies $S{\approx}R$. Also $R{\approx}S$ and $S{\approx}T$ imply $R{\approx}T$ and finally $R{\approx}R$. These are summarized in mathematical terminology by saying that similarity is an equivalence relation.

The cryptographic significance of similarity is that if $R{\approx}S$ then R and S are equivalent from the cryptanalytic point of view. Indeed if a cryptanalyst intercepts a cryptogram in system S he can transform it to one in system R by merely applying the transformation A to it. A cryptogram in system R is transformed to one in S by applying A^{-1}. If R and S are applied to the same language or message space, there is a one-to-one correspondence between the resulting cryptograms. Corresponding cryptograms give the same distribution of *a posteriori* probabilities for all messages.

If one has a method of breaking the system R then any system S similar to R can be broken by reducing to R through application of the operation A. This is a device that is frequently used in practical cryptanalysis.

Examples: As a trivial example, simple substitution where the substitutes are not letters but arbitrary symbols is similar to simple substitution using letter substitutes. A second example is the Caesar and the reversed Caesar type ciphers. The latter is sometimes broken by first transforming into a Caesar type. The Vigenère, Beaufort and Variant Beaufort are all similar, when the key is random. The "autokey" cipher primed, with the key $K_1K_2...K_d$ is similar to a Vigenère type with the key alternately added and subtracted $\pmod{26}$. The transformation A in this case is that of "deciphering" the autokey with a series of d A's for the priming key.

Two systems R and S are *weakly similar* if there exist two transformations A and B having inverse A^{-1} and B^{-1} with

$$R = ASB$$

This means that system R is the same as applying first B to the language, then S, and finally A. This relation is also an equivalence relation.

Finding a method of solution for system R with language L is equivalent to finding a solution for S with language B L.

We may note that if R is pure and S is weakly similar to R then S is pure. This follows from

$$R_iR_j^{-1}R_k = R_\ell$$
$$R_i = AS_iB$$
$$R_j^{-1} = B^{-1}S_j^{-1}A^{-1}$$
$$R_k = AS_kB$$

where we assume corresponding transformations in R and S to have the same subscripts. Hence

$$\begin{aligned} R_i R_j^{-1} &= A S_i S_j S_k B = R_\ell \\ S_i S_j^{-1} S_k &= A^{-1} R_\ell B^{-1} \\ &= S_\ell \end{aligned}$$

and S is therefore pure.

PART II

Theoretical Secrecy

Introduction

We now consider problems connected with the "theoretical secrecy" of a system. How immune is a system to cryptanalysis when the cryptanalyst has unlimited time and manpower available for the analysis of cryptograms? Does a cryptogram *have* a unique solution (even though it may require an impractical amount of work to find it) and if not how many reasonable solutions does it have? How much text in a given system must be intercepted before the solution becomes unique? Are there systems which never become unique in solution no matter how much enciphered text is intercepted? Are there systems for which no information whatever is given to the enemy no matter how much text is intercepted?

18. Perfect Secrecy

Let us suppose the possible messages are finite in number M_1, ..., M_n and have *a priori* probabilities $P(M_1), ..., P(M_n)$, and that these are enciphered into the possible cryptograms $E_1...E_m$ by

$$E = T_i M$$

The cryptanalyst intercepts a particular E and can then calculate the *a posteriori* probabilities for the various messages, $P_E(M)$. It is natural to define *perfect secrecy* by the condition that for all E, the *a posteriori* probabilities are equal to the *a priori* probabilities independently of the values of these. In this case, intercepting the message has given the cryptanalyst no information.[‡]Any action of his which depends on the information contained in the cryptogram cannot be altered, for all of his probabilities as to what the cryptogram contains remain unchanged. On the other hand, if the condition is not satisfied there will exist situations in which the enemy has certain *a priori* probabilities, and certain key and messages are chosen where the enemy's probabilities do change. This in turn may affect his actions and thus perfect secrecy has not been obtained.

[‡] A purist might object that the enemy has obtained a bit of information in that he knows a message was sent. This may be answered by having among the messages a "blank" corresponding to "no message." If no message is originated the blank is enciphered and sent as a cryptogram. Then even this modicum of remaining information is eliminated.

Hence the definition given is necessarily required by our ideas of what perfect secrecy should mean.

A necessary and sufficient condition for perfect secrecy can be found as follows. We have by Bayes' theorem

$$P_E(M) = \frac{P(M)P_M(E)}{P(E)}$$

and this must equal $P(M)$ for perfect secrecy. Hence either $P(M) = 0$, a solution that must be excluded since we demand the equality independent of the values of $P(M)$, or

$$P_M(E) = P(E)$$

for every M and E. Conversely if $P_M(E)$=$P(E)$ then

$$P_E(M) = P(M)$$

and we have perfect secrecy. Thus we have the result:

Theorem 9: A necessary and sufficient condition for perfect secrecy is that

$$P_M(E) = P(E)$$

for all M and E. That is $P_M(E)$ must be independent of M.

The probability of all keys that transform M_i into a given cryptogram E is equal to that of all keys transforming M_j into the same E.

Now there must be as many E's as there are M's, since fixing i, T_i gives a one-to-one correspondence between all the M's and some of the E's. For perfect secrecy $P_M(E) = P(E) \neq 0$ for any of these E's and any M. Hence there is at least one key transforming any M into any of the E's. But all the keys from a fixed M to different E's must be different, and therefore the number of different keys is at least as great as the number of M's. It is possible to obtain perfect secrecy with no more, as one shows by the following example. Let the M_i be numbered 1 to n and the E_i the same, and using n keys let

$$T_i M_j = E_s$$

where $s = i - j \pmod{n}$. In this case we see that $P_E(M) = \frac{1}{n} = P(E)$ and we have perfect secrecy. An example is shown in Fig 11 with $n = 5$.

These perfect systems in which the number of cryptograms, the number of messages, and the number of keys are all equal are characterized by the properties that (1) each M is connected to each E by exactly one line, (2) all keys are equally likely. Thus the three matrix representations of the system are "latin squares".

We have then concealed completely an amount of information at most *log n* with a size of key *log n*. This is the first example of a general principle which we will often see, that there is a limit to what can obtain with a given key size—the amount of uncertainty we can introduce into the solution of the cryptogram cannot be greater than the key size. Here we have concealed all the information but the key size is as large as the message space.

We now consider the case where $|M|$ is infinite; in fact suppose the message generated as an unending sequence of letters by a Markoff process. The maximum rate of this source is R_O. It is clear from our results above that no finite key will give perfect secrecy. We suppose then that the key source generates key also in the same manner, i.e. as an infinite sequence of symbols with a *mean* rate R_K. Suppose that only a certain length of key L_K is needed to encipher and decipher a length L_M of message.

Theorem 10: For perfect secrecy (when the *a priori* probabilities of various messages can be anything), for large L

$$R_O L_M \leq R_K L_K$$

and the rate $(R_K + \varepsilon)$ is asymptotically sufficient.

This may be proved by the same method (essentially) as the finite case. This case is realized by the Vernam system.

These results have been deduced on the basis of unknown or arbitrary *a priori* probabilities for the messages. The key required for perfect secrecy depends then on the total number of possible messages, or on the maximum rate R_O of the message source.

One would suspect that if the message space has fixed known statistics, so that it has a definite mean rate R of generating information, then the amount of key needed could be reduced in an average sense in just this ratio $\frac{R}{R_O}$, and this is indeed true. In fact the message can be passed through a transducer which transforms it into a normal form and reduces the expected length in just this ratio, and then a Vernam system may be applied to the result. Evidently the amount of key used per letter of message is statistically reduced by a factor $\frac{R}{R_O}$ and in this case the key source and information source are just matched—an alternative of key conceals an alternative of information. It is easily seen also, by the methods used in the "Information" paper that this is the best that can be done.

Theorem 11: Perfect secrecy (omitting the condition of independence of *a priori* probabilities) for a source with fixed statistics and a rate R of generating information can be achieved with a key source which generates at the rate $(R+\varepsilon)\frac{L_M}{L_K}$ where L_M and L_K are message and key lengths which correspond. A rate less than $R\frac{L_M}{L_k}$ is insufficient.

Perfect secrecy systems have a place in the practical picture—they may be used either where the greatest importance is attached to complete secrecy—e.g. correspondence between the highest levels of command, or in cases where the number of possible messages is small. Thus, to take an extreme example, of only two messages "yes" or "no" were anticipated a perfect system would be in order, with perhaps the transformation table.

K M	A	B
yes	0	1
no	1	0

The disadvantage of perfect systems for large correspondence systems is, of course, the equivalent amount of key that must be sent. In succeeding sections we consider what can be achieved with smaller key size, in particular with finite keys.

19. Equivocation

Let us suppose that a simple substitution cipher has been used on English text and that we intercept a certain amount, N letters, of the enciphered text. For N fairly large, more than say 50 letters, there is nearly always a unique solution to the cipher; i.e. a single good English sequence which transforms into the intercepted material by a simple substitution. With a smaller N, however, the chance of more than one solution is greater; with $N = 15$ there will generally be quite a number of possible fragments of text that would fit, while with $N = 8$ a good fraction (of the order of 1/8) of all reasonable English sequences of that length are possible, since there is seldom more than one repeated letter in the 8. With $N = 1$ any letter is clearly possible and has the same *a posteriori* probability as its *a priori* probability. For one letter the system is perfect.

This happens generally with solvable ciphers. Before any material is intercepted we can imagine the *a priori* probabilities attached to the various possible messages, and also to the various keys. As material is intercepted, the cryptanalyst calculates the *a posteriori* probabilities, and as N increases the probabilities of certain messages increase, and of most, decrease, until finally only one is left, which has a probability nearly one, while the total probability of all others is nearly zero.

This calculation can actually be carried out for very simple systems. Table 1 shows the *a posteriori* probabilities for a Caesar type cipher applied to English text, with the key chosen at random from the 26 possibilities. To enable the use of standard letter digram and trigram frequency tables the test has been started at a random point (by opening a book and putting a pencil down at random on the page). The message selected in this way begins "creases to ..." starting inside the word increases. If the message were to start with the beginning of a sentence a different set of probabilities must be used, corresponding to the frequencies of letters, digrams, etc., at the beginning of sentences.

The Caesar with random key is a pure cipher and the particular key chosen does not affect the *a posteriori* probabilities. To determine these we need merely list the possible decipherments by all keys and calculate their *a priori* probabilities. The *a posteriori* probabilities are these divided by their sum. These possible decipherments are found by the standard process of "running down the alphabet" from the message and are listed at the left. These form the residue class for the message. For one intercepted letter the *a posteriori* probabilities are equal to the *a priori* probabilities for letters and are shown in the column headed $N = 1$. For two intercepted letters the probabilities are those for digram adjusted to sum to unity and these are shown in the column $N = 2$.

Table 1
A Posteriori Probabilities for a Caesar Type Cryptogram

Decipherments	$N=1$	$N=2$	$N=3$	$N=4$	$N=5$
C R E A S	.032	.015	.111	.55	1
D S F B T	.036	.068			
E T G C U	.123	.170			
F U H D V	.023	.023			
G V I E W	.016				
H W J F X	.051	.015			
I X K G Y	.072				
J Y L H Z	.001				
K Z M I A	.005				
L A N J B	.040	.072	.250	.01	
M B O K C	.020	.019	.022	.01	
N C P L D	.072	.066			
O D Q M E	.079	.034			
P E R N F	.023	.085	.438	.43	
Q F S O G	.002				
R G T P H	.060	.013			
S H U Q I	.066	.064	.005		
T I V R J	.096	.272	.166		
U J W S K	.030				
V K X T L	.009				
W L Y U M	.020	.008	.005		
X M Z V N	.002				
Y N A W O	.019	.006			
Z O B X P	.001				
A P C Y Q	.080	.066			
B Q D Z R	.016				
Q (Digits) =	1.248	.999	.602	.340	0

Trigram frequencies have also been tabulated and these are shown in column $N = 3$. For four and five letter sequences probabilities were obtained by multiplication from trigram frequencies since approximately

$$p(ijk\ell) = p(ijk)\ P_{jk}(\ell)$$

Note that at three letters the field has narrowed down to four messages of fairly high probability, the others being small in comparison. At four there are two possibilities and at five just one, the correct decipherment.

In principle this could be carried out with any system but unless the key is very small the number of possibilities is so large that the work involved prohibits the actual calculation.

This set of *a posteriori* probabilities describes how the cryptanalyst's knowledge of the message and key gradually becomes more precise as enciphered material is obtained. This description, however, is much too involved and difficult to obtain for our purposes. What is desired is a simplified description of this approach to uniqueness of the possible solutions.

We will first define a quantity Q called the "equivocation" which measures in an average way the uncertainty of the solution, or how far it is from unicity. Suppose that a certain cryptogram E of N letters has been intercepted. The cryptanalyst can in principle calculate the *a posteriori* probabilities by the use of Bayes' theorem. Thus

$$P_E(M) = P(M)\ P_M(E)/P(E).$$

Similarly the probabilities for various keys, after E has been intercepted are given by

$$P_E(K) = P(K)\ P_K(E)/P(E).$$

The equivocation of the message should measure in some way how spread out these probabilities $P_E(M)$ are; how far they are from being concentrated at one message. In line with our general principles of measuring such dispersion, as in the case of choice, uncertainty, and generating information, we define the equivocation of the message when E has been intercepted to be

$$Q(M) = -\sum_M P_E(M)\ log\ P_E(M)$$

the summation being over all possible messages. Similarly the equivocation in key when E is intercepted is given by

$$Q(K) = -\sum_K P_E(K)\ log\ P_E(K)$$

The same general arguments used to justify our measure of information rate may be used here, to justify the equivocation measure. We note that equivocation zero requires that one message (or key) have probability one, all others zero. Equivocation is measured in the same units as information, i.e. alternatives, digits, etc., according as the logarithmic base is 2, 10, etc. In fact, equivocation is almost identical with information, the difference being one of point of view. In information we stress the notion of how much freedom we have in choosing one element from a set with certain probabilities—in equivocation we emphasize the uncertainty of our knowledge of what was chosen when the probabilities have certain values.

Although any one number can hardly be expected to describe the set $P_E(M)$ perfectly for all purposes, I think the Q defined here does as well as any single statistic can. Some of the theorems which follow indicate the mathematical "naturalness" of this particular measure.

The values of equivocation for the Caesar type cryptogram considered above have been calculated and are given in the last row of Table 1. This is the Q for both key and message, the two being equal in this case.

The definitions given above involve a particular intercepted E, and are the equivocations for that intercepted cryptogram. We wish, however, to find a measure of the equivocation for the system as a whole, which will describe this progress toward uniqueness as N increases in an average sort of way. To do this we form a weighted average of the equivocations for each particular intercepted message E, weighting in accordance with the probabilities of getting the E in question. This may be called the mean equivocation of the system, or where there is no chance of confusion with the narrower equivocation for a particular E, we abbreviate to merely the equivocation. Thus the mean equivocation of message is

$$Q(M) = -\sum_{M,E} P(E)\ P_E(M)\ log\ P_E(M)$$

the summation being over all M and all E. Since

$$P(E)\ P_E(M) = P(E, M)$$

the probability of getting both E and M, we can write this as

$$Q(M) = -\sum P(M, E)\ log\ P_E(M) = -\sum P(M, E)\ log\ P(M)\frac{P_M(E)}{P(E)}$$

Similarly

$$Q(K) = -\sum P(K, E)\ log\ P(K)\frac{P_K(E)}{P(E)}$$

Either of these mean equivocations is a theoretical measure of the secrecy value of the system. We say theoretical, since even when the equivocation is zero, which corresponds to no uncertainty as to the message, it may require a tremendous amount of labor to locate the particular message where the probability is one. It might, for example, be necessary to try each possible K in succession until one was found that transformed the intercepted E into reasonable text in the language. Thus the system would be practically very good, but theoretically solvable. The equivocation may be said to measure the degree of secrecy when the cryptanalyst has unlimited time and energy.

The equivocation is, of course, a function of N, the number of letters intercepted. The functions $Q(K, N)$ and $Q(M, N)$ will be called the equivocation characteristics of the system.

The following data will be helpful in forming a picture of what small values of equivocation represent.

An equivocation of .1 alternative would result, if (1) 9 times in 10 there was no uncertainty as to M, the tenth time two M's were equally probable, or (2) if every time there were two possibilities one with probability .983, the other with probability .017, or (3) if 99 times in 100 there was no uncertainty, the 100th time 1000 equally likely possibilities.

An equivocation of .01 would result (1) if every time there were two possibilities one with probability .999, the other with probability .001, or (2) if 99 times in 100 there is no uncertainty, the other time two equally likely possibilities, or (3) if 999 times in 1000 there is no uncertainty, the other time 6 or 7 equally likely possibilities.

