ERROR CONTROL CODING FOR A MULTI-SUBCARRIER H.F. MODEM

R M F Goodman and K Salloum

University of Hull, UK

SUMMARY

This paper investigates the performance of a number of block and convolutional coding schemes in terms of error rate and redundancy requirement when applied to a 48 subcarrier, 240 bits/sec, Kineplex type modem. We investigate both hard and soft-decision decoding schemes varying in complexity from simple data repetition to hybrid block/convolutional concatonated coding. Results of simulations using recorded error data are presented and discussed.

INTRODUCTION

There is no doubt that error-correction coding schemes have difficulties when applied to H.F. data transmission, as opposed to other more well behaved media. There are several reasons for this. Firstly, error-correcting codes yeild their most spectacular error-rate improvements at low channel error rates. As the error-rate on a high-speed H.F. modem can be 1 in 20 or worse, the <u>theoretical</u> advantages of coding imply improvements in error rate of factors of 2 to 3 as opposed to orders of magnitude at lower ($\approx 10^{-3}$) channel error rates. Secondly, the correction power of a code increases with length so that under Gaussian noise, and low error rate conditions, longer codes give better results. The high error rate/burst noise characteristics of the H.F. channel, however, dictate that short constraint (block) length codes should be used if the correction power per channel bit is not to be frequently overloaded. This implies that for a fixed redundancy penalty quite simple schemes can out-perform more complex ones. Thirdly, the modem characteristics are frequently ignored by the decoder resulting in a loss of information which could be used to advantage in the decoding scheme. For example, soft-decision information from the modem can asymptolically double the correction power of a code without further redundancy penalty, thereby significantly improving the correction power per bit problem. Also the modulation scheme of the modem usually results in specific types of error events. These are usually "interleaved out" by the error correction coding, instead of being used to advantage.

In this paper we investigate several coding schemes which range from simple to quite complex. In addition, we look at hard and soft-decision decoding, and at how the modem characteristics can be used by the error control decoder.

THE MODEM AND CHANNEL ERROR ERROR STATISTICS

The modem considered in this paper is a parallel transmission format or Kineplex type modem, operating at 2400 bits per second within a nominal transmission bandwidth of 3kHz. The data is tramsmitted in 48 bit parallel blocks or frames, using orthogonal multi-subcarrier phase shift keying at 50 frames/sec. Soft-decision information is also available from the modem, and 64 level (6 bit) quantization is used on each demodulated bit. (Ref. 1).

This type of modem-channel pair results in some very characteristic error events which can be used to advantage by the error control system. Some of these are as follows.

(a) Random errors.

(b) Errors which occur in the same place on successive frames due to stationary selective fading on one or more subcarriers within the band.

(c) Sweeping selective fades which traverse the band causing errors in repeated frames but in increasing (or decreasing) subcarrier positions.

(d) Flat fades which affect all subcarriers in several successive frames.

(e) Stationary jamming. This is an effect similar to stationary selective fading.

(f) Hopped jamming. In this case the jammer hops about the inband frequencies affecting different subcarriers pseudo-randomly. And furthermore, the time between each hop can be made to vary pseudo-randomly, thus disguising the jammer even more.

Some of these effects are shown in figures 1a to 1d. Each line represents a frame of data, consisting of 48 subcarriers across the band. Thus frequency runs horizontally and time vertically. An error in the demodulated data is represented by an asterisk. Figure 1a shows a small amount of random errors, with a bad selective fade in the region of two adjacent subcarriers which is causing errors in almost every successive frame. Figure 1b shows a bad flat and selective fading situation which is causing a large number of errors. Figure 1c shows a received sequence of bits in which two selective fades are sweeping across the band causing errors in successive frames but at increasing frequencies. Figure 1d shows a hopping jammer whose hop-time interval is approximately 6 frames. Thus different subcarriers are being completely blotted out pseudo-randomly.

