

Dr E J Yannakoudakis
Postgraduate school of Computer Science
University of Bradford
Bradford
West Yorkshire BD7 1DP
ENGLAND

ABSTRACT

A principle of information science states that the entropy of a set of symbols is maximised when the probability of occurrence of each becomes the same. This paper presents the results of a number of experiments which utilise this principle to construct fixed length keys from pertinent fields in order to locate and retrieve unique records as well as clusters with lexically homogeneous information. Each key incorporates codes derived by various positional selection methods and their discriminating strength proves to be well over 95%.

1. INTRODUCTION

The frequency of occurrence and other statistically orientated results derived from items in machine readable collections have recently formed the basis for the design of optimal information structures (Yannakoudakis and Wu, 1982). The aim has been to generate equiproportional groups of items and to use the concepts of 'entropy' and 'variance' to measure and compare alternative arrangements. The results established that the variance is a more efficient measure of equiproportionality.

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Association for Computing Machinery. To copy otherwise, or to republish, requires a fee and/or specific permission.

This paper employs the variance to group the 26 letters of the English alphabet into n equiprobable sets ($n < 26$) based on the probabilities of letters in bibliographic records. Each of the n sets is then assigned to a new symbol in a code alphabet in order to reduce each title field into a fixed length code. This is then supplemented by characters selected from the Edition, Publisher, Date and Volume of each bibliographic record and keys of 15 and 16 characters long are derived and used to retrieve source records. Grouping thus maximises the information representing capability of the code alphabet and is defined as the mapping of the source alphabet $A=(1,\dots,m)$ onto a code alphabet $G=(1,\dots,n)$ where $n < m$ such that the sets in G are equiprobable within a specified deviation.

Nugent et al (1962, pp 49) consider 'grouping' as a non-linear programming problem and use the following formula to select the best improvement for a series of arrangements:

$$(1) \text{ Find the maximum } I \text{ where } I = (\Delta G_i^2 + \Delta G_j^2 - \Delta' G_i^2 + \Delta' G_j^2)$$

where I = improvement, ΔG_i , ΔG_j are deviations of groups G_i and G_j from equiprobability before switching, and $\Delta' G_i$, $\Delta' G_j$ are deviations after the tentative switch is made.

The starting point is an arbitrary arrangement of the 26 letters into the desired number of sets. To improve the group alphabet (i.e. reduce the total variance), a series of tests is made in which each member of each set is tentatively switched with each member of every other set. An efficient algorithm for this process is presented in detail by Yannakoudakis and Wu (1982).

The decision to use single characters as the elementary units for equiprobable alphabet coding is justified by two main reasons. Firstly, all characters (A-Z and 0-9) are easily identified and their probabilities are easily estimated, and secondly, the use of any other units, such as bigrams and trigrams would be rather tedious to count and either way rather subjective (dependent on the specific types of records available). Moreover, the letters have been ranked by other authors and certain distributions have been recognised.

The use of derived search keys for information retrieval offers a first degree tool for unique and clustered retrieval of lexically homogeneous information. Although the aim here has been to increase the percentage of unique keys derived and hence to decrease the number and size of the clusters, it is obvious that once an optimum algorithm has been established the key can be shortened (by excluding certain fields from coding or by reducing the number of characters per code) in order to increase the size of selected clusters.

All experiments were carried out using Machine Readable Catalogue (MARC) records and the field identifiers or tags which were searched were 110 (Inverted corporate name), 245a (Title), 245b (Subtitle), 245g (Volume or Part number), 245h (Volume or part Title), 250a (Edition), 260b (Publisher) and 260c (Date of publication).

2. GENERATION OF EQUIFREQUENT SETS

The frequency of occurrence of the letters A to Z and the space character as calculated by Yannakoudakis (1979) is presented in Table 1. These were calculated using tags 245 (title) and 110 (inverted corporate name) from bibliographic files. Out of the 9,362,924 characters processed, 7,908,100 were A to Z (84.46%). The probability of space was estimated on the total number of characters processed and the probabilities of the letters were estimated on the total codable characters. Their rank-frequency distribution in linear and logarithmic scales will be seen in Figures 1 and 2. It will be observed that ranks 3, 4, 5 and 6 corresponding to the letters T, I, N and O respectively, represent almost equiproportional distributions. The graphs however indicate quite clearly that generally, the letters deviate very greatly from the ideal and that their distribution is hyperbolic.