20. Properties of Equivocation

Equivocation may be shown to have a number of interesting properties, most of which fit into our intuitive picture of how such a quantity should behave. We may first show, by an example, the somewhat surprising fact, that after a cryptanalyst has intercepted *certain special* E's, his equivocation as to key or message may be greater than before he intercepted anything. The intercepted material has increased his ignorance of what happened. Suppose there are only two messages M_1 and M_2 with *a priori* probabilities of p and q, and that a simple substitution is used according to the following table, the two keys K_1 and K_2 also having the *a priori* probabilities p and q.

	K_1	K_2
M_1	E_2	E_1
M_2	E_1	E_2

Before the interception, the equivocation of both key and message is $-(p\ log\ p + q\ log\ q)$, which is less than one alternative if $p \neq q$. If $p \gg q$ there is little uncertainty as to which message and key will be chosen, M_1 and K_1. Now suppose he intercepts E_1. The *a posteriori* probabilities of both keys and both messages are easily seen to be 1/2, and hence the equivocation for both key and message is one alternative, greater than before. On the other hand, if E_2 is intercepted, the more probable event, the equivocation for both key and message decreases more than enough to compensate for the other increase, and the mean equivocation of both key and message decreases. This is a general property of all secrecy systems.

Theorem 12: The mean equivocation of key, $Q_K(N)$ is a non-increasing function of N. The mean equivocation of the first A letters of the message is a non-increasing function of the number N which have been intercepted. If N letters have been intercepted, the equivocation of the first N letters of message is less than or equal to that of the key. These may be written

$$Q_K(S) \leq Q_K(N) \qquad S \geq N$$

$$Q_M(M) \leq Q_M(N) \qquad M \geq N$$

$$Q_M(N) \leq Q_K(N)$$

The qualification regarding A letters in the second result of the theorem is so that the equivocation will not be calculated with respect to the amount of message that has been intercepted. If it is, the message equivocation may (and usually does) increase for a time, due merely to the fact that more letters stand for a larger possible range of messages. The results of the theorem are what we might hope from a good measure of equivocation, since we would hardly expect to be worse off on the average after intercepting material than before. The fact that they can be proved gives additional justification to our definition.

The results of this theorem can be proved by a substitution in the property 6 of section 1. Thus to prove the first or second we have for any chance events A and B

$$Q(B) \geq Q_A(B)$$

If we identify B with the key (knowing the first S letters of cryptogram) and A with the remaining $N - S$ letters we obtain the first result. Similarly identifying B with the message gives the second result. The last result follows from

$$Q(M) \leq Q(K) + Q_K(M)$$

and the fact that $Q_K(M) = 0$ since K uniquely determines M.

Theorem 13:

$$\begin{aligned} Q(K) &= |M| - |E| + |K| \\ Q(M) &= |M| - |E| + |H| \end{aligned}$$

where

$$|H| = -\sum_{M,E} P(M,E) \log P_M(E)$$

We have

$$Q(K) = -\sum_{E,K} P(E) P_E(K) \log P_E(K)$$

$$P_E(K) = \frac{P(K) P_K(E)}{P_E}$$

Hence

$$\begin{aligned} Q(K) = -\sum P(K)P_K(E)\log P(K) - \sum P(K)P_K(E)\log P_K(E) \\ + \sum P(K)P_K(E)\log P(E) \end{aligned}$$

Summing the first term on E gives $-\sum P(K)\ log\ P(K) = |K|$.
In the second term $P_K(E)$ is $P(M)$, the unique M that gives E with key K. Summing on K then gives
$-\sum P(M)\ log\ P(M) = |M|$.
The third term is $-\sum P(E)\ log\ P(E) = |E|$.

The second equation in the theorem is proved by the same method.

$$
\begin{aligned}
Q(M) &= -\sum P(E)P_E(M)\log P_E(M) \\
&= -\sum P(M)P_M(E)\log \tfrac{P(M)P_M(E)}{P(E)} \\
&= -\sum P(M)P_M(E)\log P(M) - \sum P(M)P_M(E)\log P_M(E) \\
&\quad +P(M)P_M(E)\log P(E) \\
&= |M| - |E| - \sum P(M)P_M(E) log P_M(E)
\end{aligned}
$$

The last term here may be interpreted as follows. Group together all the different keys that transform a fixed M into the same E, giving the total probability to the group, which will be $P_M(E)$. The last term is the average size of this group space weighted according to the probability $P(M)$ of choosing among the groups leading out of M. In case no group contains more than one element (at any rate no group from an M with $P(M) > 0$) then $|H| = |K|$ and $Q(K) = Q(M)$. This is also clear since there is then a one-to-one correspondence between the keys and messages for any given E.

From the first equation of the theorem we may conclude that $Q(K) = |K|$ in case $|M| = |E|$. This latter occurs in particular if all M's are equally likely and all E's equally likely and there are the same number of each. It is easy to see that this is the case with a language in which every letter is equally likely and independent, and when almost any of the simple ciphers are used.

If we have a product system $S = T\ R$, it is to be expected that the second enciphering process does not decrease the equivocation of message and this is actually true as can be shown by the methods used above. If T and R commute either may be considered as being the first and hence in this case the equivocation with S is not less than the maximum for the two systems R and T. Simple examples show that this does not hold necessarily if R and T do not commute.

Theorem 14: The equivocation in message of a product system $S = T\ R$ is not less than that when only R is used. If $TR = R\ T$ it is not less than the maximum of those for R and T alone.

If we have a product of several systems $R\ S\ T\ U$, we can of course extend this to say that the equivocation of $R\ S\ T\ U$ is not less than that of $S\ T\ U$, which is not less than that for $T\ U$ etc.

There is no similar theorem for the inner product since for example if T and R are inverse processes their inner product is the identity and the resulting equivocation zero.

Suppose we have a system T which can be written as a weighted sum of several systems $R,\ S,\ ..., U$

$$T = p_1 R = p_2 S = \cdots + p_m U \qquad \sum p_i = 1$$

and that systems $R, S, \ldots, U$ have equivocation characteristics $Q_1, Q_2, ..., Q_m$.

Theorem 15: The equivocation Q of a weighted sum of systems is bounded by the inequalities
$\sum p_i Qi \le Q \le \sum p_i Qi - \sum p_i\ log\ p_i$

These are best limits possible. The Q's may refer either to key or to message.

The upper limit is achieved, for example, in strongly ideal systems (to be described later) where the decomposition is into the simple transformations of the system. The lower limit is achieved if all the systems R, $S \ldots$, U go to completely different cryptogram spaces. This theorem is also proved by the general inequalities governing equivocation,

$$Q_A(B) \le Q(B) \le Q(A) + Q_A(B).$$

We identify A with the particular system being used and B with the key or message.

There is a similar theorem for weighted sums of languages.

Theorem 16: Suppose a system can be applied to languages L_1, L_2, $\ldots$, L_m and has equivocation characteristics Q_1, Q_2, $\ldots$, Q_m. When applied to the weighted sum $\sum p_i L_i$, the equivocation Q is bounded by
$\sum p_i Qi \le Q \le \sum p_i Qi - \sum p_i\ log\ p_i$

These limits are the best possible and the equivocations in question can be either for key or message.

The proof here is essentially the same as for the preceding case.

An important consequence of the result

$$Q(K) = |K| + |M| - |E|$$

is the following.

Theorem 17: In any closed system, or any system where the total number of possible cryptograms is equal to the number of possible messages of N letters $Q(K) \geq |K| - (|M_0| - |M|) = |K| - D_N$ where $M_0 = log\ H$ with H the number of possible messages of N letters. D_N is the total redundancy for N letters.

This is true since $|M_0| \geq E$, the equality holding only if all cryptograms are equally likely. The theorem shows that in a closed system the key is determined only by the redundancy of the language—the equivocation can decrease only as the redundancy comes into action and at no greater rate.

Suppose we have a pure system and let the different residue classes of messages be C_1, C_2, C_3, ..., C_r. The corresponding set of residue classes of cryptograms is C'_1, C'_2, ..., C'_r. The probability of each E in C_i is the same:

$$P(E) = \frac{P(C_k)}{\varphi_i} \quad E \in C_i$$

where φ_i is the number of different messages in C_i. Thus we have

$$\begin{aligned} |E| &= -\textstyle\sum_i \varphi_i \frac{P(C_i)}{\varphi_i} \log \frac{P(C_i)}{\varphi_i} \\ &= -\textstyle\sum P(C_i) \log \frac{P(C_i)}{\varphi_i} \end{aligned}$$

Substituting in our equation for Q we obtain:

Theorem 18: For a pure cipher

$$Q = |K| + |M| + \sum_i P(C_i) \log \frac{P(c_i)}{\varphi_i}$$

This result can be used to compute Q in many cases of interest.

From the analytic point of view pure ciphers have a simple structure. If a cryptogram is intercepted its residue class gives the complete information obtained by the cryptanalyst. Within the residue class the system is perfect—each message in the class has an *a posteriori* probability equal to its *a priori* probability. For large N, beyond the unicity point, there will usually only be one M in the class of reasonable probability, and the problem is to determine this M.

The theorem on equivocation of pure ciphers can be altered to show this. We have

$$\begin{aligned} \textstyle\sum p(C_i) \log \frac{P(C_i)}{\varphi_i} &= \textstyle\sum P(C_i) \log P(C_i) - \sum P(C_i) \log \frac{k}{\varphi_i} \\ &+ \textstyle\sum P(C_i) \log k \\ &= \textstyle\sum P(C_i) \log P(C_i) + Q_M(K) - |K| \end{aligned}$$

Hence

$$\begin{aligned} Q(K) &= |K| + |M| + \textstyle\sum P(C_i) \log \frac{P(C_i)}{\varphi - i} \\ &= |M| + Q_M(K) + \textstyle\sum P(C_i) \log P(C_i) \end{aligned}$$

and

$$Q(M) = |M| - [-\sum P(C_I) \log P(C_i)]$$

The equivocation of message is the equivocation of message before the cryptogram was intercepted less the information imparted by a specification of its residue class.

21. Key Appearance Characteristic

Suppose the cryptanalyst has N letters of message and N letters of the equivalent cryptogram. Then he can calculate the *a posteriori* probabilities of the various keys on the basis of this information, and if N is small there will remain a certain equivocation of key. For example in simple substitution, knowing 20 letters of message and cryptogram does not disclose the entire key, since only about 12 letters of the 26 will be represented. Thus there is a residual equivocation of log (26-12)!, if exactly 12 letters appear. We define the mean residual key equivocation as

$$Q_M(K) = \sum_{E,M,K} P(E,M) P_{E,M}(K) \log P_{E,M}(K)$$

when $P(E,M)$ is the *a priori* probability of having message M and cryptogram E, and $P_{E,M}(K)$ is the conditional probability of K with E and M given.

This may be written by obvious arguments (assuming all keys equally likely)

$$Q_M(K) = \sum_{M,K} \log \lambda(M,K)$$

where $\lambda(M,K)$ is the number of different keys from M in parallel with K, that is which go to the same E as K.

For simple substitution let $P\lambda$ be the probability that a received cryptogram of N letters has λ *different* letters appearing in it. Then

$$Q_M(K) = \sum p_\lambda \log(26-\lambda)!$$

Approximately

$$Q_M(K) = \sum P_\lambda (26-\lambda) \left[\log \frac{(26-\lambda)}{e} + \log \sqrt{2\pi(26-\lambda)}\right]$$

The bracketed terms vary slowly with λ and if $P\lambda$ is fairly well concentrated, we may take the bracket out replacing λ by its mean value λ_1. This gives, after recombination

$$Q_M(K) \doteq \log(26 - \lambda_1)!$$

This residual key equivocation is shown for simple substitution on English in Fig. 12. It measures how much of the key has not been used in enciphering N letters of text on the average.

Theorem 19: $Q(K) = Q(M) + Q_M(K)$

That is, the total key equivocation (when we don't know the message) is the sum of the message equivocation and the residual key equivocation, i.e.., the equivocation there would be in the key if we did know the message. This follows from the fact that the key uniquely determines the message and properties 4 and 5 in Section 1.

22. Equivocation for Simple Substitution on an Independent Letter Language

We will now calculate the mean equivocation in key or message when simple substitution is applied to a two letter language, probabilities p and q for 0 and 1, with successive letters independent. We have

$$Q_M = Q_K = -\sum P_E \; P_E(K) \; log \; P_E(K)$$

The probability that E contains exactly s 0's in a particular permutation is

$$P_{E_s} = \frac{1}{2}(p^s q^{N-s} + q^s p^{N-s})$$

and the *a posteriori* probabilities of the identity and inverting substitutions are respectively

$$P_E(0) = \frac{p^s q^{N-s}}{(p^s q^{N-s} + q^s p^{N-s})} \qquad P_E(1) = \frac{p^{N-s} q^s}{(p^s q^{N-s} + q^s p^{N-s})}$$

There are $\binom{N}{s}$ terms for each s and hence

$$Q(N) = -\sum_s \binom{N}{s} p^s q^{N-s} \log \frac{p^s q^{N-s}}{(p^s q^{N-s} + q^s p^{N-s})}$$

This may be written

$$\begin{aligned} Q(N) &= -\sum \tbinom{N}{s} p^s q^{N-s}[s\log p + (N-s)\log q] \\ &\quad - \log(p^s q^{N-s} + q^s p^{N-s}) \\ &= [p\log p + q\log q] + \sum \left(\tfrac{N}{s}\right) \log(p^s q^{N-s} + q^s p^{N-s}) \\ &= NR + \tfrac{1}{2}\sum \tbinom{N}{s}(p^s q^{N-s} + q^s p^{N-s})\log(p^s q^{N-s} + q^s p^{N-s}) \end{aligned}$$

For $p = 1/3$, $q = 2/3$, and for $p = 1/8$, $q = 7/8$, Q has been calculated and is shown in Fig. 13.

Now assume the language contains r different letters chosen independently and with probabilities p_1, p_2, ... p_r. By approximately the same argument we have

$$Q(N) = -\sum_{(s_1 \ldots s_r)} p_1^{s_1} p_2^{s_2} \ldots p_r^{s_r} \log \frac{p_1^{s_1} p_2^{s_2} \ldots p_r^{s_r}}{\sum_p p_1^{s_1} \ldots p_\eta^{s_r}}$$

where $\sum s_i = N$ and $\sum_p$ is over all permutations of $1, 2, \ldots, n$ for $\alpha, \ldots, \eta$.

Hence, by obvious transformation

$$Q(N) = NR + \frac{1}{r!}\sum \binom{N}{s_1 \ldots s_r} \sum_p p_\alpha^{s_1} \ldots p_\eta^{s_r} \log \sum_p p_\alpha^{s_1} \ldots p_\eta^{s_r}$$

where $R = -\sum p_i \log p_i$. In particular

$$\begin{aligned} Q(0) &= \tfrac{1}{r!} r! \log r! = |K| \\ Q(1) &= R + \tfrac{1}{r!} r(r-1)! \log(r-1)! \\ &= R + \log(r-1)! \end{aligned}$$

This checks the evident answer for $Q(1)$—the first symbol has equivocation R and the parts of the key not used add $log\ (r-1)!$

23. The Equivocation Characteristic for a "Random" Closed Cipher.

In the preceding section we have calculated the equivocation characteristic for a simple substitution applied to an independent letter language. This is about the simplest type of cipher and the simplest language structure possible, yet already the formulas are so involved as to be nearly useless. What are we to do with cases of practical interest, say the involved transformations of a fractional transposition system applied to English with its extremely complex statistical structure? This complexity itself suggests the method of approach. Sufficiently complicated problems can frequently be solved statistically. In order to do this we define the notion of a "random" cipher.

We suppose that the possible messages of length N can be divided into two groups, one group of high and fairly uniform probability, while the total probability in the second group is small. This is usually possible in information theory if the messages have any reasonable length. Let the total number of messages be

$$H = 2^{R_0 N}$$

where R is the maximum rate and N the number of letters. The high probability group will contain about

$$S = 2^{RN}$$

where R is the statistical rate.

The deciphering operation defines a function $M = g(K, E)$ which can be thought of as a series of lines, k for each E going back to various M's. By a random cipher we will mean one in which all keys are equally likely and the k lines from any E go back to random M's. The equivocation in key is given by

$$Q(K) = \sum P(E)\ P_E(K)\ \log\ P_E(K)$$

The probability of exactly m lines going back to the high probability group is

$$\binom{k}{m}\left(\frac{S}{H}\right)^m\left(1-\frac{S}{H}\right)^{k-m}$$

If a cryptogram with m lines going to high probability messages is intercepted, the equivocation is *log m.* The probability of intercepting such a cryptogram is easily seen to be $\frac{mH}{Sk}$.