Dr. R.M.F. Goodman and Mr. K. Salloum are with the Department of Electronic Engineering, University of Hull.

We now consider methods of mapping the users serial data stream onto this modem structure. This mapping determines the type of error patterns that the particular decoding scheme will have to handle. Consider first a serial mapping in which the data is simply packed into 48 bit frames and then transmitted sequentially by the modem. In this case, adjacent bits are transmitted on adjacent sub-carriers (except for every 48th bit which wraps-around to the lowest frequency again). We assume that the data is then unpacked at the receiver and fed to the decoder serially. In such a system the time separation of adjacent bits on the channel is zero (because they are in the same time frame, while their frequency separation is one (subcarrier). The effect of selective fading on the decoding input bit stream would be short bursts of errors with a marked period (48 bits)between bursts. A flat fade on the other hand would cause long dense bursts of errors, particularly if the fade lasted for several frames. It can be seen therefore that any decoding scheme used with such a system must have both bursts and random error-correction capability. However, it is unlikely that any decoder could cope with a bad flat fade lasting for several frames.

We therefore now consider interleaving schemes in order to improve the decoder's chances of success. Firstly, consider the situation in which successive frames are stored in an array at the encoder, but read out to the channel with a separation of Δ frames in time. In this case the effect of a flat fade which spreads over several frames is reduced, but successive bits in one frame can still result in a long burst of errors being input to the decoder. However, a burst error correction scheme may be able to cope with this, but could certainly not have coped with the burst running over several frames as in the case of no interleaving. In order to reduce the effect of a flat fade on the decoder, it is necessary to interleave successive bits in both time and frequency. This is performed by reading the encoded data stream into a 48 by Δ array where Δ is the interleaving depth in frames; and reading out the bits to the modem in a diagonal manner. The inverse operation of de-interleaving is performed at the receiver, before feeding bits to the error-correction decoder. Given a fixed parallel frame of 48 bits, there exists a trade-off between the time and frequency separation of adjacent bits in the de-interleaved stream. For example, if $\Delta = 48$, then bits are separated in time by 48 frames, but will have been transmitted on the same sub-carrier. Thus, stationary selective fades will cause long bursts of errors in the de-interleaved bit stream. In this paper we consider interleaving to a depth of 8 frames, which gives adjacent bits a separation of 8 bits in time and $\Delta/8=6$ bits in frequency. If the decoding algorithm is a random error-correcter only (such as the Viterbi algorithm) then such interleaving is essential if the decoder is not to be hopelessly overloaded with bursts.

The process of interleaving is one of randomising the modem channels' error characteristics. This is also a process of throwing information away.

It is, however, possible to use this information to the decoder's advantage. For example, if we assign a separate decoder to each sub-carrier we can monitor the 'difficulty' that the decoder is experiencing in decoding the bits on its particular sub-carrier. This difficulty measure can take the form of say monitoring the frequency of non-zero syndromes in the case of a block code, or the search effort involved in a sequential convolutional decoder. This difficulty measure gives us an idea of the error levels on each subcarrier in real time. The decoder can then use this information in the final decoding. In addition, this information can be processed to indicate the type of noise being experienced. For example, a stationary fade that affects one or more subcarriers could be spotted. A sweeping selective fade would result in subcarriers having decoding difficulty one after the other, and with a relatively fixed periodicity. Flat fades and interference would affect all decoders simultaneously. Note that it is not necessary to have 48 decoders. So long as we keep track of the subcarrier that each bit has been transmitted upon we can analyse the decoded data and extract the channel information, even if one decoder and interleaving has been used.

THE CODING SCHEMES

In this section we outline various coding schemes that we have tried on data recorded from the modem described in the last section. We have used both hard-decision and soft-decision decoding, and both block and convolutional coding schemes. Table 1 summarises the results.