Table 1. Occurrence frequencies of letters
in MARC records (tags 245, 110)

LETTER	OCCURRENCE	PROBABILITY
A	657483	0.083
B	206160	0.026
C	309989	0.039
D	322251	0.041
E	846780	0.107
F	160458	0.020
G	164536	0.021
H	285541	0.036
I	609767	0.077
J	33982	0.004
K	61894	0.008
L	362654	0.046
M	216785	0.027
N	603827	0.076
O	601970	0.076
P	170906	0.022
Q	7452	0.001
R	575760	0.073
S	486219	0.061
T	611711	0.077
U	205010	0.026
V	74209	0.009
W	83978	0.011
X	15221	0.002
Y	220855	0.028
Z	12702	0.002
SPACE	1454824	0.155

Formula (1) was programmed so that the letters were arranged into approximately equiproport sets and seven groups were generated with cardinalities 7, 8, 9, 10, 11, 12 and 13 and the variance of each was recorded. Table 2 contains an example of all the arrangements obtained for a group of cardinality 10. Arrangement number 7 indicates the optimal distribution and the loop starts at arrangement number 8 where it becomes impossible to improve the variance.

Since the natural alphabet is of fixed size (26), one logically expects that the higher the cardinality of the group the greater will be the deviation of its members from equiprobability. To prove this hypothesis, the variance of each optimal quasi-equiproport group was plotted against its cardinality and the result will be seen in Figure 3. The graph illustrates quite clearly that the hypothesis is

true for cardinalities greater than 10 where an almost exponential increase in variance can be observed, but not equally so for cardinalities less than 10 where it remains, comparatively, rather constant.

This is a most striking result leading to two fundamental conclusions: (a) For an optimal n-ary code alphabet, n must be less than or equal to 10, and (b) for maximum record discrimination and key efficiency n must be as close to 10 as possible.

Table 2.
Quasi-equifrequent letter groups.

1.	E	AJW	FT	MR	BO	NPZ	GIX	CS	DLQV	HKUY	
									Variance =	0.004955	
2.	E	AJW	FT	MR	BO	NPZ	GIQ	CS	DLVX	HKUY	
									Variance =	0.004719	
3.	E	AJW	FT	RU	BO	NPZ	GIQ	CS	DLVX	HKMY	
									Variance =	0.004543	
4.	E	AJW	PT	RU	BO	FNZ	GIQ	CS	DLVX	HKMY	
									Variance =	0.004443	
5.	E	AJW	PT	BR	OU	FNZ	GIQ	CS	DLVX	HKMY	
									Variance =	0.004382	
6.	E	AJW	PT	BR	OU	GNZ	FIQ	CS	DLVX	HKMY	
									Variance =	0.004377	
7.	E	AJW	IP	BR	OU	GNZ	FQT	CS	DLVX	HKMY	
									Variance =	0.004365	

3. CHOOSING THE APPROPRIATE FIELDS

Before the key is derived it is necessary to adopt certain guideline criteria on the basis of which it becomes possible to select appropriate fields for coding. A number of questions must therefore be answered and these include the following:

- (a) Frequency of use: To what extent is a field used either for cataloguing or for record control in general ?
- (b) Usage errors: Is the field liable to be misused or subject to any of the known errors of transposition, transcription, etc. ?
- (c) Discriminating strength: Given an average field length what is the maximum possible number of different entries that can be constructed ?

(d) Compatibility: Is the field compatible with others selected for use in coding ? Is there any overlap ?

(e) Ease of location: Can the field be isolated from within the record and processed easily ?

Some of the MARC fields are of course easy to evaluate and assess their suitability; perhaps because we have all in the past, somehow, attempted to use them in one way or another while searching a library catalogue. Ayres et al (1968) give details of a survey carried out in a large scientific special library on the comparative accuracy of the author and title information which the user brings to the catalogue. Their results show the title to be more accurate. Ryans (1978) reports the results of a study of cataloguing records input into the OCLC database by participating libraries. The two fields on the catalogue record that generated the majority of errors were the subject heading (31%) and the collation fields (25%).