Hence the mean equivocation is

$$Q=\frac{H}{Sk}\sum_{m=1}^{k}\binom{k}{m}\frac{S}{H}^m\left(1-\frac{S}{H}\right)^{k-m}m\log m$$

We wish to find an approximation to this for large K. If the expected value of m, namely $\overline{m}=\frac{S}{H}\ k\ is \gg 1$, the variation of *log m* over the range where the binomial distribution assumes large values will be small and we can replace *log m* by *log* $\overline{m}$. This then comes out of the summation leaving the expected m. Hence in this condition

$$\begin{aligned} Q &= \log\tfrac{S}{H}k \\ &= \log S-\log H+\log k \\ &= |K|-|M|+|M_0| \\ &= |K|-ND \end{aligned}$$

If $\overline{m}$ is small compared to the large k, the binomial distribution can be approximated by a Poisson distribution*.[*]

$$\binom{k}{m}p^m q^{k-m}=\frac{e^{-\lambda}\lambda^m}{m!}\lambda=\frac{S}{H}k$$

Hence

$$\begin{aligned} Q &= \tfrac{1}{\lambda}e^{-\lambda}\textstyle\sum_2^\infty \frac{\lambda^m}{m!}\log m \\ &= e^{-\lambda}\textstyle\sum_1^\infty \frac{\lambda^m}{m!}\log(m+1) \end{aligned}$$

[*] Fry, Probability and Its Engineering Uses, p.214.

When we write $(m+1)$ for m. This may be used in the regions where λ is near unity. For $\lambda \ll 1$ the only important term in the series is $m = 1$; omitting the others

$$\begin{aligned} Q &= e^{-\lambda}\lambda \log 2 \\ &= \lambda \log 2 \\ &= 2^{|K|} 2^{-ND} \log 2 \end{aligned}$$

Thus $Q(K)$ starts off at $|K|$, and decreases linearly with slope $-D$ out to the neighborhood of $N = |K|/D$. After a short transition region, Q follows an exponential with "half life" distance $1/D$ if D is in alternatives per letter. If D is in digits per letter $1/D$ is the distance for a decrease by a factor of 10. The behavior is shown in Fig. 14 with the approximating curves.

By a similar argument given in the appendix, the equivocation of message can be calculated. It is

$$\begin{array}{lll} Q(M) = |M_0| = R_0 N & for & R_0 N \ll Q(K) = |K| - DN \\ Q(M) = Q(K) & & R_0 N \gg Q(K) \\ Q(M) - Q(K) - \varphi(N) & & R_0(N) - Q(K) \end{array}$$

where $\varphi(N)$ is the function of Fig. 14, with N scale reduced by a factor of $\frac{D}{R_0}$. $Q(M)$ rises linearly with slope R_0 until this line intersects the $Q(K)$ line. After a rounded transition it follows $Q(K)$ down.

Most ciphers have an equivocation characteristic of this general type, approaching zero rather sharply. We will call the number of letters required for near unicity of solution the unicity distance.

24. Application to Standard Ciphers

The characteristic derived for the random cipher may be expected to apply approximately in many cases, providing some precautions are taken and certain corrections are made. The main points to be observed are the following:

1. We assumed in deriving the random characteristic that the possible decipherments of a cryptogram are a random selection from the possible messages. This is not true in actual cases, but becomes more nearly true as the complexity of the operations used in the enciphering process and the complexity of the language structure increase. The more complicated the type of cipher, the more it

should follow the random characteristic. In the case of a transposition cipher it is clear that letter frequencies are preserved. This means that the possible decipherments are chosen from a more limited group—not the entire message space—and the formula should be changed. In place of R_0 one uses R_1 the rate for independent letters but with the regular frequencies. This changes the re-redundancy from
$D = R_0 - R \doteq .707$ digits/letter
to
$D = R1 - R \doteq .538$ digits/letter
and the equivocation reduces more slowly. In some other cases a definite tendency toward returning the decipherments to high probability messages can be seen. If there is no clear tendency of this sort, and the system is fairly complicated, and the language a natural one (with its very complex statistical structure)—then it is reasonable to make the random cipher assumption.

2. In many cases the key does not all appear as soon as it might. For example in simple substitution one must wait for a long time to find all letters of the alphabet represented in the message and thus deduce the complete key. The message becomes unique long before this point. Obviously our random assumption falls down in such a case, since all the different keys which differ only in the letters not yet appearing lead back to the same message, and are not randomly distributed. This error is easily corrected by the use of the key appearance characteristic. One uses at a particular N, the amount of key that may be expected at that point in the formula for Q.

3. There are certain "end effects" due to the definite starting of the message which produce a discrepancy from the random characteristics. If we take a random starting point in English text the first letter (when we do not observe the preceding letters) has a possibility of being any letter with

the ordinary letter probabilities. The next letter is more completely specified since we then have digram frequencies. This decrease in choice value continues for some time. The effect of this on the curve is that the straight line part is displaced, and approached by a curve depending on how much the statistical structure of the languag is spread out over adjacent letters. As a first approximation the curve can he corrected by shifting the line over to the half redundancy point—i.e., the number of letters where the language redundancy is half its final value.

If account is taken of these three effects, reasonable estimates of the equivocation characteristic and unicity point can be made. The calculation can be done graphically as indicated in Figs. 15 and 16. One draws the key appearance characteristic $|K| - Q_M(K)$ and the total redundancy curve $|M_0| - |M|$ (which is usually sufficiently well represented by the line NR). The difference between these out to the neighborhood of their intersection is $Q(M)$. For the simple substitution the characteristic is shown in Fig. 17. In so far as experimental checks could be carried out they fit this curve very well. For example, the unicity point, at about 27 letters, can be shown experimentally to lie between the limits 22 and 30. With 30 letters one nearly always has a unique solution to a cryptogram of this type and with 22 it is usually easy to find a number of them.

With transposition of period d, the unicity point occurs as about 1.5 *d log d/c.* This also checks fairly well experimentally. Note that in this case Q is defined only for integral multiples of d.

With the Vigenère the unicity point will occur at about $2d + 2$ letters, and this too is about right. The Vigenère characteristic with the same key size as simple substitution will be approximately as shown in Fig. 18. The Vigenère, Playfair and Fractional cases are more likely to follow the theoretical formulas for random ciphers than simple substitution and transposition. The reason for this is that they are more complex and give better mixing characteristics to the messages on which they operate.

The mixed alphabet Vigenère (each of d alphabets mixed independently and used sequentially) has a key size,

$$|K| = d \; log \; 26 = 26.3 \; d$$

and its unicity point should be at about 53 $d + 2$ letters.

These conclusions can also be put to a rough experimental test with the Caesar type cipher. In the particular cryptogram analyzed in Table I, section 19, the function $Q(N)$ has been calculated and is given below, together with the values for a random cipher

N	0	1	2	3	4	5
Q (observed)	1.41	1.25	1.00	.60	.34	0
Q (calculated)	1.41	1.25	.98	.54	.15	.03

The agreement is seen to be quite good, especially when we remember that the observed Q should actually be the average of many different cryptograms, and that D for the larger values of N is only roughly estimated.

It appears then that the random cipher analysis can be used to estimate equivocation characteristics and the unicity distance for the ordinary types of ciphers.

25. Solving Systems Using Only N-Gram Structure

The preceding analysis can also be applied to cases where the cryptanalyst is assumed to know or use only a limited knowledge of the structure of the language. If no data about the language other than the digram frequencies is used in solving cryptograms the equivocation curves may be computed, using for the redundancy curve that obtained from D_2 alone. This curve lies below the curve for all redundancy and the unicity point will therefore be moved to a larger N. Fig. 19 shows the Q curves for simple substitution on normal English when the cryptanalyst uses only digram structures.

26. Validity of a Cryptogram Solution

The equivocation formulas are relevant to questions which sometimes arise in cryptographic work regarding the validity of an alleged solution to a cryptogram. In the history of cryptography one finds many cryptograms, or possible cryptograms, where clever analysts have found a “solution”. It involved, however, such a complex process, or the material was so scanty; that the question arose as to whether the cryptanalyst had “read a solution” into the cryptogram. See for example the Bacon-Shakespeare ciphers and the “Roger Bacon” manuscript.*

In general we may say that if a proposed system and key solves a system for a length of material considerably greater than the unicity distance the solution is trustworthy. If the material is of the same order or shorter than the unicity distance the solution is highly suspicious.

This effect of redundancy in gradually producing a unique solution to a cipher can be thought of in another way which is helpful. The redundancy is essentially a series of conditions on the letters of the message, which insure that it be statistically reasonable. These consistency conditions produce corresponding consistency conditions in the cryptogram. The key gives a certain amount of freedom to the cryptogram, but as more and more letters are intercepted, the consistency conditions use up the freedom allowed by the key. Eventually there is only one message and key which satisfy all the conditions and we have a unique solution. In the random cipher the consistency conditions are in a sense “orthogonal” to the “grain of the key” and have their full effect in eliminating messages and keys as rapidly as possible. This is the usual case. However, by proper design it is possible to “line up” the redundancy of the language with the “grain of the key” in such a way that the consistency conditions are automatically satisfied and Q does not approach zero. These “ideal” systems are of such a nature that the transformations T_i all induce the same probabilities in the E space. Ideal characteristics are shown in Fig. 20.

27. Ideal Secrecy Systems

We have seen that perfect secrecy requires an infinite amount of key. With a finite key size, the equivocation of key and message generally approach zero, but not necessarily so. In fact it is possible for $Q(K)$ to remain constant at its initial value $|K|$. Then, no matter how much material is intercepted, there is not a unique solution but many of comparable probability. We will define an “ideal” system as one in which $Q(K)$ and $Q(M)$ do not approach zero as $\rightarrow \infty$. A “strongly ideal” system is one in which $Q(K)$ remains constant at $|K|$.

* See Fletcher Pratt, “Secret and Urgent”

An example is a simple substitution on an artificial language in which all letter probabilities are the same and each letter independently chosen. It is clear that $Q(K) = |K|$ and $Q(M)$ rises linearly along a line of slope R_0 until it strikes the line $Q(K)$, after which it remains constant at this value.

With natural languages it is in general possible to approximate the ideal characteristic—the unicity point can be made to occur for as large N as is desired. The complexity of the system needed usually goes up rapidly as we attempt to do this, however. It is not always possible to actually attain the ideal characteristic with any system of finite complexity.

To approximate the ideal equivocation, one may first operate on the message with a transducer which reduces to the normal form—i.e., with all redundancies removed. After this almost any simple ciphering system—substitution, transposition, Vigenère, etc., is satisfactory. The more elaborate the transducer and the nearer the output is to normal form, the more closely will the secrecy system approximate the ideal characteristic. Theorem 20: A necessary and sufficient condition that T be strongly ideal is that for any two keys $T_i^{-1}T_j$ is a measure preserving transformation of Ω_M into itself.

This is true since the *a posteriori* probability of each key is equal to its *a priori* probability if and only if this condition is satisfied.

28. Examples of Ideal Secrecy Systems

Suppose our language consists of a sequence of letters all chosen independently and with equal probabilities. Then the redundancy is zero, $|M_0| = |M|$, and from Theorem 15, $Q(K) = |K|$. We obtain the result

Theorem 21: If all letters are equally likely and independent any closed cipher is strongly ideal.

The equivocation of message will rise along the key appearance characteristic $|K| - Q_M(K)$ which will usually approach $|K|$, although in some cases it does not. In the cases of N-gram substitution, transposition, Vigenère and variations, fractional, etc., we have strongly ideal systems for this simple language with $Q(M) \to |K|$ as $N \to \infty$.

If the letters are independent but are not all equally probable, the transposition cipher characteristics remain essentially the same. The asymptotic equivocations of both key and message are clearly $|K|$. In the substitution cipher they will be less. If all the letter probabilities are different, then the asymptotic equivocations of both key and message are zero. The letters can all eventually be determined by frequency count (apart from certain exceptional sequences of zero measure). Suppose now that there are 7 letters with probabilities,

$$P_1 = P_2 < P_3 < P_4 = P_5 = P_6 < P_7$$

In this case we cannot separate p_1 from p_2 or p_4 p_5 and p_6 from each other, but the different unequal probability groups can be eventually separated.

If all substitutions are *a priori* equally likely, there will be an asymptotic uncertainty among

$$2! \times 3!$$

equally likely (*a posteriori*) keys. Hence, the asymptotic Q will be

$$Q_\infty(M) = Q_\infty(K) = \log\ 2!\,3!$$

In general it is clear that the asymptotic equivocation with a substitution where the different substitutions are equally likely is

$$Q_\infty(M) = Q_\infty(K) = \log\ H$$

where H is the order of the group of substitutions on the letter probabilities $p_1 \cdots p_s$ which leave this set invariant.

More generally we can consider an arbitrary pure system T and a pure language L. Suppose that T operates only "locally" on the letters of M in the sense that the nth letter of cryptogram depends only on n and a certain finite number of the letters of M in the neighborhood of the nth one:

$$e_n = f(K, n, m_n, m_{n=1}, \cdots m_{n=p})$$

Then we can show that there is a certain subgroup of the transformations $T_i^{-1}T_j$ which are probability preserving in the language L. In the limiting cases these would consist of the identity or of the whole group $T_i^{-1}T_j$.

Theorem 22: Under these conditions the asymptotic equivocation of key is the logarithm of the order of this subgroup of measure preserving transformations.

An ideal secrecy system suffers from a number of disadvantages.

1. The system must be closely matched to the language. This requires an extensive study of the structure of the language by the designer. Also a change in statistical structure or a selection from the set of possible messages as in the case of probable words (words expected in this particular cryptogram) renders the system vulnerable to analysis.

2. The structure of natural languages is extremely complicated, and this reflects in a complexity of the transformations required to reduce them to the normal form. Thus any machine to perform this operation must necessarily be quite involved, at least in the direction of information storage, since a "dictionary" of magnitude greater than that of an ordinary dictionary is to be expected.

3. In general, reduction of a natural language to a normal form introduces a bad propagation of error characteristic. Error in transmission of a single letter produces a region of changes near it of size comparable to the length of statistical effects in the original language.

29. Multiple Substitute Ideal Systems

There is another way of obtaining ideal or nearly ideal characteristics using multi-valued secrecy systems. Suppose our language contains only three letters with probabilities 1/8, 3/8, and 4/8, and that successive letters in a message are chosen independently.

Let there be 1 substitute for the first letter, 3 for the second and 4 for the third, and choose at random among the possible substitutes for a letter. It is clear that this system is ideal. If the different probabilities are incommeasurable, we cannot exactly achieve the ideal behavior, but can approximate it, by using enough substitutes, as closely as desired.

If the language is more complex, with transition probabilities, this general method can still be used, but it becomes more involved. Suppose the choice of a letter depends only on the two preceding letters, not on any more remote part of the message. The transition probabilities $p_{i_j}(k)$ completely describe the statistical structure of the language. We supply substitutes for k when it follows i, j in proportion to $p_{i_j}(k)$. Of all our m substitutes $mp_{i_j}(k)$ represent k after the pair i, j. As before one chooses from the possible substitutes for a letter at random. The cryptogram will then be a random sequence of the m substitute letters.

As an example, suppose the $p_i(j)$ are the only statistics of the language and the values are given by

i	j	1	2	3
1		.1	.3	.6
2		.2	.5	.3
3		.9	.1	0

With 10 substitutes 0, 1, 2, . . . , 9 we construct a substitute table assigning substitutes (chosen randomly) in proportion to the frequencies. The following is a typical key.

i	j	1	2	3
1		7	$0, 5, 6$	$1, 2, 3, 4, 8, 9$
2		$3, 9$	$1, 2, 5, 6, 7$	$0, 4, 8$
3		$0, 1, 2, 3, 5, 6, 7, 8, 9$	4	

If a 3 follows a 2 in the message we substitute one of 0, 4, 8 for it, the choice being random. A second table must be supplied for the first letter of the message, corresponding to the unconditional probabilities of the three letters.

Although of theoretical interest it is doubtful whether such systems would be of much use practically because of their complexity and message expansion in ordinary cases. However, the first approximation to such systems, matching letter frequencies, has been used in ciphers and is standard practice in codes (where one matches word frequencies).

30. Equivocation Rate

We now return briefly to cases where the key is not finite, but is supplied constantly, as in the Vernam system and the running key cipher. In such cases we may define equivocation "rates". One considers the equivocation $Q(N)$ of the message when N letters have been intercepted. The equivocation rate for the message is defined as the limit (assuming it exists):

$$\lim_{N\to\infty} \frac{Q(N)}{N} = Q'$$

The rate for equivocation of key would be defined similarly, using the equivocation in the part of the key that has been used only, but of course these two are the same. There are results for these parameter analogous to those obtained with finite key cases. Let R' be the mean rate of using key.

Theorem 23:

$$Q' \leq R'$$

In case the equality holds we have the analogue of ideal systems where the complete information of the key goes into equivocation. If $R' > R$ the rate of the message source, we can obtain perfect secrecy—in fact we may define perfect secrecy as the case in which $Q' = R$.

In the random case we have the analogous result

$$Q' = R' - D$$

31. Further Remarks on Equivocation and Redundancy

We have taken the redundancy of "normal English" to be about .7 digits per letter or 50% of R_0. This is on the assumption that word divisions were omitted. It is an approximate figure based on statistical structure of the order of lengths of perhaps 8 letters, and assumes the text to be of an ordinary type, such as newspaper writing, literary work, etc. Various methods of calculating redundancy have been devised and will be described in the memorandum on information mentioned in the introduction. We may note here two methods of roughly estimating this number which are of cryptographic interest.