Rate 2/3 Schemes

These schemes have a redundancy of 1/3, resulting in a user data rate of 2/3 x 2400 = 1600 bits/sec. The code used is a rate 2/3 systematic convolutional code, decoded with a minimum distance no (decoding algorithm (2,3), over a constraint length of 33 bits. The minimum distance over this length is 7 resulting in a (hard) correcting power of 3 bits over this length.

In a previous paper we have shown that soft-decision decoding symptotically doubles the correcting power of a code. We therefore expect the correction power to be upper bounded by 6 bits, when the modem's soft-decision capability is used by the decoder. The scheme was applied to the channel data in both a direct sequential bit-by-bit manner, and via an interleaver which gave adjacent bits a separation of 8 bits in time and 6 in frequency.

Results are shown for both hard and soft-decision decoding. The soft-decision decoding has been done in several ways which trade off performance and decoder search effort. The "true" soft-decision results are the full search results. Partial search indicates that the decoder stops whenever the first better path is found (this may not be the best). Finally, pre-decoding indicates a hard-decision decoder which chooses the next path segment(s) via soft-decision instead of hard-decision. This is the fastest form of soft-decision decoding.

Rate ½ Schemes

These schemes have a redundancy of ½ resulting in a user data rate of 1200 bits per second. They are as follows.

(a) Soft Decision Data Duplication. In this scheme the data is simply sent twice, sequentially. At the decoder, if the two bits disagree the bit with the lower soft-error (5) is taken as the decision.

(b) A half-rate burst and random-error-correcting diffuse code (6) with a constraint length of 58 bits, which can correct any 2 random errors, or any bursts of length 20 bits within the constraint length. The decoding scheme used is threshold decoding (7) for which we have developed a soft-decision version (8).

(c) The (23, 12) perfect Goley code. The hard decoder uses simple look-up decoding. The soft-decision decoder uses permutation decoding (9).

Rate 1/3 Schemes

(a) Data triplication. The hard version of this scheme involves simply taking a majority vote on the three repeated incoming bits. Such a scheme is capable of correcting one error in a block of 3. The soft-decision version (as in the ½ rate scheme) involves choosing the codeword (000 or 111) which results in the smallest 'soft' error. Assuming 8-level quantisation, hard 000 and 111 are separated by 24 soft levels. Thus an error pattern of 11 levels can be corrected which implies a 'hard' correcting power of Int (11/4) = 2. This makes simple data triplication a very powerful scheme.

(b) One-third rate Convolutional Code with Minimum Distance Decoding. This scheme employs either hard or soft-decision decoding on a convolutional code with constraint length 21 bits. The hard correction power over this length is 4 which implies an asymptotic soft power of 8. The scheme has been applied to the modem structure in two ways. Firstly, in one-decoder mode, with and without interleaving. Secondly, 48 decoder mode, in which each decoder deals with the bits on one sub-carrier only.

Hybrid Schemes

We have investigated several hybrid block and convolutional coding schemes including block/convolutional concatonated and block/convolutional product. Some of these are outlined below.

(a) Block/Convolutional Concatonated.

In this scheme data is first encoded into an outer (48, 42) single error correcting Hamming code. This code system is then encoded with a 1/3 rate convolutional inner code. At the receiver the 1/3 rate code is maximum-likelihood decoded first, and then Hamming decoded.

(b) Block/Convolutional Product

In this scheme each sub-carrier is allocated to an individual rate 1/3 convolutional decoder. A (48,42) single error correcting Hamming code is then used across the frame giving an overall rate of 0.3.

In addition to the above schemes we are in the process of trying other hybrid schemes and hope to report further on these at the conference.

CONCLUSIONS

The results presented in table 1 lead us to several main conclusions. Firstly, the use of soft-decision decoding dramatically improves the error rate on all channels, compared with hard-decision decoding. We would expect this as correction power per bit is being effectively doubled by this technique. Secondly, we see that interleaving also improves the error rate significantly. It seems essential to stop the decoder being hit 'hard' by bursts, and interleaving accomplishes this. Thirdly, it seems necessary to go to coding schemes of about 1/3 rate, resulting in a user data rate of 800 bits/sec, in order to get really low output error rates. Finally, and perhaps most surprisingly, we see that simple schemes such as data replication often perform better than more complex schemes. This is undoubtedly because of their high correction power over a short block length.