Authors' names pose many problems especially when the name is an approximate translation from another language. Take for example the name of the present author. If it were found in a database in Greece, 'J' would be interpreted as the father's Christian name but if it were used in England 'J' would be a second Christian name. Moreover, the surname can be translated into Yiannakoudakis or Giannakoudakis depending on whether the Cataloguer happens to be Athenian or Cretan.

An obvious field for use in any coding scheme must surely be that of the title. This has of course in the past been used for the creation of codes and in a recent study it formed the sole basis for the creation of keys used to test merging and identification of duplicate records in multiple files (Williams et al, 1977). The key size varied from 20 to 40 characters long whereas the present keys are not more than 16 characters long.

Ayres (1974) describes a novel type methodology for the construction of control numbers for bibliographical records and suggests the use of the information contained in the Title, Language, Date, Edition, Volume and Publisher fields. These proposals are regarded as the first challenge to the ISBN, and this is confirmed by an evaluation and improvement of the scheme carried out by Yannakoudakis et al, 1980. The original Tables suggested by Ayres (1974) were revised by Beale and Lynch (1975).

In conclusion, the elements Title, Date, Edition, Volume and Publisher seem the most logical for use in the key. Alternative reasons for their choice are given by Ayres (1974, 1976). The fields conform to retrieval criteria which were applied in conjunction with key design objectives and as such they formed the basis of our tests.

The number of characters coded from each field was determined empirically and it should be clear that the core of the key lies within the title field. The contribution of the date, edition, and volume fields to the length of the key is obviated by the characteristics of the fields themselves. For example one alpha character will discriminate up to 26 editions and two numeric characters suffice for up to 100 volumes. The two characters of the publisher element are sufficient to discriminate cases where the same work is published by several different publishers in the same year. There are also rare cases where the same title is given to different works by different publishers in the same year.

4. INITIAL SPECIFICATION OF THE KEY

What follows is the complete specification of all six elements of the key which utilises the seventh quasi-equiprevalent arrangement of letters from Table 2 for the title element.

(1) EDITION (250a)

Zero is used to represent all cases where the first character in the edition statement is non-numeric. This would cover cases such as New ed., Rev. ed., Paperback ed. and [New ed.]. The digit 1 is used when tag 250a is not present. When the field contains one numeric character this is used, and when it contains more than one the last is used. This means that the 2nd and 12th edition and the 1982 edition will all be coded as 2.

(2) PUBLISHER (260b)

The improved Table suggested and tested by Beale and Lynch (1975) is used:

Letter	Code
-----	-----
BG	0
C	1
DUV	2
EW	3
FL	4
HQ	5
ISX	6
JMZ	7
KOY	8
NR	9

The Table omits the letters A, P and T in order to avoid coding 'and' in cases such as Mills and Boon, 'the' in cases such as the Council of the BNB, and 'Publications' or 'Press' in cases such as Pergamon Press or Dover Publications. The code operates on the initial letters of publisher information. Where this would give only a one figure code, the second letter of the first name is used (e.g. Dover Publications is coded as 28). Where no code at all would result, the second and third letters of the first name would be used (e.g. Pergamon Press is coded as 39).

(3) DATE (260c)

The last three digits of the date of publication are used. Where no date is given three zeroes are used. Where the date is given as 1982-83 the last three digits are used (283 in this case).

(4) VOLUME (245g)

Numbers are allocated by taking the first two numeric characters of the volume or part number. Where volumes are also divided into parts the code will utilise both. For example vol. 1 part 2, and vol. 1 part 3 will be coded as 12 and 13 respectively.

(5) 1st TITLE (245a,b)

The following quasi-equifrequent sets (see Table 2) are used:

Letter	Code
-----	----
E	0
AJW	1
IP	2
BR	3
OU	4
GNZ	5
FQT	6
CS	7
DLVX	8
HKMY	9

The method of application is to code two initial letters from the beginning of the title. Zero is used where the title is not covered or partly covered by code.

(6) 2nd TITLE (245a,b,h)

The sets used for the 1st title element are also used here, however the method is to code the first three non-initial letters and the last two non-initial letters.