A running key cipher is a Vernam type system where in place of a random sequence of letters the key is a meaningful text. Now it is known that running key ciphers can usually be solved uniquely. This shows that English can be reduced by a factor of two to one and implies a redundancy of at least 50%. This figure cannot be reduced very much, however, for a number of reasons, unless long range "meaning" structure of English is considered.

The running key cipher can be easily improved to lead to ciphering systems which could not be solved without the key. If one uses in place of one English text, about 4 different texts as key, adding them all to the message, a sufficient amount of key has been introduced to produce a high positive equivocation rate. Another method would be to use say every 10th letter of the text as key. The intermediate letters are omitted and cannot be used at any other point of the message. This has the same effect, since the mean rate for these spaced letters must be over .8 R_0.

These methods might be useful for spies or diplomats who could use books or magazines for the key source.

A second way of showing the high redundancy of English is to delete all vowels from a passage. In general it is possible to fill them in again uniquely and recover the original, without knowing it in advance. As the vowels constitute about 40% of the text this puts a limit on the redundancy. Actually there is considerable redundancy left, the various letter and digram frequencies being far from uniform.

This suggests a simple way of greatly improving almost any simple ciphering system. First delete all vowels, or as much of the message as possible without running the risk of multiple solutions, and then encipher the residue. Since this reduces the redundancy by a factor of perhaps 3 or 4 to 1, the unicity point will be moved out by this factor. This is one way of approaching ideal systems—using the decipherer's knowledge of English as part of the deciphering system.

Two extremes of redundancy in English prose are represented by Basic English and Joyce's "Finnegans Wake". The basic English vocabulary consists of only 850 words, and a rough estimate puts the redundancy at about 70%. A cipher applied to this sort of text would rapidly approach unicity. Joyce, on the other hand, would be relatively easy to encipher. The small redundancy is disclosed by the difficulty in filling in correctly even a single missing letter from "Finnegans Wake". What the numerical value is, would be difficult to determine; it varies widely throughout the book.

The mathematical extremes of redundancy, 0 and 100%, can be constructed in artificial languages. In the first we have e.g., a single possible message. $Q(M) = 0$ identically and $Q(K)$ in the random cipher case declines as rapidly as possible i.e., as rapidly as one sends information on the system. In the other extreme all letter sequences are equally likely, and any closed ciphering system is ideal.

We may refer here to a memorandum by Nyquist (Enciphering—Effect of Redundancy in Language, May 30, 1944) in which some questions of the type we are considering here are discussed.

32. Distribution of Equivocation

A more complete description of a secrecy system applied to a language than is afforded by the equivocation characteristics can be found by giving the distribution of equivocation. For N intercepted letters we consider the fraction of cryptograms for which Q (for these particular E's, not the mean Q) lies between certain limits. This gives a density distribution function

$$P(Q, N)\ dQ$$

for the probability that for N letters Q lies between the limits Q and $Q + dQ$. The mean equivocation we have previously studied is the mean of this distribution

$$\int P(Q, N)\ Q\ dQ.$$

The function $P(Q, N)$ can be thought of as plotted along a third dimension, normal to the paper, on the Q, N plane. If the language is pure, with a small influence range (compared to $\frac{K}{D}$) and the cipher is pure the function

$P(Q, N)$ will usually be a ridge in this plane whose highest point follows approximately the mean Q, at least until near the unicity point. In this case, or when the conditions are nearly verified, the mean Q curve gives a reasonably complete picture of the system.

On the other hand, if the language is not pure, but made up of a set of pure components

$$L = \sum p_i \; L_i$$

having different equivocation curves with the system, say $Q_1, Q_2 \ldots Q$ then the total Q distribution will usually be made up of a series of ridges. There will be one for each L_i weighted in accordance with its p_i. The mean equivocation characteristic will be a line somewhere in the midst of these ridges and may not give a very complete picture of the situation. This is shown in Fig. 21.

A similar effect occurs if the system is not pure but made up of several systems with different Q curves. There is then a series of ridges in the $P(Q, N)$ plot, and the mean Q strikes an average which may lie between ridges and be a very improbable value of Q for a particular cryptogram. These effects are illustrated in Fig. 22.

The effect of mixing pure languages which are near to one another in statistical structure is to increase the width of the ridge. Near the unicity point this tends to raise the mean equivocation, since equivocation cannot become negative and the spreading is chiefly in the positive direction. We expect therefore, that in this region the calculations based on the random cipher should be somewhat low.

PART III

Practical Secrecy

33. The Work Characteristic

After the unicity point has been passed there will usually be a unique solution to the cryptogram. The problem of isolating this single solution of high probability is the problem of cryptanalysis. In the region before the unicity point we may say that the problem of cryptanalysis is that of isolating all the possible solutions of high probability (compared to the remainder) and determining their various probabilities.

Although it is always possible in principle to determine these solutions (by trial of each possible key for example) different enciphering systems show a wide variation in the amount of work required. The average amount of work to determine the key for a cryptogram of N letters $W(N)$ measured say in man hours may be called the work characteristic of the system. This average is taken over all messages and all keys with their appropriate probabilities.

For a simple substitution on English the work and equivocation characteristics would be somewhat as shown in Fig. 23. The dotted portion of the curve is where there are numerous possible solutions and these must all be determined. In the solid portion after the unicity point only one solution exists in general, but if only the minimum necessary data are given a great deal of work must be done to isolate it. As more material is used the work rapidly decreases toward some asymptotic value—where the additional data no longer reduces the labor.

This is the work characteristic for the key. It is clear that after the unicity point this function can never increase. There is also a work characteristic for the message; the average amount of work to determine the message (or all reasonable messages). This will, in ordinary cases, be below or at any rate not far above the work characteristic for the key, out to fairly large N, since generally if the key is determined it is easy to find M by the deciphering transformation. For very large N, however, this function will increase due merely to the labor of deciphering the large amount of intercepted material.

Essentially the behavior shown in Fig. 23. can be expected with any type of secrecy system where the equivocation approaches zero. The scale of man hours required, however, will differ greatly with different types of ciphers, even when the Q curves are about the same. A Vigenère or compound Vigenère, for example, with the same key size would have a much better (i.e., much higher) work characteristic. A good practical secrecy system is one in which the $W(N)$ curve remains sufficiently high out to the number of letters one expects to transmit with the key, to prevent the enemy from actually carrying out the solution, or to delay it to such an extent that the information is obsolete.

We will consider in the following sections ways of keeping the function $W(N)$ large, even though Q may be practically zero. This is essentially a "max min" type of problem as is always the case when we have a battle of wits.* In designing a good cipher we must maximize the minimum amount of work the enemy must do to break it. It is not enough merely to be sure none of the standard methods of cryptanalysis work—we must show that no method whatever will break the system easily. This, in fact, has been the weakness of many systems—they were designed to resist all the known methods of solution but had a structure leading a to a new method which applied to them. In the history of cryptography there have been many ciphers which were at first thought unbreakable but later disclosed weaknesses of their own.

The problem of good cipher design is essentially one of finding difficult problems, subject to certain other conditions. This is a rather unusual job for the mathematician, who ordinarily is seeking the simple and easily soluble problems in a field.

How can we ever be sure that a system which is not ideal and therefore *has* a unique solution for sufficiently large N will require a large amount of work to break with every method of analysis? There are two approaches to this problem. (1) We can study the possible

* See von Neumann and Morgenstern, "Theory of Games". The situation between the cipher designer and cryptanalyst can be thought of as a "game" of a very simple structure; a zero-sum two person game with complete information, and just two "moves". The cipher designer chooses a system for his "move". Then the cryptanalyst is informed of this choice and chooses a method of analysis. The "value" of the play is the average work required to break a cryptogram in the system by the method chosen.

methods of solution available to the cryptanalyst and attempt to describe them in sufficiently general terms to cover any methods he might use. We then construct our system to resist this "general" method of solution. (2) We may construct our ciphers in such a way that breaking it is equivalent to (or requires at some point in the process) the solution of some problem known to be laborious. Thus, if we could show that solving a system requires at least as much work as solving a system of simultaneous equations in a large number of unknowns, of a complex type, then we will have a lower bound of sorts for the work characteristic.

The next three sections are aimed at these general problems. It is difficult to define the pertinent ideas involved with sufficient precision to obtain results in the form of mathematical theorems, but it is believed that the conclusions, in the form of general principles, are correct.

34. Generalities on the Solution of Cryptograms

After the unicity distance has been exceeded in intercepted material, any system can be solved in principle by merely trying each possible key until the unique solution is obtained—i.e., a deciphered message which "makes sense" in Ω_M. A simple calculation shows that this method of solution (which we may call *complete trial and error*) is totally impractical except when the key is absurdly small.

Suppose, for example, we have a key of 26! possibilities or about 26.3 digits, the same size as in simple substitution on English. This is, by any significant measure, a small key. It can be written on a small slip of paper, or memorized in a few minutes. It could be registered on 27 switches each having ten positions or on 88 two position switches.

Suppose further, to give the cryptanalyst every possible advantage, that he constructs an electronic device to try keys at the rate of one each microsecond (perhaps automatically selecting from the results by a χ^2 test for statistical significance). He may expect to reach the right key about half way through, and after an elapsed time of about

$$\frac{2 \times 10^{26}}{2 \times 60^2 \times 24 \times 365 \times 10^6} = 3 \times 10^{12}\, years$$

In other words, even with a small key complete trial and error will never be used in solving cryptograms, except in the trivial case where the key is extremely small, e.g., the Caesar with only 26 possibilities, or 1.4 digits. The trial and error which is used so commonly in cryptography is of a different sort, or is augmented by other means. If one had a secrecy system which required complete trial and error it would be extremely safe. Such a system would result, it appears, if the original messages, all say of 1000 letters, were a random selection of 2^{RN} from the set of all 2^{R_0N} sequences of 1000 letters. If any of the simple ciphers were applied to these it seems that little improvement over complete trial and error would be possible.

The methods actually used often involve a great deal of trial and error, but in a different way. First, the trials progress from more probable to less probable hypotheses, and second, each trial disposes of a large group of keys, not a single one. Thus the key space may be divided into say 10 subsets, each containing about the same number of keys. By at most 10 trials one determines which subset is the correct one. This subset is then divided into several secondary subsets and the process repeated. With the same key size ($K = 26!$ $= 2 \times 10^{26}$) we would expect about 26 × 5 or 130 trials as compared to 10^{26} by complete trial and error. The possibility of choosing the most likely of the subsets first for test would improve this result even more. If the divisions were into two compartments (the best way) only 90 trials would he required. Whereas complete trial and error requires trials to the order of the number of keys, this subdividing trial and error requires only trials to the order of the key size in alternatives.

This remains true even when the different keys have different probabilities. The proper procedure then to minimize the expected number of trials is to divide the key space into subsets of equiprobability. When the proper subset is determined, this is again subdivided into equiprobability subsets. If this process can be continued the number of trials expected when each division is into two subsets will be

$$h = \frac{|K|}{\log 2}$$

If each test has S possible results and each of these corresponds to the key being in one of S equiprobability subsets; then

$$h = \frac{|K|}{\log S}$$

trials will be expected. The intuitive significance of these results should be noted. In the two compartment test with equiprobability, each test yields one alternative of information as to the key. If the subsets have very different probabilities as in testing a single key in complete trial and error only a small amount of information is obtained from the test. This with 26! equiprobable keys, a test of one yields only

$$-\left[\frac{26!-1}{26!}\log\frac{26!-1}{26!}+\frac{1}{26!}\log\frac{1}{26!}\right]$$

or about 10^{-25} alternatives of information. Dividing into S equiprobability subsets maximizes the information obtained from each trial at *log* S, and the expected number of trials is the total information to be obtained, that is the key size, divided by this amount.

The question here is similar to various coin weighing problems that have been circulated recently. A typical example is the following: It is known that one coin in 27 is counterfeit, and slightly lighter than the rest. A chemists balance is available and the counterfeit coin is to be isolated by a series of weighings. What is the least number of weighings to do this? The correct answer is 3, obtained by first dividing the coins into three groups of 9 each. Two of these are compared on the balance. The three possible results determine the set of 9 containing the counterfeit. This set is then divided into 3 subsets of 3 each and the process continued. The set of coins corresponds to the set of keys, the counterfeit coin to the correct key, and the weighing procedure to a trial or test.

This method of solution is feasible only if the key space can be divided into a small number of subsets, with a simple method of determining to which subset the correct key belongs. Starting in another way, it is possible to solve for the key bit by bit. One does not need to assume a complete key in order to apply a consistency test and determine if the assumption is justified—an assumption on a part of the key (or as to whether the key is in some large section of the key space) can be tested.

This is one of the greatest weaknesses of most ciphering systems. For example, in simple substitution, an assumption on a single letter can be checked against its frequency, variety of contact, doubles or reversals, etc. In determining a single letter the key space is reduced by 1.4 digits from the original 26. The same effect is seen in all the elementary types of ciphers. In the Vigenère, the assumption of two or three letters of the key is easily checked by deciphering at other points with this fragment and seeing whether clear emerges. The compound Vigenère is much better from this point of view, if we assume a fairly large number of component periods, producing a repetition rate larger than will be intercepted. Here as many key letters are used in enciphering each letter as there are periods—although this is only a fraction of the entire key, at least a fair number of letters must be assumed before a consistency check can be applied.

Our first conclusion then, regarding practical small key cipher design, is that a considerable amount of key should be used in enciphering each small element of the message.

35. Statistical Methods

It is possible to solve many kinds of ciphers by statistical analysis. Consider again simple substitution. The first thing a cryptographer does with an intercepted cryptogram is to make a frequency count. If the cryptogram contains say 200 letters it is safe to assume that few, if any, letters are out of their frequency groups, this being a division into 4 sets of well defined frequency limits. The log of the number of keys within this limitation may be calculated as

$$\log\ 2!\,9!\,9!\,6! = 14.28$$

and the simple frequency count thus reduces the key uncertainty by 12 digits, a tremendous gain.

In general, a statistical attack proceeds as follows. A certain statistic is measured on the intercepted cryptogram E. This statistic is such that for all reasonable M it assumes about the same value, S_K, the value depending only on the particular key K that was used. The value thus obtained serves to limit the possible keys, to those which would give values of S in the neighborhood of that observed. A statistic which does not depend on K or which varies as much with M as with K is net of value in limiting K. Thus in transposition ciphers, the frequency count of letters gives no information about K—every K leaves this statistic the same. Hence one can make no use of a frequency count in breaking transposition ciphers.

More precisely one can ascribe a "*solving power*" to a given statistic S. For each value of S there will be a conditional equivocation of the key $Q_S(K)$, the equivocation when S has its particular value and that is all that is known concerning the key. The weighted mean of these values

$$\sum P(S)\ Q_S(K)$$

gives the mean equivocation of the key when S is known, $P(S)$ being the *a priori* probability of the particular value S. The key size $|K|$ less this mean equivocation measures the "solving power" of S.

In a strongly ideal cipher all statistics of the cryptogram are independent of the particular key used. This is the measure preserving property of $T_j T_k^{-1}$ on the E space or $T_j\ ^{-1} T_k$ on the M space mentioned above.

There are good and poor statistics, just as there were good and poor methods of trial and error. Indeed the trial and error testing of hypothesis *is* a type of statistic, and what was said above regarding the best types of trials holds generally. A good statistic for solving a system must have the following properties:

1. It must be simple to measure.

2. It must depend more on the key than on the message if it is meant to solve for the key. The variation with M should not mask its variation with K.

3. The values of the statistic that can be "resolved" in spite of the "fuzziness" produced by variation in M should divide the key space into a number of subsets of comparable probability, with the statistic specifying the one in which the correct key lies. The statistic should give us sizable information about the key, not a tiny fraction of an alternative.

4. The information it gives must be simple and usable. Thus the subsets in which the statistic locates the key must be of a simple nature in the key space.

Frequency count for simple substitution is an example of a very good statistic.

Two methods (other than recourse to ideal systems) suggest themselves for frustrating a statistical analysis. These we may call the methods of *diffusion* and *confusion*. In the method of diffusion the statistical structure of M which leads to its redundancy is "dissipated" into long range statistics—i.e., into statistical structure involving long combinations of letters in the cryptogram. The effect here is that the enemy must intercept a tremendous amount of material to tie down this structure, since the structure is evident only in blocks of very small individual probability. Furthermore even when he has sufficient material, the analytical work required is much greater since the redundancy has been diffused over a large number of individual statistics. An example of diffusion of statistics is operation on a message $M = m_1, m_2, m_3, \ldots$ with a "smoothing" operation, e.g.

$$y_n = \sum_{i=1}^{s} m_{n+i} \bmod 26$$

adding s successive letters of the message to get a letter y_n. One can show that the redundancy of the y sequence is the same as that of the m sequence, but the structure has been dissipated. Thus the letter frequencies in y will be more nearly equal than in m, the digram frequencies also more nearly equal etc. Indeed any reversible operation which produces one letter out for each letter in and does not have an infinite "memory" has an output with the same redundancy as the input. The statistics can never be eliminated without compression, but they can be spread out.