ACKNOWLEDGEMENTS

The authors would like to thank John Gordon and Nigel Montague of the Hatfield Polytechnic for their assistance.

REFERENCES

1. Montague, N., 1977, "An Application of Error Protection to a Vocoded Speech Channel", I.E.R.E. Conf. Proc., No.37.

2. Ng, W.H., and Goodman, R.M.F., 1978, "An Efficient Minimum-Distance Decoding Algorithm for Convolutional Error Correcting Codes", Proc. IEE, Vol.125, No.2.

3. Goodman, R.M.F., and Winfield, A.F.T., 1981, "Soft-Decision Minimum-Distance Sequential Decoding for Convolutional Codes", Proc. IEE, Vol. 128, pt.F, No.3.

4. Farrell, P.G., and Goodman, R.M.F., 1980 "Soft-Decision Error Control for H.F. Data Transmission", Proc. IEE, Vol. 127, Pt.F, No.5,

5. Goodman, R.M.F. and Green, A.D. and Winfield, A.F.T., 1979, "Soft-Decision Error-Correction Coding Schemes for H.F. Data Transmission" Proc. IEE, Conf. on Advances in H.F. Communications, London.

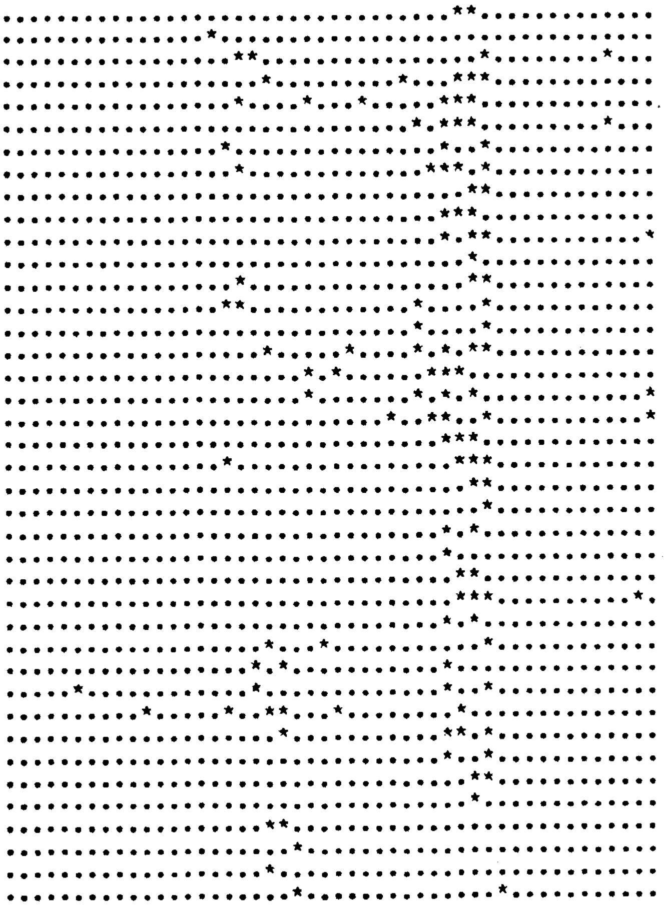
6. Brayer, K., 1971, "Error Correction Code Performance H.F., Troposcatter, and Satellite Channels" IEEE Trans. Com.19,

7. Massey, J.L., 1963 "Threshold Decoding", M.I.T. Press, Cambridge Mass.

8. Goodman, R.M.F., 1979 "Soft-Decision Threshold Decoders", Algebraic Coding Theory and Applications, Ed. G. Longo, Springer-Verleg, Geneva.