5. TEST RESULTS AND KEY IMPROVEMENTS

Throughout this section, and in order to avoid unnecessary repetition of the terminology regarding the various efficiency measures, when the results are given as four values they will denote, (1) percentage of distinct keys, (2) percentage of unique keys, (3) mean cluster, and (4) cardinality of the largest cluster. The mean cluster was calculated using the following formula:

$$(2) M = \frac{1}{n} \sum_{j=1}^{\eta} (|K_j| - 1)$$

where n is the total number of records and $|K_j|$ is the cardinality of each cluster.

The complete British National Bibliography (BNB) 1971 file containing a total of 30,651 records was analysed. The results were, 97.550% distinct keys, 96.127% unique keys, 0.1326 mean cluster and a maximum cardinality of 26.

The same method was applied on a file from Southampton University (SUL) which contained 15,219 records and the results were, 77.535%, 67.074%, 1.6122 and 39 respectively. However when the retrievals were examined it was discovered that only the title field had been picked up under the tags used and therefore the results reflected the discriminating capability of the title alone.

Following examination of the BNB clusters it was noticed that the inclusion of T, O and A in the quasi-equifrequent sets had resulted in the coding of articles such as 'The', 'Of', 'And', 'An', and others. Also, the clusters seemed to have discriminating information in tag 245g (Volume/Part number). It was therefore decided to search tags 245a, 245b and 245g for the second title element with the following Table (Beale and Lynch, 1975):

Non-initial letters Table

BFX	1
C	2
D	3
GKZ	4
JVY	5
L	6
MQW	7
PH	8
U	9

The following sets were formed in order to exclude the

coding of the various articles from the title element.

Initial letters Table

BLZ	1
CK	2
DEQ	3
FWX	4
GRV	5
HM	6
INJ	7
PUY	8
S	9

The introduction of these Tables produced a minor improvement. The results from BNB were 97.576%, 96.366%, 0.1911 respectively with a maximum cardinality of 43. On the SUL file the method gave 82.804%, 74.650%, 1.0672 and 32. In this case the SUL key included the date, publisher and the title elements only.

By now it was clear that the original quasi-equifrequent alphabet could not be used without the elimination of certain highly frequent letters peculiar to the bibliographical records. On the other hand it was necessary to establish the true value of such an alphabet. To this end the following test was carried out. Instead of coding the digit of each letter set for the title and publisher elements, the program was altered so that the corresponding letter was included in the key. The rest of the specification for the elements of date, edition and volume remained the same as in the last test. When applied to BNB file this method gave 97.746% distinct keys and 96.647% unique keys. The improvement (+1.099% uniqueness) was indeed insignificant. The mean cluster was 0.1821 as compared to the previous 0.1911 and the largest clustered set had the same cardinality (43). Similarly, when applied to the SUL file the method gave 88.738%, 80.216%, 0.2950 and 7 respectively. On this file therefore the method gave a more even distribution of clusters.

These results tempt us to conjecture that it makes, practically, no difference whether a digit or a letter is coded, and that the letters of the coding Tables are indeed equifrequent.

Throughout the above tests and when the title and publisher elements were coded, searches for initial letters were performed as follows: When a letter is preceded by any of the symbols, space, apostrophe, (, ", [, /, -, and full stop, it is taken as an initial letter. All other letters are regarded as non-initial.

Following visual examination of full MARC clustered records, it was discovered that in long titles the discriminating characters were usually at the end. Also, a substantial number of clusters had the edition field appended at 245g rather than in 250a as was expected, and searched for. Two immediate remedies to this were implemented in the program as follows. Firstly, the edition field was selected from 245g and secondly, the initial letter indicators were restricted to space, (, ", [and /. Also, to avoid coding 245g twice, the title element was filled from 245a and 245b only rather than from 245a, 245b and 245g as was previously the case.

The BNB file was again used for the test and the results were, 96.183%, 94.411%, 0.2868 and 26. This implied that although the reduction of initial letter indicators had worked in the long titles, the edition element required revision because it was clashing with the volume element which also operated on 245g. To rectify this, the program was altered to check the first character from 245g and if it is alpha to use it in the edition element. If not, to select the first digit from the end of 245g. This would enable full utilisation of 245g since the volume element would select the first two digits from the beginning of 245g. The results from BNB this time were slightly better than the previous test. The measures were 96.423%, 94.826%, 0.2801 and 26 respectively.