The method of confusion is to make the relation between the simple statistics of E and the simple description of K a very complex and involved one. In the case of simple substitution, it was easy to describe the limitation of K imposed by the letter frequencies of E. If the connection is very involved and confused the enemy can still evaluate a statistic S_1 say which limits the key to a region of the key space. This limitation, however, is to some complex region R in the space—folded over many times, and he has a difficult time making use of it. A second statistic S_2 limits K still further to R_2, hence it lies in the intersection region $R_1 R_2$, but this does not help much because it is so difficult to determine just what the intersection is.

To be more precise let us suppose the key space has certain "natural coordinates" $k_1, k_2, \ldots, k_p$ which he wishes to determine. He measures a set of statistics $s_1, s_2, \ldots, s_n$ and these are sufficient to determine the k_i. However, in the method of confusion, the equations connecting these sets of variables are involved and complex. We have, say,

$$\begin{aligned} f_1(k_1, k_2, \cdots, k_p) &= s_1 \\ f_2(k_1, k_2, \cdots, k_p) &= s_2 \\ &\vdots \\ f_n(k_1, k_2, \cdots, k_p) &= s_n \end{aligned}$$

and all the f_i involve all the k_i . The cryptographer must solve this system simultaneously—a difficult job. In the simple (not confused) cases the functions involve only a small number of the k_i—or at least some of these do. One first solves the simpler equations, evaluating some of the k_i and substitutes these in the more complicated equations.

The conclusion here is that for a good ciphering system steps should be taken either to diffuse or confuse the redundancy (or both).

36. The Probable Word Method

One of the most powerful tools for breaking ciphers is the use of probable words. The probable words may be words or phrases expected in the particular message due to its source, or they may merely be common words or syllables which occur in any text in the language, such as the, and, tion, that, etc.

In general, the probable word method is used as follows. Assuming a probable word to be at some point in the clear, the key or a part of the key is determined. This is used to decipher other parts of the cryptogram and provide a consistency test. If the other parts come out in clear, the assumption is justified.

There are few of the classical type ciphers that use a small key and can resist long under a probable word analysis. From a consideration of this method we can frame a test of ciphers which might be called the acid test. It applies only to ciphers with a small key (less than say 50 digits), applied to natural languages, and not using the ideal method of gaining secrecy. The acid test is this: How difficult is it to determine the key or a part of the key knowing a sample of message and corresponding cryptogram? Any system in which this is easy cannot be very resistant, for the cryptanalyst can always make use of probable words, combined with trial and error, until a consistent solution is obtained.

The conditions on the size of the key make the amount of trial and error small, and the condition about ideal systems is necessary, since these automatically give consistency checks. The existence of probable words and phrases is implied by the condition of natural languages. Conversely, it seems reasonable that if the key is difficult to obtain, knowing a text and its cryptogram, then the system should be strong.

Note that this requirement by itself is not contradictory to the requirements that enciphering and deciphering be simple processes. Using functional notation we have for enciphering

$$E = f(K, M)$$

and for deciphering

$$M = g(K, E)$$

Both of these may be simple operations on their arguments without the third equation

$$K = h(M, E)$$

being simple.

We may also point out in investigating a new type of ciphering system one of the best methods of attack is to consider how the key could be determined if a sufficient amount of M and E were given.

With a small key, the work required to solve a system, given a large amount of data, may be expected to be not more than a few orders of magnitude greater than the work required to obtain the key from a small amount of data when both M and E are known.

The same principle of confusion can be (and must be) used here to create difficulties for the cryptanalyst. Given $M = m_1\ m_2 \ldots m_s$ and $E = e_1\ e_2 \ldots e_s$ the cryptanalyst can set up equations for the different key elements $k_1\ k_2 \ldots k_r$ (namely the enciphering equations).

$$\begin{gathered} e_1 = f_1(m_1, m_2, \ldots m_s; K_i \ldots, k_r) \\ e_2 = f_2(m_1, m_2, \ldots m_s; K_i \ldots, k_r) \\ \vdots \\ e_s = f_s(m_1, m_2, \ldots m_s; K_i \ldots, k_r) \end{gathered}$$

All is known, we assume, except the k_i. Each of these equations should therefore be complex in the k_i, and involve many of them. Otherwise the enemy can solve the simple ones and then the more complex ones by substitution.

From the point of view of increasing confusion, it is desirable to have the f_i involve several m_i, especially if these are not adjacent and hence less correlated. This introduces the undesirable feature of error propagation, however, for then each e_i will generally affect several m_i in deciphering, and an error will spread to all these.

We conclude that much of the key should be used in an involved manner in obtaining any cryptogram letter from the message to keep the work characteristic high. Further a dependence on several uncorrelated m_i is desirable, if some propagation of error can be tolerated. We are led by all three of the arguments of these sections to consider "mixing transformations".

37. Mixing Transformations

A notion that has proven valuable in certain branches of probability theory is the concept of a "mixing transformation". Suppose we have a probability or measure space Ω, and a measure preserving transformation T of the space into itself, i.e., a transformation such that the measure of a transformed region TR is equal to the measure of the initial region R. The transformation is called mixing if for any function defined over the space and any region R.

$$\lim_{n\longrightarrow\infty}\int_{T^nR} f(P)dP = \int_R dp \int_\Omega f(P)dP.$$

This means that any initial region of the space R under successive applications of T is mixed into the entire space Ω with uniform density. In general T^nR becomes a region consisting of a large number of thin filaments spread throughout the region. As n increases the filaments become finer and their density more nearly constant.

An example of a mixing transformation is shown in Fig. 21. Here measure is identified with Euclidean area. The space is the triangle, and $T^\lambda P$ is the point λ units of distance above point P providing this does not go outside the triangle. When the top of the triangle is reached a point is transferred first to the point directly beneath, and then over to the right an irrational fraction of the base width. If this carries the point beyond the right edge, the extra distance is measured from the left edge. Successive transforms of a square region are shown in Fig. 21. For λ very large the square is turned into a uniform grating of nearly parallel thin strips covering the triangle.

A mixing transformation in this precise sense can occur only in a space with an infinite number of points, for in a finite point space the transformation must be periodic. Speaking loosely, however, we can think of a mixing transformation as one which distributes any reasonably cohesive region in the space fairly uniformly over the entire space. If the first region could be described in simple terms, the second would require very complex ones. In the case of cryptographic interest, the original region is all of a certain simple statistical structure—after the mix the region is distributed and the structure diffused and confused.

Good mixing transformations are often formed by repeated products of two simple non-commutating operations. See for example the mixing of pastry dough discussed by Hopf.* The dough is first rolled out into a thin slab, then folded over, then rolled, and then folded again, etc.

In a good mixing transformation of a space with natural coordinates $X_1, X_2, ..., X_s$ the point X_i is carried by the transformation into a point X_i, with

$$X_i = f_i(X_1, X_2, \ldots X_S) \qquad i = 1, 2, \ldots, S$$

and the functions f_i are complicated, involving all the variables in a "sensitive" way. A small variation of any one, X_3, say, changes all the X_i considerably. If X_3 passes through its range of possible variation the point X_i traces a long winding path around the space.

Various methods of mixing applicable to statistical sequences of the type found in natural languages can be devised. One which looks fairly good is to follow a preliminary transposition by a sequence of alternating substitutions and simple linear operations, adding adjacent letters mod 26 for example. Thus

$$H = LSLSLT$$

where T is a transposition, L is a linear operation, and S is a substitution.

* E. Hopf, On Causality, Statistics and Probability, Journal of Math. and Physics, V 13, pp. 51-102, 1934.

38. Ciphers of the Type $T_k H S_j$

Suppose that H is a good mixing transformation that can be applied to sequences of letters and that T_k and S_j are any two simple families of transformations, i.e., two simple ciphers, which may be the same. For concreteness we may think of them as both simple substitutions.

It appears that the cipher THS will be a very good ciphering system from the standpoint of its work characteristic. In the first place it is clear on reviewing our arguments about statistical methods that no simple statistics will give information about the key—any significant statistics derived from E must be of a highly involved and very sensitive type—the redundancy has been both diffused and confused by the mixing H. Also probable words lead to a complex system of equations involving all parts of the key (when the mix is good), which must be solved simultaneously. The bad features of such a system are propagation of errors and complexity of operations, both of which get worse as the mixing of H gets better.

It is interesting to note that if the cipher T is omitted the remaining system is similar to S and thus no stronger. The enemy merely “unmixes” the cryptogram by application of H^{-1} and then solves. If S is omitted the remaining system is much stronger than T alone if the mix is good, but still not comparable to THS.

The basic principle here of simple ciphers separated by a mixing transformation can of course be extended. For example one could use

$$T_k H_i S_j H_2 R_1$$

with two mixes and three simple ciphers. One can also simplify by using the same ciphers, and even the same keys (inner product) as well as the same mixing transformations. This might well simplify the mechanization of such systems.

The mixing transformation which separates the two (or more) appearances of the key acts as a kind of barrier for the enemy—it is easy to carry a known element over this barrier but an unknown (the key) does not go easily.

By supplying two sets of unknowns, the key for S and the key for T, and separating them by the mixing transformation H we have “tangled” the unknowns together in a way that makes solution very difficult.

Although systems constructed on this principle would be extremely safe they possess one grave disadvantage. If the mix is good then the propagation of errors is bad. A transmission error of one letter will affect several letters on deciphering.

39. The Compound Vigenère

In the compound Vigenère several keys of length $d_1, d_2, \ldots$ d_s are written under the message and added to it modulo 26 to obtain the cryptogram. The result is a Vigenère with key of special ??type, whose repetition is of period d, the least common multiple of $d_1, d_2, \ldots$ d_S. If we have three keys of periods 2, 3, 5 the total period d is 30 and the total key size $(2+3+5) \times 1.41 = 14.1$ digits The situation is then

$$M = m_1\ m_2\ m_3\ m_4\ m_5\ m_6$$
$$K_1 = a_1\ a_2\ a_1\ a_2\ a_1\ a_2$$
$$K_2 = b_1\ b_2\ b_3\ b_1\ b_2\ b_3$$
$$K_3 = c_1\ c_2\ c_3\ c_4\ c_5\ c_1$$
$$E = e_1\ e_2\ e_3\ e_4\ e_5\ e_6$$

with

$$e_1 = m_1 + a_1 + b_1 + c_1$$
$$e_2 = m_1 + a_2 + b_1 + c_2$$

etc.

If we assume M and E known then, letting $h_i = e_i - m_i$:

$$\begin{array}{ll} a_1 + b_1 + c_1 = h_1 & a_2 + b_3 + c_1 = h_5 \\ a_2 + b_2 + c_2 = h_2 & a_1 + b_1 + c_2 = h_7 \\ a_1 + b_3 + c_3 = h_3 & a_2 + b_2 + c_3 = h_8 \\ a_2 + b_1 + c_4 = h_4 & a_1 + b_3 + c_4 = h_9 \\ a_1 + b_2 + c_5 = h_5 & a_2 + b_1 + c_5 = h_{10} \end{array}$$

These equations are easily solved for the key, although not as easily as in the simple Vigenère or other simple ciphers. As the number of constituent periods increases the solution becomes more involved and time consuming. In any case we have a system of simultaneous equations each involving S of the total of $B = \sum_i^S \, d_i$ unknowns. The unicity point will occur at about $2B$ letters and if several times this amount of material is intercepted no great difficulty should be encountered in breaking the cipher, providing S is not more than say 6 or 8. With the first 9 primes as periods we have a key size of 100 letters or about 141 digits, the unicity distance is about 200 letters and the key does not repeat for 223,092,870 letters. This system, although much better than such methods as simple substitution, transposition and simple Vigenère with equivalent key size, does not utilize the available key fully in making the cryptanalyst work for the solution. The equations only involve S of the B key unknowns and these in a simple fashion. The equations easily combine and reduce to eliminate unknowns. If a large amount of material is available, compared to the unicity distance, particular sets of equations can be combined to eliminate unknowns very easily. The system possesses the important advantage, however, of not expanding errors. One incorrect letter of cryptogram produces one incorrect letter of deciphered text.

By relatively simple changes this system could be strengthened considerably. If the equations for the key elements (with M and E known) could be made into higher degree equations rather than linear ones the difficulty of solution would increase tremendously. This could easily be done in a mechanical device by successive multiplications (Mod 26) of the key letters according to some prearranged scheme.

40. Incompatibility of the Criteria for Good Systems

The five criteria for good secrecy systems given in section 12 appear to have a certain incompatibility when applied to a natural language with its complicated statistical structure. With artificial languages having a simple statistical structure it is possible to satisfy all requirements simultaneously, by means of the ideal type ciphers. In natural languages it seems that a compromise must be made and the valuations balanced against one another with a view toward the particular application.

If any one of the five criteria is dropped, the other four can be satisfied fairly well, as the following examples show.

1. If we omit the first requirement (amount of secrecy) any simple cipher such as simple substitution will do. In the extreme case of omitting this condition completely, no cipher at all is required and one sends the clear!.
2. If the size of the key is not limited the Vernam system can be used.
3. If complexity of operation is not limited, various extremely complicated types of enciphering process can be used. The modified compound Vigenère described above with many different periods compounded is fairly satisfactory as an example here, although it falls down somewhat on the key size condition. Ideal systems and enciphered codes are also fair examples although not too good from the propagation of error point of view.
4. If we omit the propagation of error condition systems of the type THS would be very good, although somewhat complicated.
5. If we allow large expansion of message, various systems are easily devised where the "correct" message is mixed with many "incorrect" ones (misinformation). The key determines which of these is correct.

A rough argument for the incompatibility of the five conditions may be given as follows.

From condition 5, secrecy systems essentially as studied in this paper must be used; i.e., no great use of nulls, etc. Perfect and ideal systems are excluded by condition 2 and by 3 and 4, respectively. The high secrecy required by 1 must then come from a high work characteristic, not from a high equivocation characteristic. If the key is small, the system simple, and the errors do not propagate, probable word methods will generally solve the system fairly easily, since we then have a fairly simple system of equations for the key.

This reasoning is too vague to be conclusive, but the general idea seems quite reasonable. Perhaps if the various criteria could be given quantitative significance, some sort of an exchange equation could be found involving them and giving the best physically compatible sets of values. The two most difficult to measure numerically are the complexity of operations, and the complexity of statistical structure of the language.

Appendix 1

Deduction of $-\sum p_i \log p_i$

It will be shown that the measure of choice $-\sum p_i \log p_i$ is a logical consequence of three quite reasonable assumptions about the desired properties of such a measure. The three assumptions are:

(1) There exists a function $C(p_1, p_2, \ldots, p_n)$ continuous in the p_i, measuring the amount of "choice" when there are n possibilities with probabilities p_i.

(2) C has the property that if a given choice be broken down into two successive choices the total amount of choice is the weighted sum of the individual choices. For example, suppose the choice is from 4 possibilities A, B, C, D with probabilities .1, .2, .3, .4. This can be broken down into a preliminary choice between the pair A, B and the pair C, D. Pair A, B has a total probability .1 + .2 = .3 and pair C, D probability .3 + .4 = .7. If pair A, B is chosen a second choice between A and B must be made with probabilities $\frac{.1}{.1+.2} = \frac{1}{3}$ and $\frac{.2}{.1+.2} = \frac{2}{3}$. If pair C, D is chosen a second choice between C and D must be made with probabilities $\frac{3}{7}$ and $\frac{4}{7}$. Thus broken down we have a preliminary amount of choice C (.3, .7) and .3 of the time a secondary choice of C $(\frac{1}{3}, \frac{2}{3})$ while .7 of the time the secondary choice is C $(\frac{3}{7}, \frac{4}{7})$. Our condition requires that the total choice $C(.1, .2, .3, .4)$ be the same as the weighted sum of the different choices when decomposed, weighted in accordance with the frequency of occurrence. Thus we require in this case $C(.1, .2, .3, .4) = C(.3, .7) + .3\ C(\frac{1}{3}, \frac{2}{3}) + .7\ C(\frac{3}{7}, \frac{4}{7})$.

(3) If $A(n) = C(\frac{1}{n}, \frac{1}{n}, \ldots \frac{1}{n})$, i.e. the choice when there are n equally likely possibilities, the $A(n)$ is monotonic increasing in n.

Theorem: Under these three assumptions

$$C(p_1, p_2, \ldots p_n) = -K \sum p_i \log p_i$$

where K is a positive constant.