9. Goodman, R.M.F., and Green, A.D., 1978 "Microprocessor Controlled Permutation Decoding of Block Error-Correcting Codes", IERE Conf. Proc., No. 41.

TABLE 1	FILE 1		FILE 2		FILE 3	
Channel Characteristics:	STATIONARY SELECTIVE FADES		BAD FLAT FADES		SWEEPING SELECTIVE FADES	
	NON-INT	INTER.	NON-INT	INTER.	NON-INT	INTER.
ERROR RATE:	1 in 22		1 in 11		1 in 16	
ERRORS:	964		1951		1350	
Rate 2/3 = 1600 Bits/sec	NO OF OUTPUT ERRORS IN THE DATA					
H.D. Convolutional	528	519	665	658	573	
S.D. Conv. Part. Search	94		163		37	
S.D. Conv. Full. Search	5					
H.D. with 1 Seg. Pre-Dec	61	23	117		16	
H.D. with 2 Seg. Pre-Dec	15	3	67		3	
Rate ½ = 1200 Bits/sec						
S.D. Data Duplication	134	82	298		45	
H.D. Diffuse Conv.	104	131	749	797	211	246
S.D. Diffuse Conv.	65	59	484	506	69	73
H.D. Golay Block	151	76	652	656	170	158
S.D. Golay Block	98	21	436	410	59	29
Rate 1/3 = 800 Bits/sec						
H.D. Data Triplication	109	38	261	293	177	47
S.D. Data Triplication	50	10	144	133	28	1
H.D. Conv. 1 Decoder	52	8	262	248	66	20
S.D. Conv. 1 Decoder	27	3	231	243	39	0
H.D. Conv. 48 Decoders	111	48	306	281	251	
S.D. Conv. 48 Decoders	17		89		6	
Rate .29 = 700 Bits/sec						
H.D. Block/Conv Concat.	28	30	306	308	112	86
S.D. Block/Conv Concat.						
H.D. Block/Conv Prod.	69	8	291	268	228	
S.D. Block/Conv Prod.						



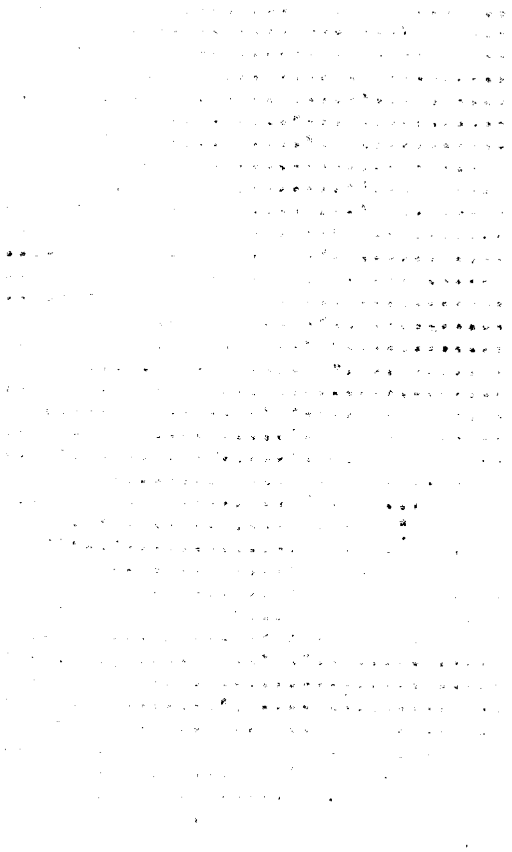
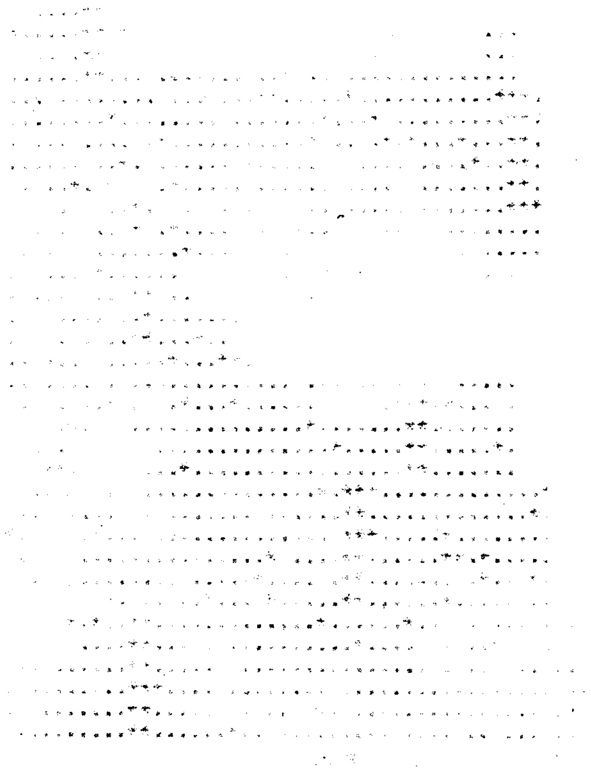