At this stage it was decided to put together all experience gained and to code without the use of the edition element which didn't seem to contribute too much in key uniqueness. Instead, a new element, the WEIGHT, was introduced which contained the last decimal digit from the count of all characters in the title. For example, when the count is 28 the digit 8 is used and when the count is 30 the digit 0 is used. Also, the title elements are filled by selecting the letters themselves rather than the corresponding digits. Thus, the first title element uses the first two initials excluding T, O, S and A. The second title element now uses the first two non-initial letters and the last three non-initial letters (in both cases the letters A, E, I, N, O, R, S and T are excluded). Similarly, the publisher element is now filled by taking the first two initial letters excluding A, P and T. The volume element is now filled by selecting the last two digits from 245g.

The method this time gave 98.355% distinct keys and 97.573% unique keys. Also, the mean cluster had been reduced from 0.2801 to 0.1162 and the cardinality came down to 20 from a maximum of 43 in the second and third tests. However the exclusion of the edition element meant that records such as:

245a : BIG BALL
245c : BY CECILIA AND JEAN HINDE
260a : EDINBURGH
260b : OLIVER AND BOYD
260c : 1970

and

245a : BIG BALL
245c : BY CECILIA AND JEAN HINDE
250a : [I.T.A. ED.]
260a : EDINBURGH
260b : OLIVER AND BOYD
260c : 1970

generated the same code:

Weight : 3
Date : 970
1st Title : BB
2nd Title : GLLG
Volume : 00
Publisher : OB

It was therefore decided to retain the edition and the weight elements and to simplify the construction of the key on the basis of the following algorithm.

(1) EDITION (250a)

Select the first alpha character if there is one, otherwise select the last digit.

(2) PUBLISHER (260b)

Select the first two initial letters excluding A, P and T.

(3) DATE (260c)

Select the last three digits.

(4) VOLUME (245g)

Select the last two digits.

(5) 1st-TITLE (245a,b)

Select the first two initial letters excluding T, O, S and A.

(6) 2nd-TITLE (245a,b)

Select the first two non-initial letters and the last three

non-initial letters. In both cases exclude the letters A, E, I, N, O, R, S and T.

(7) WEIGHT

Count all characters in the title field and separate the last digit from the counter for use in the key.

A final test was carried out on the BNB file with the above specification and the results were, 98.419%, 97.7%, 0.1149 and 20 respectively. It is clear that in all tests carried out the cardinality never actually received values less than 20 and it was therefore essential to establish the characteristics of these clusters. To this end, a number of subroutines was written to scan the keys and to merge the various clusters. The results showed that in most cases the source records were in fact the same and nearly all were H.M.S.O. publications. Some examples of these are:

Card.	Key Generated	Begininning of Title
-----	-----	-----
5	7/971/ROULUCC/0/07/HM	Annual Report And Accounts.
7	7/971/RFPDHCD/0/17/HM	Report and Accounts for the year ended ...
11	0/971/RIULHCD/0/17/HM	Annual Report And Accounts Including Report Of Gas...
20	9/971/RFPDHCD/0/17/HM	Report And Statement of Accounts ...

6. THE AFFECT OF DEGREE OF EQUIFREQUENCY ON KEY PERFORMANCE

The previous section tested the complete key comprised of the elements date, edition, publisher, volume, title and weight. Therefore the true value of equiprequent alphabet coding could not be fully established. To enable the study of the affect of variance on key performance, it was decided to carry out a series of experiments using only the title element (fields 245a, 245b, and 245h). The BNB file was used to generate a 7-character key for each of the first 6,260 records with the following algorithm:

- (1) Take an arrangement from Table 2 and assign the digits 0 to 9. For example,

Set	Digit
---	-----
E	0
AJW	1
FT	2
MR	3
BO	4
NPZ	5
GIX	6
CS	7
DLQV	8
HKUY	9

- (2) Edit fields 245a, 245b and 245h to include full words separated by one space indicating an initial letter.
- (3) Select two initial letters from the beginning of the edited buffer and use the corresponding digits in the key.
- (4) Select three non-initial letters from the beginning of the buffer and use the corresponding digits in the key.
- (5) Select two non-initial letters from the end of the buffer and use the corresponding digits in the key.