From condition (2) we can decompose a choice from S^m equally likely possibilities into a series of m choices each from S equally likely possibilities and obtain

$$A(S^m) = mA(s)$$

Similarly

$$A(t^n) = nA(t)$$

We can choose n arbitrarily large and find an m to satisfy

$$S^m \leq t^n < S^m + 1$$

Thus, taking logarithms and dividing by n log S,

$$\frac{m}{n} \leq \frac{\log t}{\log S} \leq \frac{m}{n} + \frac{1}{n} \text{ or } \left| \frac{m}{n} - \frac{\log t}{\log S} \right| < \varepsilon$$

where ε is arbitrarily small.
Now from the monotonic property of $A(n)$

$$A(S^m) \leq A(t^n) \leq A(S^{m+l})$$

$$mA(S) \leq nA(t) \leq (m+1)\ A(S)$$

Hence, dividing by $nA(S)$,

$$\frac{m}{n} \leq \frac{A(t)}{A(S)} \leq \frac{m}{n} + \frac{1}{n} \text{ or } \left| \frac{m}{n} - \frac{A(t)}{A(S)} \right| < \varepsilon$$

$$\left| \frac{A(t)}{A(S)} - \frac{\log t}{\log S} \right| \leq \varepsilon \quad A(t) = -K \log t$$

where K must be positive to satisfy (3).
Now suppose we have a choice from n possibilities with commeasurable probabilities $p_i = \frac{n_i}{sum n_i}$ where the n_i are integers. We can break down a choice from $\sum n_i$ possibilities into a choice from n possibilities with probabilities $p_1 \ldots p_n$ and then, if the ith was chosen, a choice from n_i with equal probabilities. Using condition 2 again, we equate the total choice from $\sum n_i$ as computed by two methods

$$K \log \sum n_i = C(p_1, \ldots, p_n) + K \sum\ p_i\ \log\ n_i$$

Hence

$$\begin{aligned} C &= K\left[p_i \log \sum n_i - \sum p_i \log n_i\right] \\ &= -K \sum p_i \log \frac{n_i}{\sum_{n_i}} = -K \sum p_i \log p_i \end{aligned}$$

If the p_i are incommeasurable, they may be approximated by rationals and the same expression must hold by our continuity assumption. Thus the expression holds in general. The choice of coefficient K is a matter of convenience and amounts to the choice of a unit of measure.

Appendix 2

Proof of Theorem 4

Select any message M_1 and group together all cryptograms that can be obtained from M_1 by an enciphering operation T_i. Let this class of cryptograms be C_1. Group with M_1 all M_K that can be obtained from M_1 by $T_i^{-1}T_jM_1$, and call this class C_1. The same C_1' would be obtained if we started with any other M in C_1 since

$$T_ST_j^{-1}T_iM_1 = T_iM_1$$

Similarly the same C_1 would be obtained.

Choosing an M (if any exist) not in C_1 we construct C_2 and C_2' in the same way. Thus we obtain the residue classes with properties (1) and (2). Let M_1 and M_2 be in C_1 and suppose

$$M_2 = T_2\ T_1^{-1}\ M_1$$

If E_1 is in C_1' and can be obtained from M_1 by

$$E_1 = T_\alpha\ \ M_1 = T_\beta\ \ M_1 = \ldots T_\eta M_1$$

then

$$E_1 = T_\alpha\ T_2\ T_1\ M_2 = T_\beta\ T_2^{-1}T_1\ M_2 = \cdots$$
$$= T_\lambda\ M_2 = T_\mu\ M_2 \cdots$$

Thus each M_i in C_1 transforms into E_1 by the same number of keys. Similarly each E_i in C_i is obtained from any M in C_1 by the same number of keys. It follows that this number of keys is a divisor of the total number of keys and hence we have properties (3) and (4).

Appendix 3

Equivocation of Message for Random Cipher

As before let $M_1 \ldots M_S$ be high probability messages and $M_S+1\ldots, M_H$ have zero probability. Let $P(m_1,\ m)$ be the probability of just m_1 lines going from a particular E, say E_1 to a particular high probability M, say M_1, with a total of m lines to all high probability M. Then

$$P(m_1,m)=\binom{k}{m}\binom{m}{m_1}(\frac{1}{H})^{m_1}(\frac{S-1}{H})^{m-m_1}(\frac{1-S}{H})^{k-n}$$

The probability of intercepting an E with m lines to high probability M's is

$$\frac{m}{S_k}$$

The $Q(M)$ expected can be thought of as contributed to by the various M_1 in the high probability group. Thus M_1 contributes

$$-\frac{m_1}{m}\log\frac{m_1}{m}=\frac{m_1}{m}\log\frac{m}{m_1}$$

if there are m_1 lines to M_1 and a total of m to high probability M's. The expected Q is then

$$Q(M)=HS_{m1}\sum_m P(m_1,m)\frac{m}{S_k}\frac{m_1}{m}\log\frac{m}{m_1}$$

The factor H sums over the various E_i and the S sums over the different $M_i(i=1,\ldots,S)$. Hence,

$$Q(M)=\frac{H}{k}\sum P(m_1,m)m_1[\log m-\log m_1]$$

the term

$$\sum P(m_1,m)m_1$$

summed on m_1, gives the expected m_1, when m lines go to high probability M's, i.e., m/s. Hence the first term is

$$\frac{H}{ks}\sum mP(m)\log m=Q(K)$$

by our previous work. The second term is

$$-\frac{H}{k}\sum P(m_1,m)\ m_1\ \log\ m_1$$

If the expected m_1 is $\ll 1$ this term is small since it vanishes for $m_1 = 0$ *or* 1. The expected m_1 is k/H. Thus beyond this point $Q(M)$ approaches closely to$Q(K)$. The point in question is where $|K| = |M_0| = R_0N$
or

$$N = |\frac{K}{R_0}|$$

If the expected $m_1 \gg 1$ the log m_1 can be taken out as log $\overline{m_1} =$ log k/H_1 and we have

$$-\tfrac{H}{k}\log\tfrac{k}{H}\sum P(m_1,m)m_1$$

$$= -\log\tfrac{H}{k} = |M_0| - |K|$$

In this region then

$$Q(M) = |M_0| - |K| + Q(K)$$

but here $Q(K) = |K| - |M_0| + |M|$, and therefore

$$Q(M) = |M| = RN$$

In the transition region $\overline{m_1}$ is about 1 and $\overline{m}$ will in ordinary cases be very large. It is admissable then to replace $P(m_1, m)$ by $P(m_1)$, since this will not depend on m to any extent except for values of m of very small probability. Thus we obtain for this region

$$Q(M) = Q(K) - \frac{H}{k}\sum_{1}^{k} P(m_1)\ m_1\ \log\ m_1$$

The sum has the same form as our expression for $Q(K)$ but with $1/H$ in place of S/H. The calculations for $Q(K)$ can be used, therefore, with only a change of the N scale by a factor of R_0/D.

Appendix 4

Key Appearance in Simple Substitution with Independent Letters

If successive letters are chosen independently and the different letters have probabilities $p_1\ p_2 \ldots p_s$, we can calculate the expected number of different letters when N letters have been intercepted. It is

$$\overline{d}(N) = S - \sum_i (1 - p_i)^N$$

To prove this, imagine all the possible sequences of N letters written down, each with a frequency corresponding to its probability, giving a total of say A sequences. Letter 1 does not appear in $(1 - P_1)^N A$ of these; letter 2 does not appear in $(1 - P_2)^N A$ etc. Therefore, the total number of letters missing from sequences is

$$A \sum (1 - p_i)^N$$

Dividing by A gives us by definition the expected number of missing letters from a random sequence, $\sum (1 - p_i)^N$. The number of different letters expected in a sequence is the total number of letters S minus this, giving the desired result.

If all the p_i are equal this reduces to $S - S(1 - p)^N$, an exponential approach to S. In the general case there are a series of exponentials with different time constants, corresponding to different p_i, which are added to give $\overline{d}(N)$.

With the frequencies of normal English used for the p_i we obtain the curve shown in Fig. 25, along with an experimental curve. The small discrepancy can be attributed to the influences of nearby letters. In English there is less tendency to double letters than there would be if the letters were independent but with the same probabilities. For English the probability of a doubled digram is

$$\sum p(i, i) = .0315$$

while if letters were independent it would be

$$\sum p_i^2 = .0670$$

Appendix 5

A Theoretical Case Where All Invariant Statistics of E Are Independent of K.

By an invariant statistic of a sequence of letters $E = \ldots m_{-2}$ $m_{-1}\ m_0\ m_1\ m_2 \ldots m_3 \ldots$, we will mean a statistic which is averaged along the length of the sequence E. More precisely a statistic of the form:

$$\lim_{n \longrightarrow \infty} \frac{1}{(2n+1)} \left(F(E_{-n}) + \cdots + F(E_{-1}) + F(E) + F(E_1) + F(E_2) + \cdots + F(E_n)\right)$$

where F is any function whose argument is a possible sequence, and $E\pm_n$ is the sequence E shifted N letters to the right or left. Such statistics as the relative frequency of a given letter, of a given n-gram, transition frequencies, and frequencies with which letter i is followed by letter j at a distance n are all invariant.

We will describe a system in which every invariant statistic which the cryptanalyst can construct from the (infinite) intercepted E is independent of both K and M, and thus gives no information to him. This effect and still more occurs with the ideal ciphers of course, but here it is obtained independently of the original message statistics and without any matching of the cipher to the language.

Let N be a "random" sequence of letters;

$$N = \cdots n_{-2}\ n_{-1}\ n_0\ n_1\ n_2 \cdots n_s \cdots$$

this is supposedly a known sequence (to the enemy) and thus a part of the system, not of the key. Apply any simple cipher to the message and then add N letter by letter to the result (mod 26). The "sum" is the enciphered message. It is evident that any invariant statistic on E will be (with probability 1) the same as that for a random sequence. Hence it is independent of both K and M.

We need hardly add that such a system is easily broken—the enemy merely subtracts N from E and then solves the simple residual cipher, which may often be done with invariant statistics.

Appendix 6

Maximum Repetition Rate in Compound Systems for a Given Total Key

We consider briefly the question of how to arrange the component periods in a compound Vigenère or Transposition system to obtain the longest period for a given total key size. If the component periods are $P_1, P_2, \ldots P_s$ it is clear that they should be coprime. Otherwise the total key, which is $\sum P_i$, could be reduced without changing the period, which is the least common multiple of the P_i, merely by deleting a factor which appears in several of the P_i from all but one. Also each P must be a power of a prime, for if it contains two primes, it can be divided into these parts, reducing the key and not affecting the period. Thus the component periods are selections from the series of primes and powers of primes:

$$A : 2, 3, 4, 5, 7, 8, 9, 11, 13, 16, 17, 19, 23, 25, 27, \ldots$$

the selection being pairwise coprime.

It appears from empirical evidence that the best choice of component periods for a given total size S is found by the following process.

1. Determine the largest M such that $\sum_1^M p_i \leq \ S$ where the p_i are the primes in increasing order. This is the maximum number of periods where the periods are coprime, and is the number of periods to be used.
2. Choose from the sequence A, M elements, consecutive except for the fact that no prime is represented more than once, the M elements being as great as possible with sum $\leq S$.
3. If the sum is $\leq S$ move as many as po ssible of the top elements in this block up a notch in the sequence A, still satisfying the conditions on the sum and coprimality.
4. Repeat 3 to either part of the original block if possible. This process eventually ends and apparently gives the proper decomposition.
   For example with $S = 50$, the sum of the first 6 primes is 41, of the first 7 is 58. Hence 6 periods will be used.

we have

$$11 + 9 + 8 + 7 + 5 + 3 = 43$$
$$13 + 11 + 9 + 8 + 7 + 5 = 53$$

hence we start with the block 11, 9, 8, 7, 5, 3. The top 4 elements 11, 9, 8, 7 can be moved up a notch to give

$$13 + 11 + 9 + 8 + 5 + 3 = 49$$

No further improvement seems possible. We obtain

$$P = 13 \times 11 \times 9 \times 8 \times 5 \times 3 = 154,440$$

The products and sums of the first n primes are given below:

n	1	2	3	4	5	6	7	8	9
p_n	2	3	5	7	11	13	17	19	23
Sum	2	5	10	17	28	41	58	77	100
Product	2	6	30	210	2310	30030	510510	9699590	223092870

C. E. SHANNON

Att.
Figures 1-25

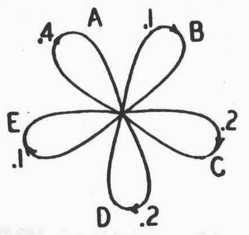


FIG. 1

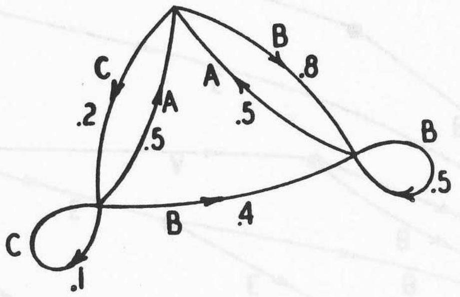


FIG. 2

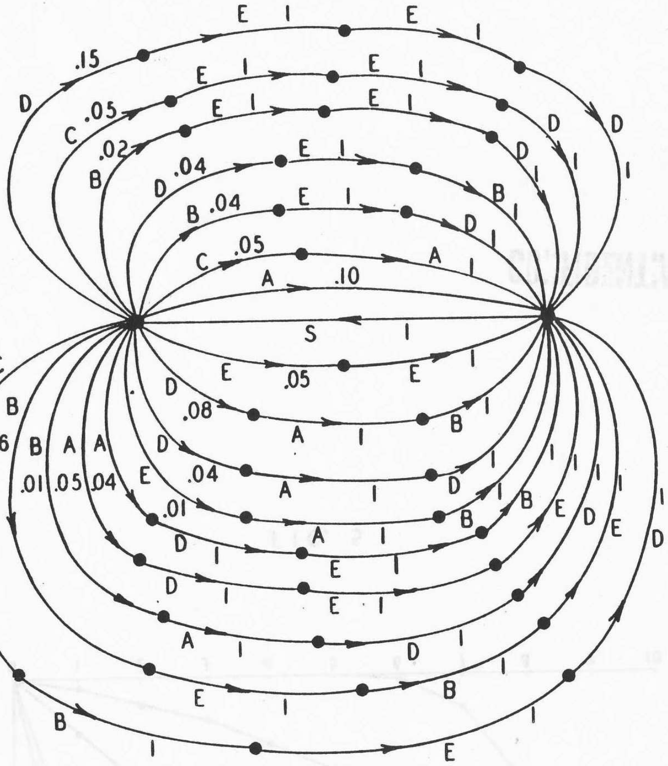

E 1 E 1
.15
D
E 1 E 1
C .05
D D
E 1 E 1
D 1 1
.02
B
D .04 E 1
B 1
B .04 E 1
D 1
C .05 A 1
A .10
S 1
C
E .05 E 1
D
.11 B
.08
A 1 B 1
.16 B A A
D
.01 .05 .04 E .04
A 1 D 1
.01
A 1 B
B E D E D
D 1 E 1
A
D 1 E 1
1
A 1 D 1
E 1 B
B
1 E