Fig. 1a



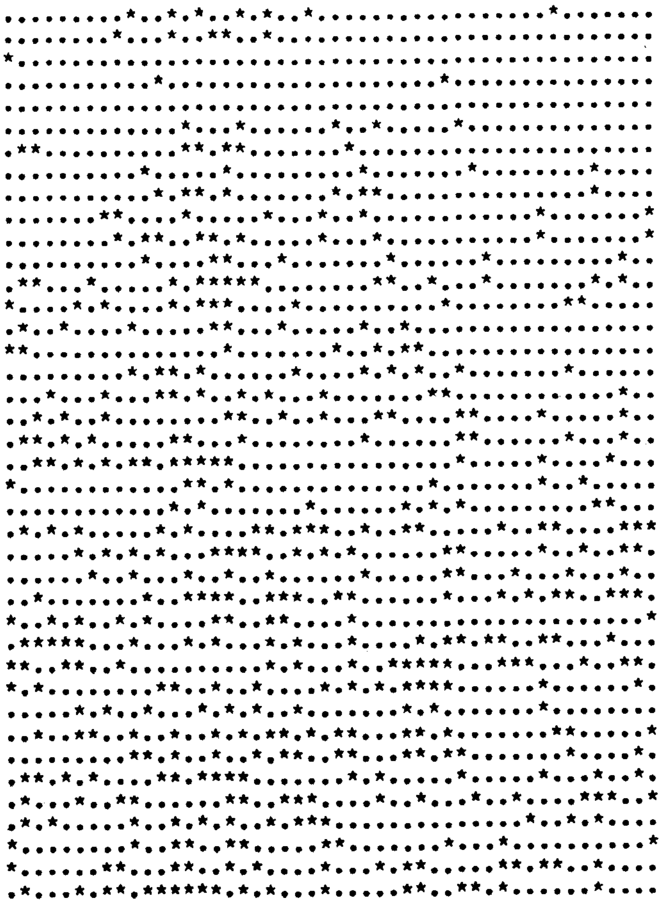
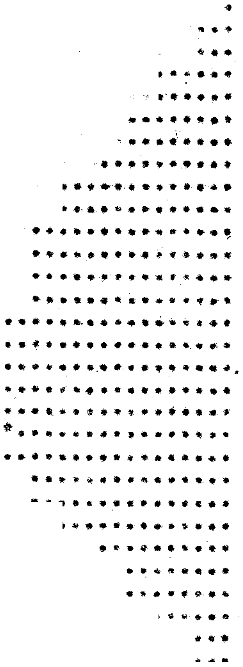