Table 2 contains 7 different quasi-equipotent arrangements the last one being the optimal with a variance of 0.004365. Besides, two more arrangements of higher variance, 0.009852 and 0.005983 respectively, were formed. Figure 4 shows that the variance of these arrangements decreases exponentially and therefore the sets form an adequate base for testing the affect of the variance on the key.

All appropriate software was then written to create 9 different files corresponding to each of the 9 different arrangements. Following analyses of each file it was established that a small variance does not necessarily result in high percent uniqueness. For example variance 0.009852 gives 67.236422% unique keys whereas variance 0.005983 gives 66.932907% unique keys. Moreover, the affect of variance on key performance is marginal (see Table 3 for the results from each arrangement).

Table 3. Test results from nine different letter arrangements (7-character long keys)

Arrange- ment	Variance	Percent Distinct Keys	Percent Unique Keys
1	0.009852	76.597444	67.236422
2	0.005983	76.501597	66.932907
3	0.004955	76.932907	67.523962
4	0.004719	76.869010	67.444089
5	0.004543	76.773163	67.316294
6	0.004443	76.773163	67.252396
7	0.004382	76.900958	67.444089
8	0.004377	76.677316	67.268371
9	0.004365	76.677316	67.092652

7. CONCLUSIONS AND FINAL TESTS

The results enable us to draw certain general conclusions on the use of quasi-equiprequent sets for coding as compared to positional selection of letters and to other empirically established methods.

Although equiprequent alphabet coding has a theoretical basis for its methodology, it can not by itself form an integrated mechanism. This must be complemented by other information pertinent to specific types of databases.

The exclusion of certain highly frequent symbols which occur in records prior to equiprequent alphabet coding, in conjunction with positional character selection, form a viable combination for high discrimination and key uniqueness.

Certain hypotheses were put to test regarding the frequencies of letters used in the above tests. One of them was that perhaps the fact that the frequencies were based on the complete tags 245 and 110 even though only part of tag 245 had been utilised by the code, had indeed affected the performance of the key. To test this, and in order to avoid repetition of the tests, the frequencies of letters in 245a (see Table 4) were calculated and used for further investigation. Here again the probability of space was

estimated on the total number of characters processed and the probabilities of the letters were estimated on the total codable characters.

The equiprequent set generator was then run with the new frequencies and quasi-equiprequent groups of cardinalities 7, 8, 9, 10, 11, 12 and 13 were formed. In order to study the change in variance and hence the change in the degree of equiprequency, the corresponding values were plotted and the graph is presented in Figure 5. A direct comparison of the graph in Figure 3 regarding the letters in tags 245 and 110 with Figure 5 indicates that their almost identical slopes present sufficient evidence to disprove the hypotheses and to endorse the original conclusions.

Table 4.
Occurrence frequencies of letters in MARC titles (245a)

LETTER	OCCURRENCE	PROBABILITY
-----	-----	-----
A	310049	0.082
B	51188	0.014
C	160979	0.043
D	137406	0.037
E	417205	0.111
F	84939	0.023
G	84677	0.023
H	139468	0.037
I	306057	0.081
J	6936	0.002
K	27837	0.007
L	167101	0.044
M	105700	0.028
N	294045	0.078
O	307271	0.082
P	88171	0.023
Q	4845	0.001
R	264218	0.070
S	250975	0.067
T	297508	0.079
U	103572	0.028
V	35369	0.009
W	34490	0.009
X	7916	0.002
Y	65493	0.017
Z	6066	0.002
SPACE	642151	0.146