FIG. 3

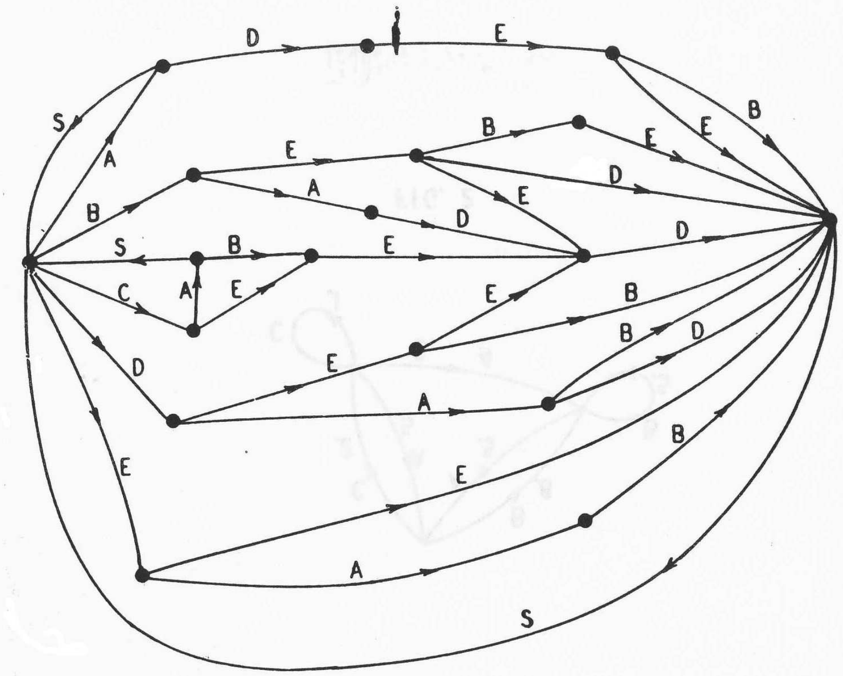


FIG. 4

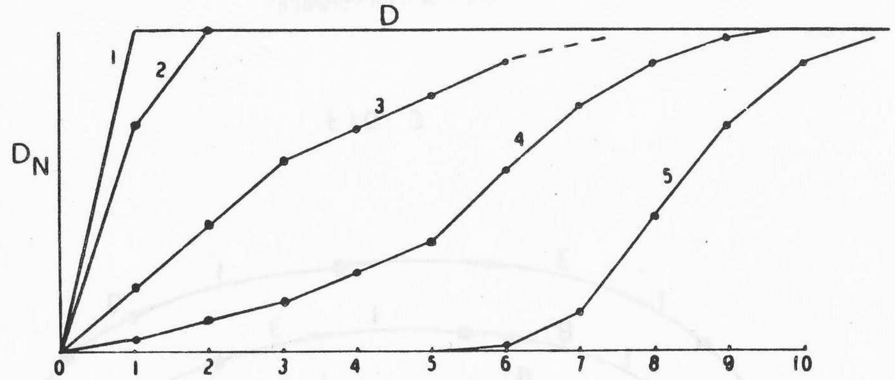


FIG. 5

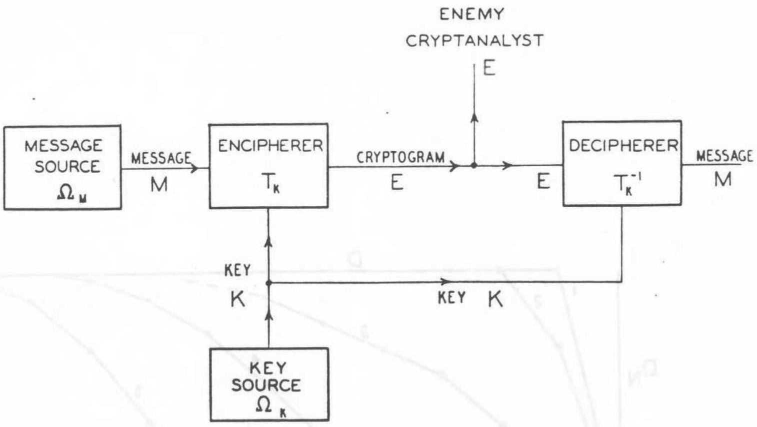

ENEMY
CRYPTANALYST
E
MESSAGE
SOURCE
Ω_M
MESSAGE
M
ENCIPHERER
T_K
CRYPTOGRAM
E
E
DECIPHERER
T_K^{-1}
MESSAGE
M
KEY
K
KEY K
KEY
SOURCE
Ω_K

FIG. 6

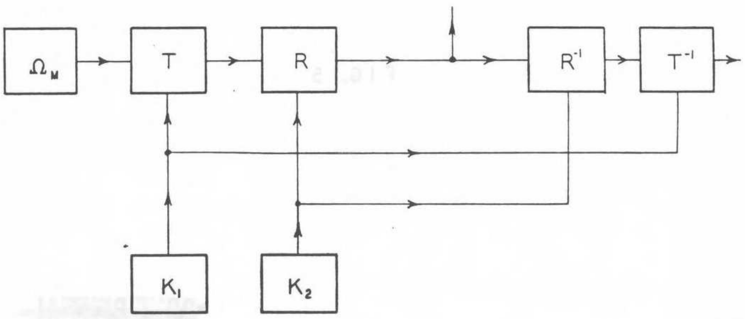
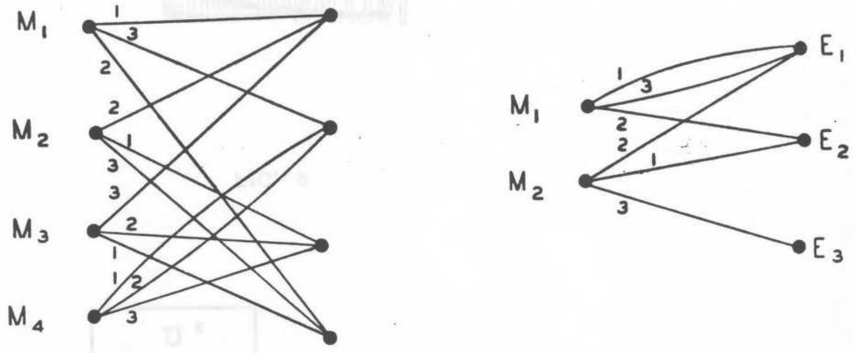

Ω_M
T
R
R^{-1}
T^{-1}
K_1
K_2

FIG. 8



CLOSED SYSTEM

NOT CLOSED

FIG. 7

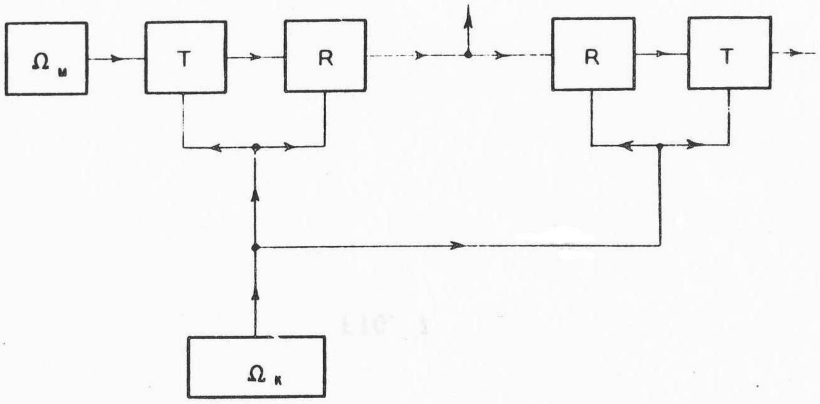
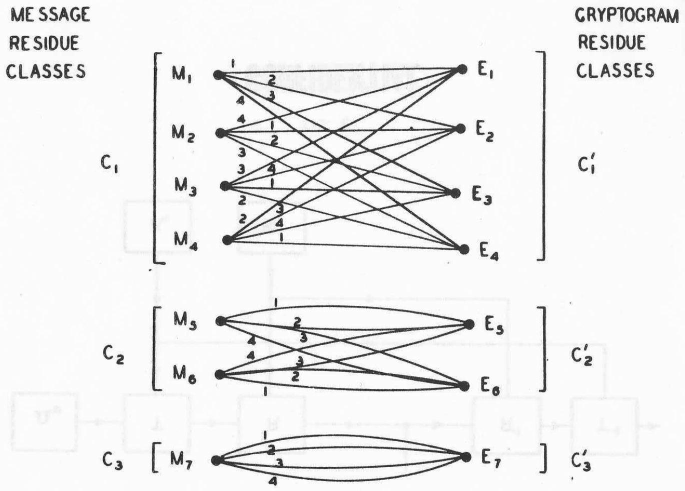
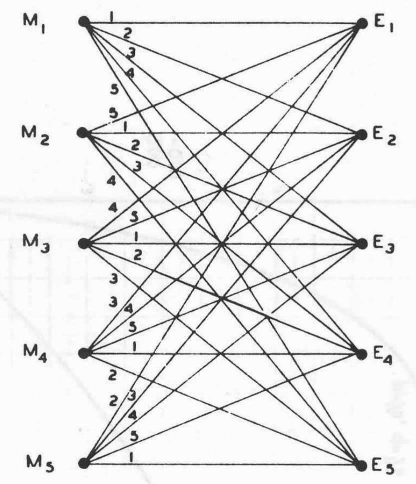


FIG. 9



PURE SYSTEM

FIG. 10



PERFECT SYSTEM

FIG. 11

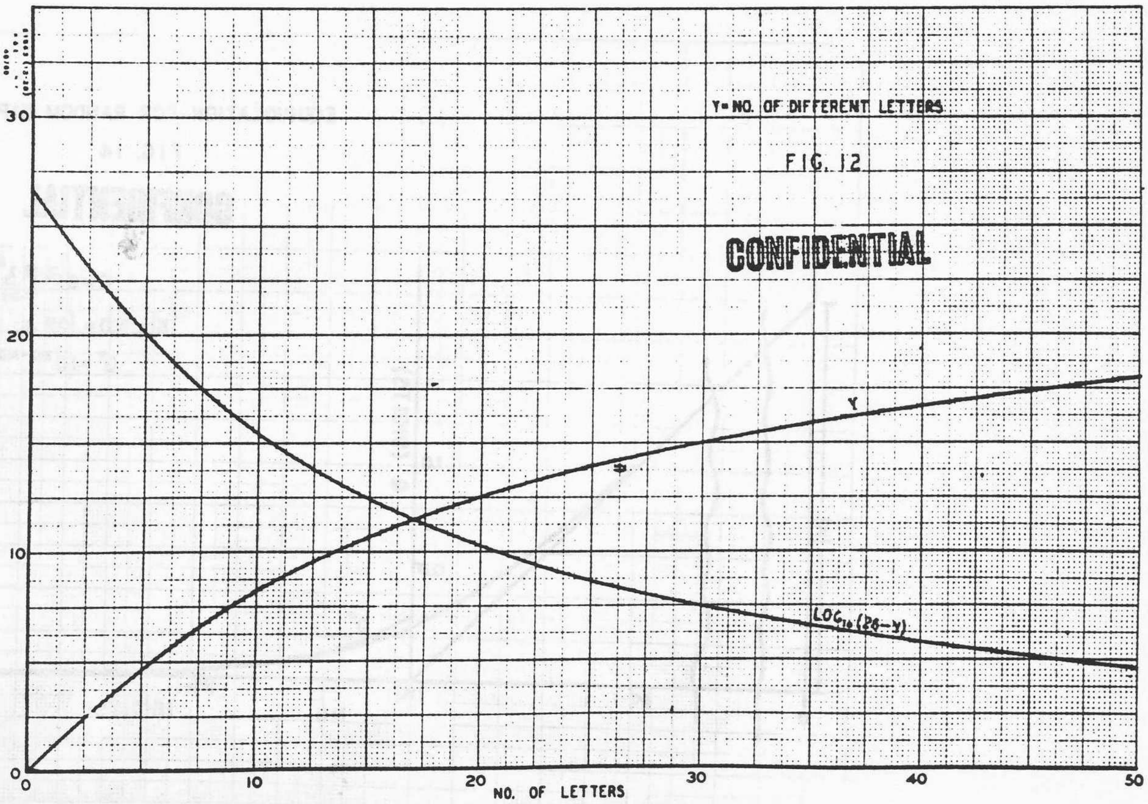
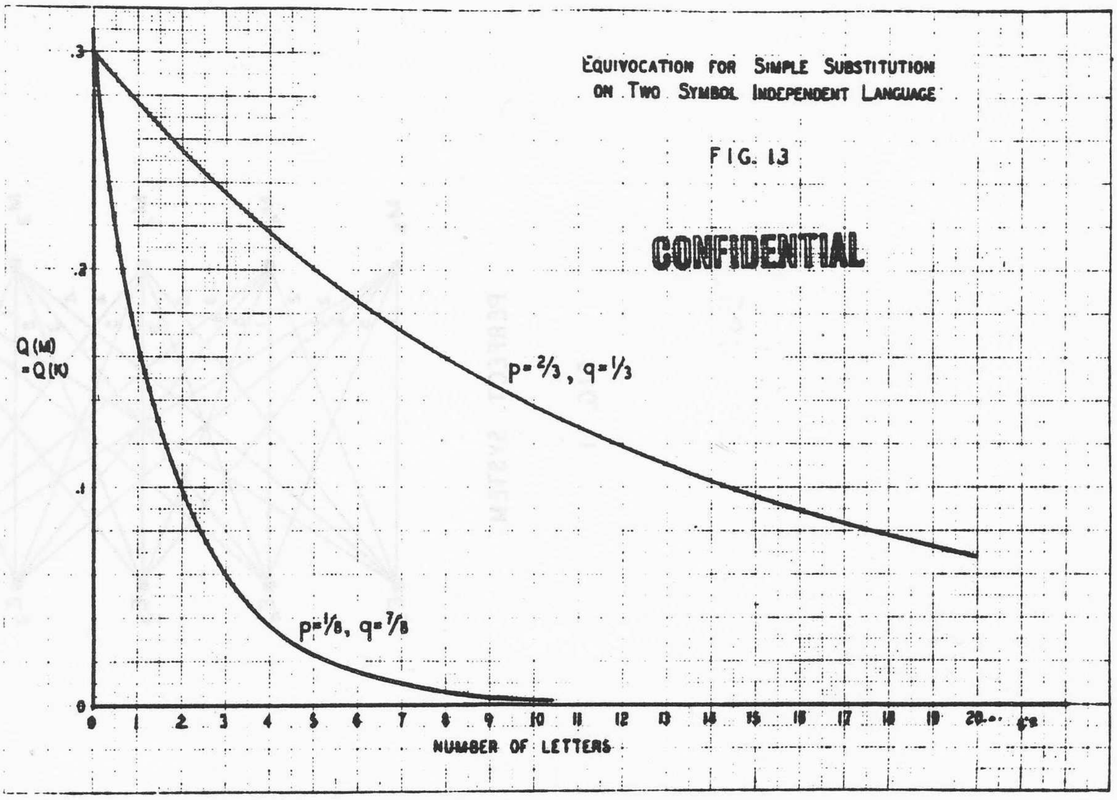