Fig. 1b



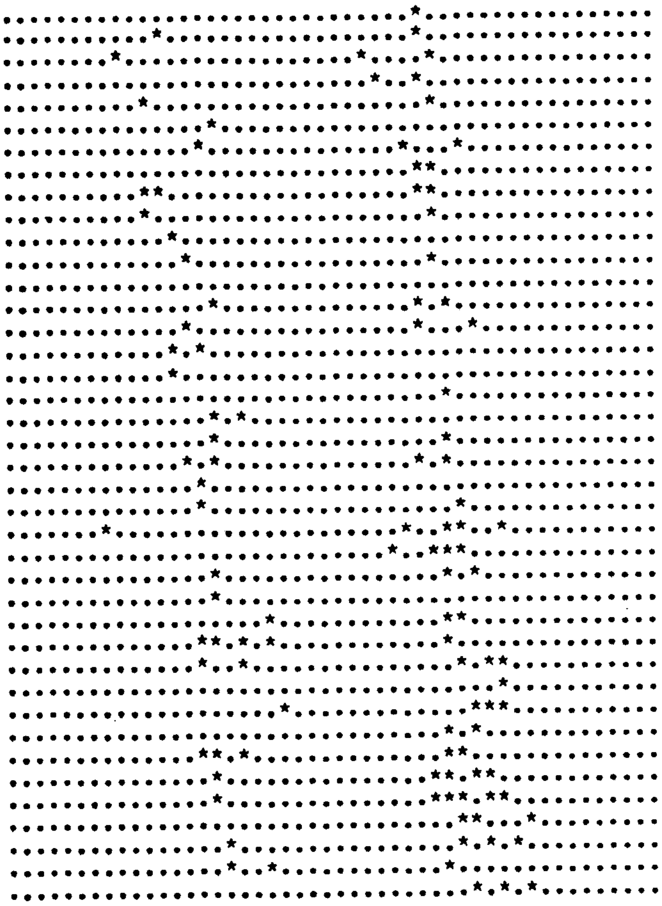


Fig. 1c

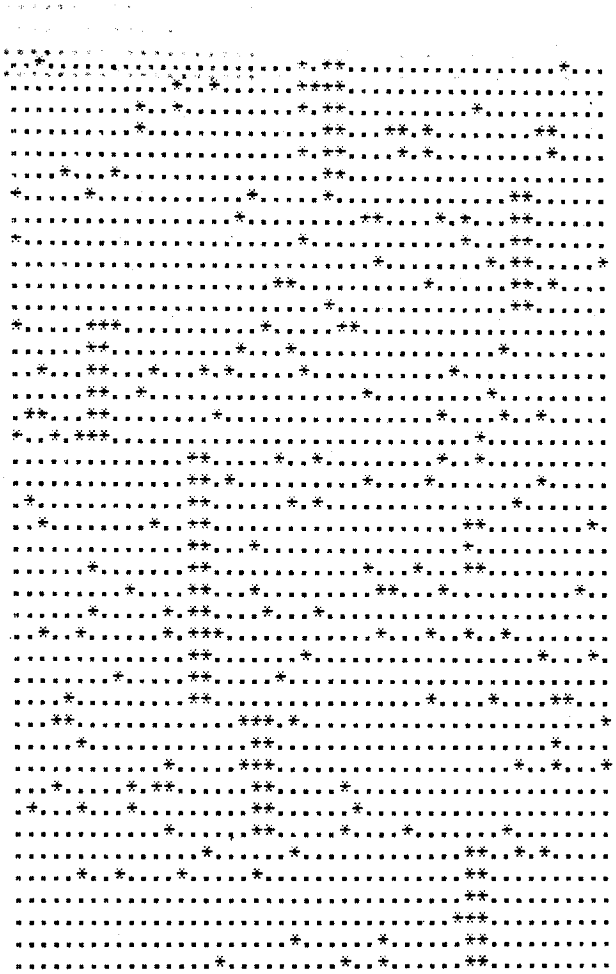


Fig. 1d