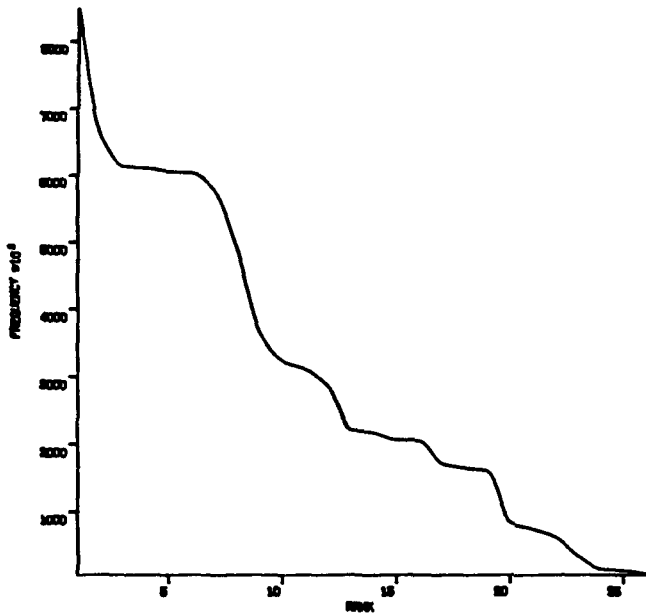


Figure 1. Letter frequency distribution - Linear Scales

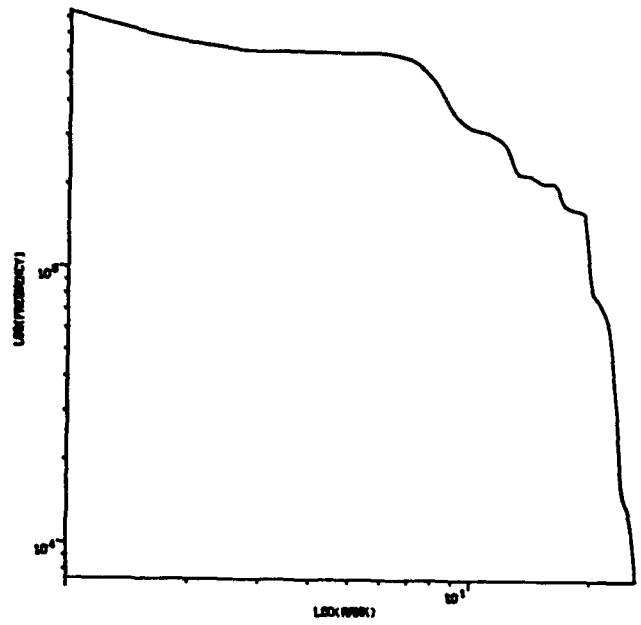


Figure 2. Letter frequency distribution - Logarithmic Scales

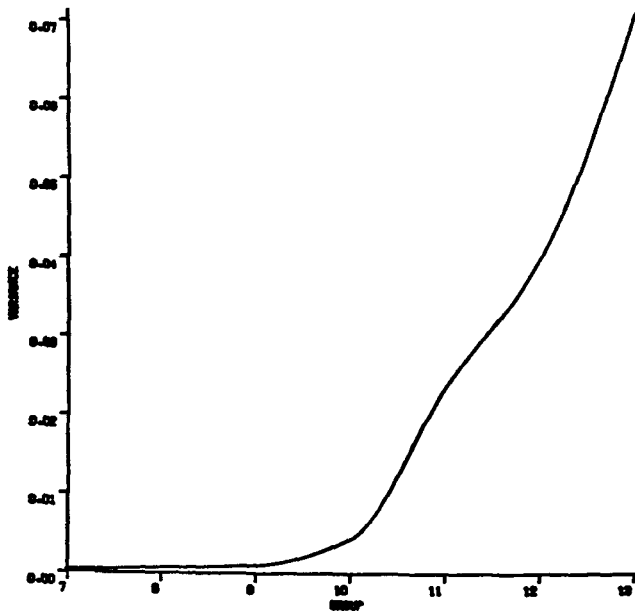


Figure 3. Change of variance with respect to group size
(Letters of complete fields 245 and 110)