FIG. 12

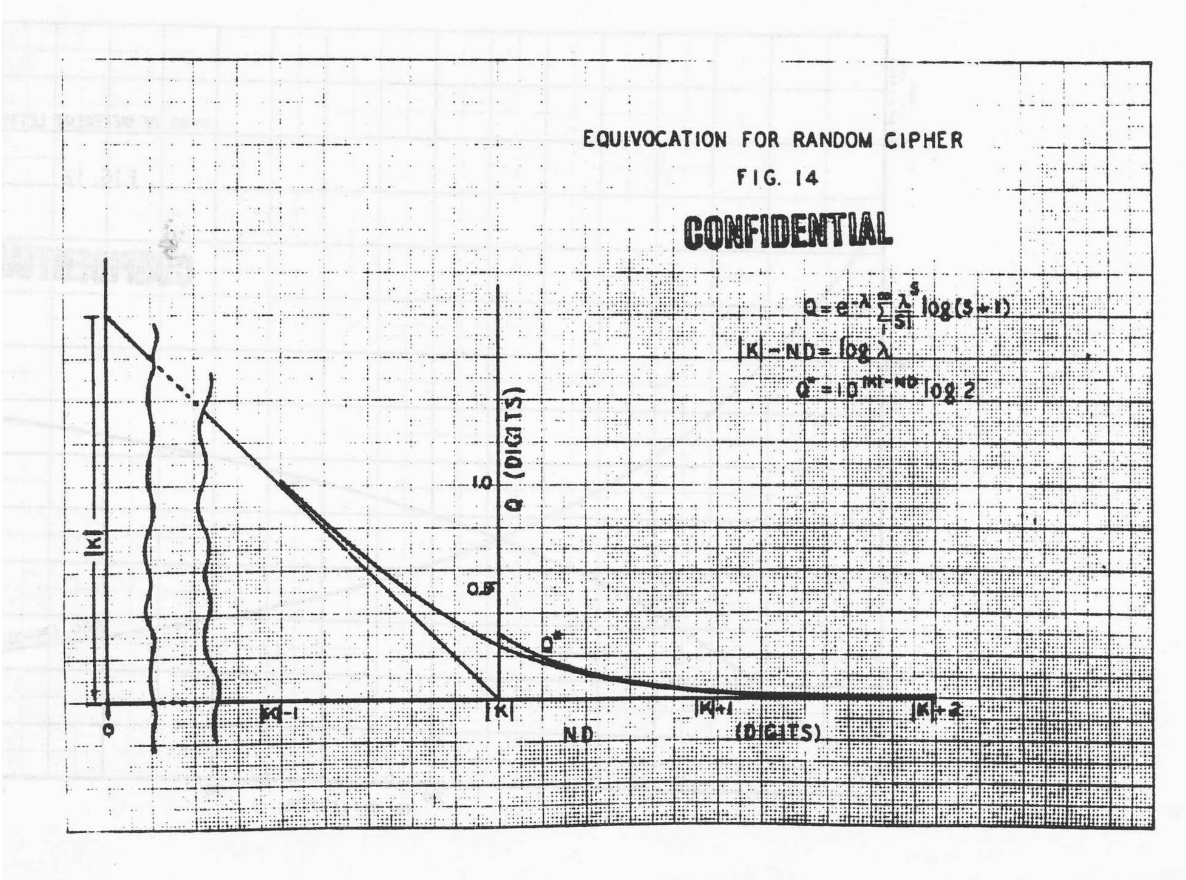
CONFIDENTIAL



EQUIVOCATION FOR SIMPLE SUBSTITUTION ON TWO SYMBOL INDEPENDENT LANGUAGE

FIG. 13

CONFIDENTIAL



EQUIVOCATION FOR RANDOM CIPHER

FIG. 14

CONFIDENTIAL

$$Q = e^{-\lambda} \sum_{1}^{\infty} \frac{\lambda^{S}}{S!} \log (S+1)$$

$$|K| - ND = \log \lambda$$

$$Q^{*} = 10^{|K| - ND} \log 2$$

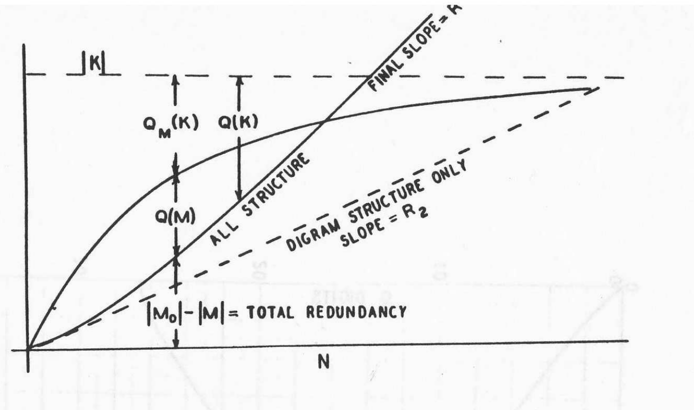
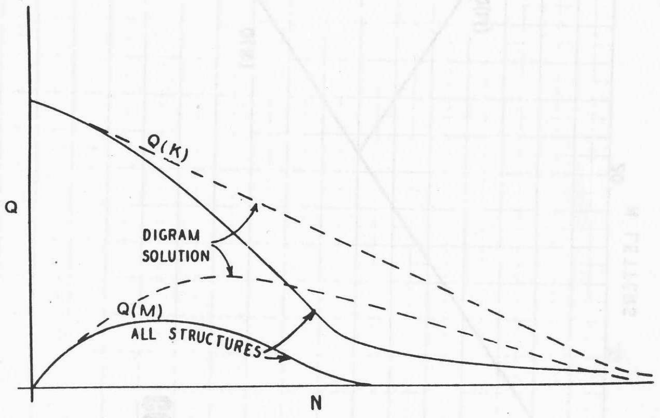
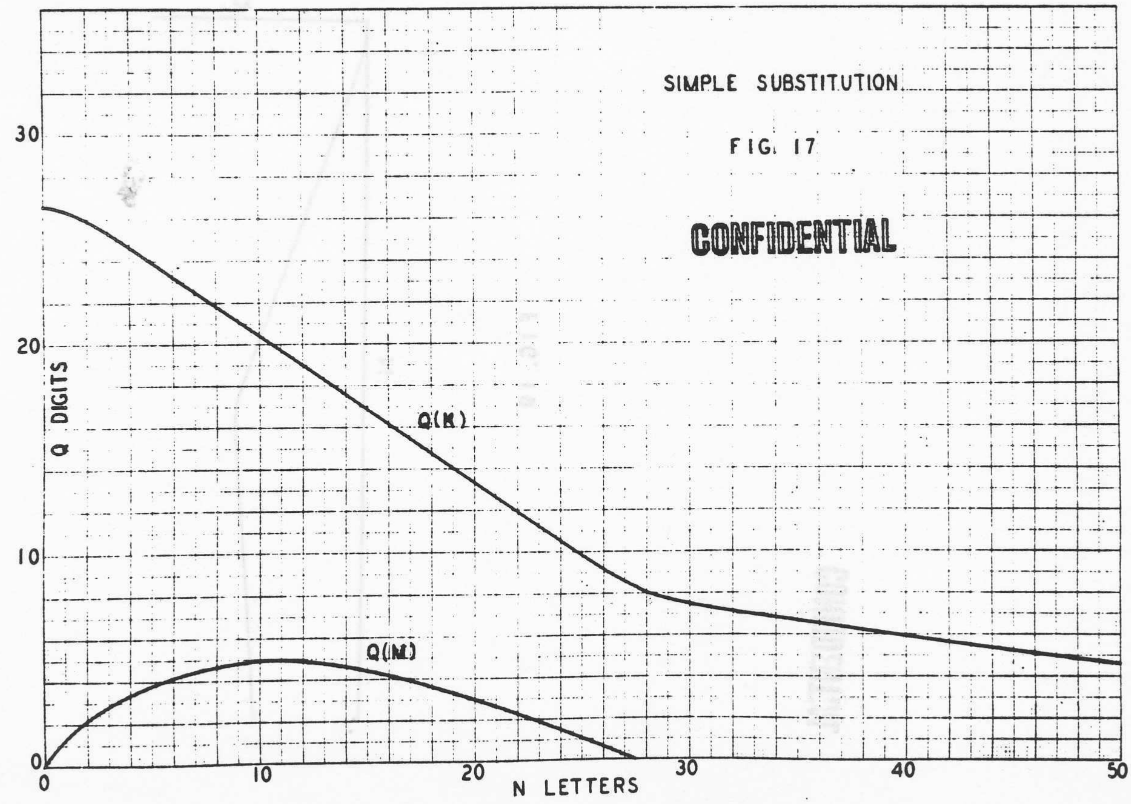


FIG. 15



CALCULATION OF Q CURVES

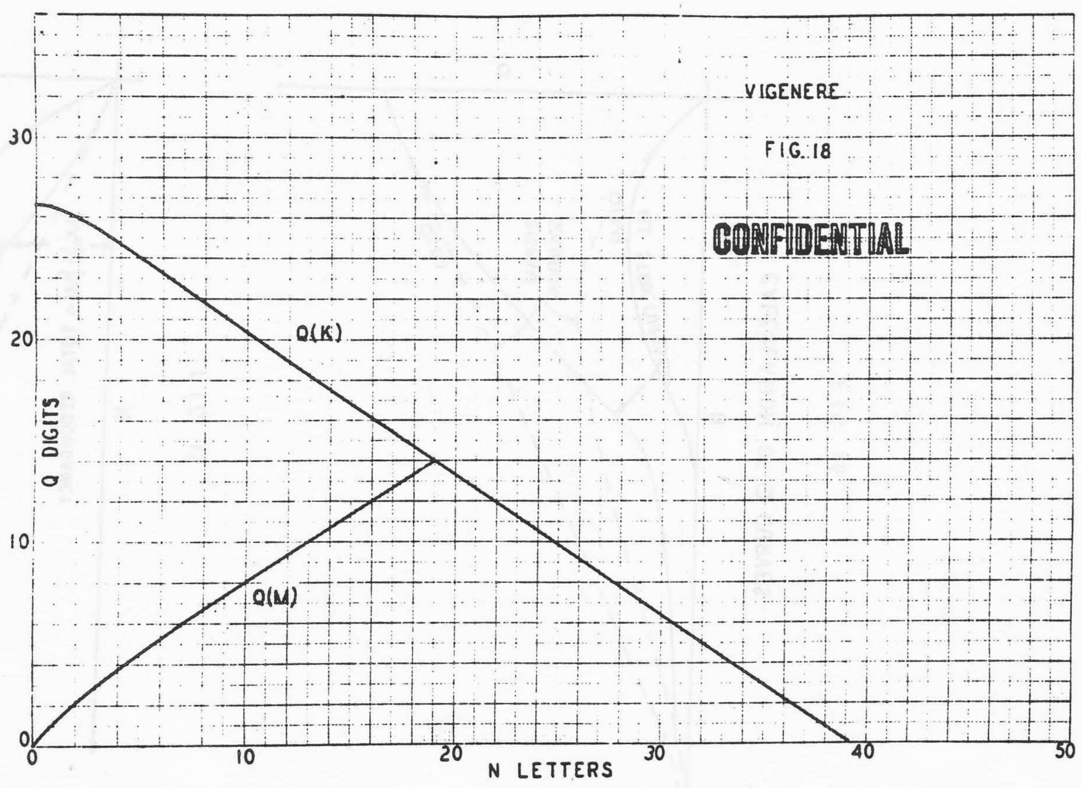
FIG. 16



SIMPLE SUBSTITUTION.

FIG. 17

CONFIDENTIAL


VIGENERE
FIG. 18
CONFIDENTIAL
Q(K)
Q(M)
Q DIGITS
N LETTERS
0
10
20
30
0
10
20
30
40
50

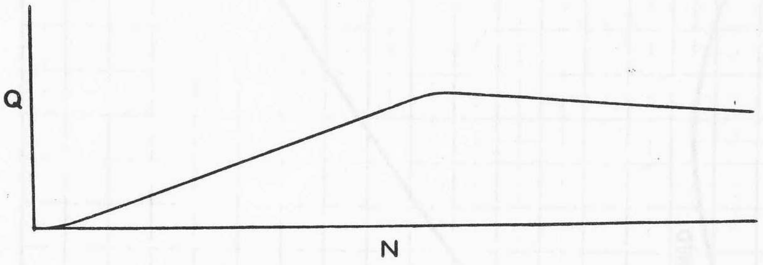
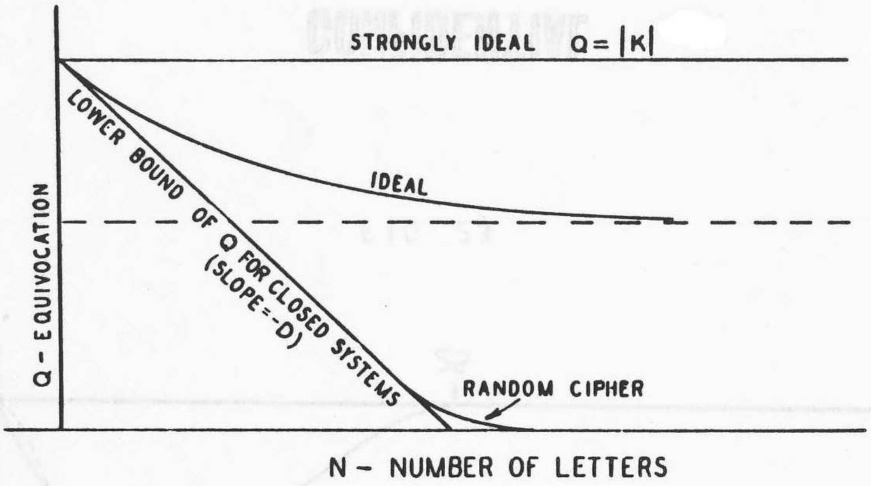


FIG. 19



IDEAL CHARACTERISTICS

FIG. 20

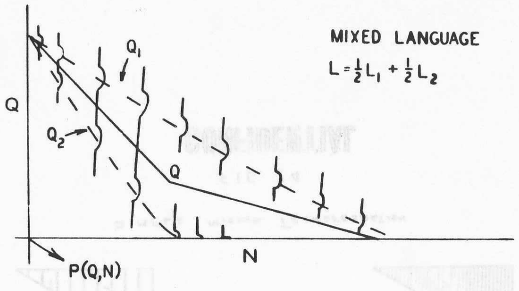
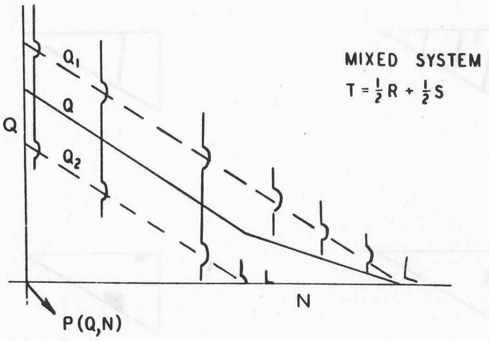


FIG. 21



CONFIDENTIAL

FIG. 22

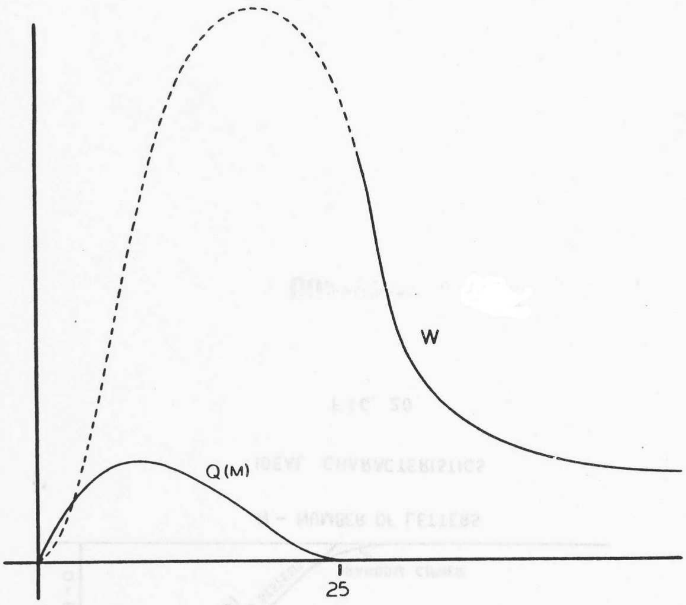
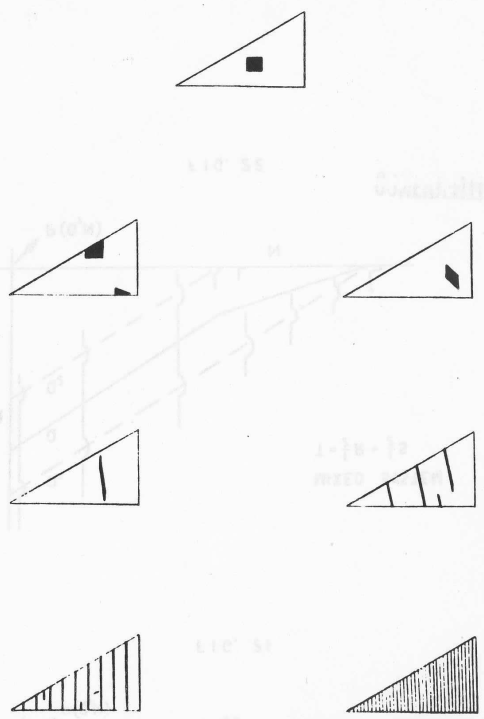
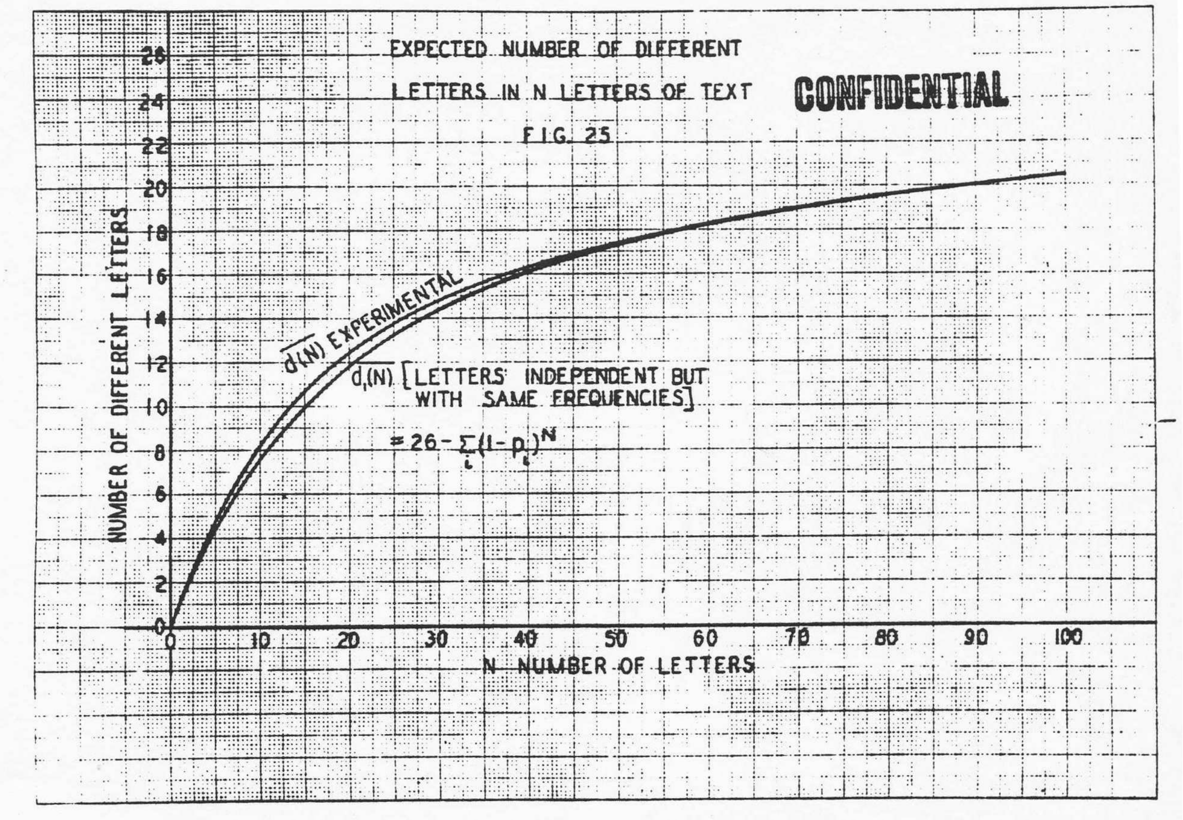


FIG. 23



SIMPLE MIXING TRANSFORMATION.

FIG. 24



EXPECTED NUMBER OF DIFFERENT LETTERS IN N LETTERS OF TEXT

FIG. 25

CONFIDENTIAL