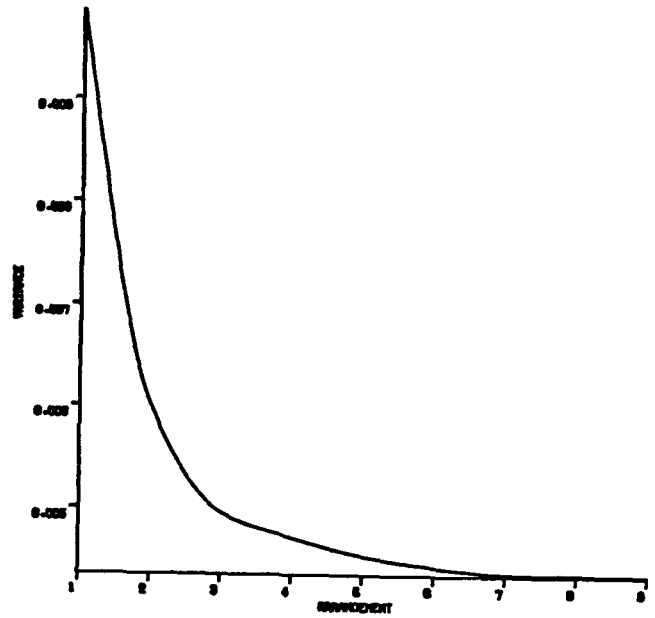


Figure 4. Variance as a function of arrangement improvement
(Step 1 = Worst, Step 9 = Best)

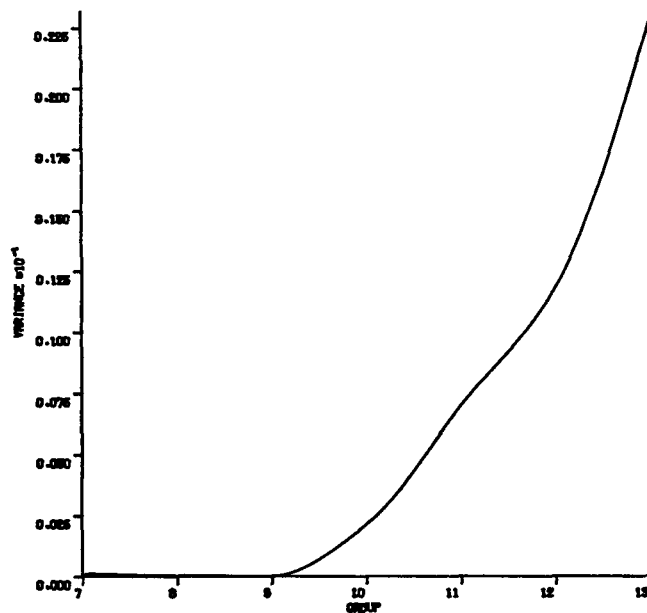


Figure 5. Change of variance with respect to group size
(Letters of field 245a)

8. REFERENCES

- Ayres, F. H., German, J., Loukes, N. and Searle, R. H. (1968). Author versus title: A comparative survey of the accuracy of the information which the user brings to the library catalogue. *J. of DOCUMENTATION*, 24(4), 266-272.
- Ayres, F. H. (1974). The Universal Standard Book Number (USBN): A new method for the construction of control numbers for bibliographical records. *PROGRAM*, 8(3), 166-173.
- Ayres, F. H. (1976). The Universal Standard Book Number (USBN): Why, how and a progress report. *PROGRAM*, 10(2), 75-80.
- Beale, I. J., and Lynch, M. F. (1975). An evaluation of, and improvement on Ayres's Universal Standard Book Number. *PROGRAM*, 9(2), 35-45.
- Nugent, W. R. and Vegh, A. (1962). Automatic word coding techniques for computer language processing. Rome Air Development Centre, RADC-TDR-62-13, Vol 1.
- Ryans, C. C. (1978). A study of errors found in non-MARC cataloguing in a machine-assisted system. *J. of LIBR. AUTO.*, 11(2), 125-132.
- Williams, M. E. and Maclaury, K. D. (1977). A state-wide union catalog feasibility study: Final report on project III-FY, Univ. of Illinois, Information Retrieval Research Laboratory.
- Yannakoudakis, E. J. (1979). Towards a universal record identification and retrieval scheme. *J. of INFORMATICS*, 3(1), 7-11.
- Yannakoudakis, E. J., Ayres, F. H. and Huggill, J. A. W. (1980). Character coding for bibliographical record control. *COMPUTER J.*, 23(1), 53-60.
- Yannakoudakis, E. J. and Wu, A. K. P. (1982). Quasi-equiprevalent group generation and evaluation. *COMPUTER J.*, 25(2), 183-187.