The 'on-ground' compression of satellite data

L. F. Turner

Electrical Engineering Department, Imperial College of Science and Technology, Exhibition Road, London SW7 2BT

This paper deals with aspects of the 'on-ground' compression of satellite data. The Shannon-Fano, Run-Length and Hadamard Transformation methods of data compression are considered and the results obtained when applying the techniques to the compression of ESRO 1 satellite data are presented and compared.

(Received December 1973)

1. Introduction

In many space research programs the information obtained by satellites is transmitted back to earth where it is then stored for subsequent processing and use. During the last decade, spacecraft data rates have been relatively low and, it has been possible to process the received data, store them and, at some later time, to recover them from store without too much difficulty. More recently, however, the trends have been towards developing spacecraft with data rates of the order of 100K bits/sec. and this has meant that problems have started to arise with respect to the amount of transmission channel bandwidth that is necessary and the amount of 'on ground' storage that has to be provided if the received data are to be preserved for subsequent analysis.

If the data could be compressed in the spacecraft, that is, if the redundancy could be removed by equipment placed in the spacecraft, then the channel bandwidth and on-ground storage requirements would both be eased. At present, however, the complexity and weight of the hardware necessary to provide worthwhile redundancy reduction tends to limit the 'on-board' compression and this means that the problem of reducing the data storage requirements has to be considered in terms of the on-ground compression of the received data.

This paper is devoted to a consideration of various techniques for removing redundancy from the received data. The techniques have been implemented using a general purpose computer, and associated software compression packages have been developed. The compression techniques are designed to take advantage of the statistical regularity and predictability of the data. By applying the techniques to the incoming data it is possible to remove the non-information bearing elements from the received signals and leave behind, for storage, only those elements that actually carry information. In this way the 'on-ground' storage requirements for satellite data can be reduced. In the sections which follow, the Shannon–Fano, a modified Shannon–Fano, Run-Length and Hadamard Transformation methods of data compression are considered and the results obtained when applying them to ESRO 1 satellite data are given. It is shown that it is possible to reduce the necessary data storage capacity by factors of up to 10:1, with some slight loss of accuracy. The amount of reduction that can be achieved depends on the statistics of the data and the fidelity with which it is desired to reconstruct the actual input data from the stored compressed data.

2. Data compression techniques

2.1. *Introduction*

The rigorous scientific foundation for bandwidth and data compression is to be found in Shannon's classic work on the mathematical theory of communication (Shannon, 1948). Shannon's work can be divided into two quite distinct parts. The first part deals with measures of information and with the efficient encoding of the output of a data source. The second part, which is not important in the context of the present paper, deals with the problem of the reliable transmission of information over noisy communication channels, and gives rise to specialised subjects such as 'error correcting codes'.

In his work on information measures and source encoding, Shannon developed a measure of information called the 'entropy function' and, in his 'noiseless coding theorem', he gave the entropy function direct physical meaning when he proved that it is possible, without any loss of fidelity, to either transmit or store the output of a data source using an average number of binary digits that can be made as close to the entropy function as desired. Shannon further proved that it is not possible to either transmit, or store, the output of a source with complete fidelity when using an average number of binary digits which is less than the value of the entropy function. The branch of information theory called 'Rate Distortion Theory' (Berger, 1971) deals quantitatively with the question of how many binary digits have to be used in order to transmit or store the output of a data source to within a desired degree of fidelity.

To encode a data source, and to do so using an average number of binary digits that approaches the minimum number specified by the entropy function, it is necessary, in general, to employ an increasingly complex encoder and decoder. In general, the average number of binary digits used approaches the entropy function as a limiting process and only in certain rather special cases can the source be encoded using an average number of binary digits equal to the entropy function without using a complex encoder/decoder. The study of data compression is really a study of how to encode a data source so as to reduce the average number of digits used in encoding to a value that approaches the value of the entropy function, and how to do this with fairly simple encoding/decoding equipment and procedures.

In this paper a number of methods of data compression are considered. The results of applying the methods to ESRO 1 satellite data are given and the efficiency of the compression is compared with that known to be possible from the zero-order entropy of the source. As mentioned previously, the processes of data compression and decompression were carried out using a general purpose computer and associated software packages.

2.2. *Shannon–Fano, Run-Length and Hadamard transformation methods of data compression*

2.2.1. *Compression by Shannon–Fano encoding*

Two fairly simple methods exist for the encoding of finite-length sequences of source symbols. The first method is due to Shannon and Fano (1948) and the second method is due to Huffman (1952).

The Huffman encoding scheme is optimum in the sense that no other encoding scheme uses a smaller average number of code symbols and it is, therefore, generally slightly superior to the Shannon–Fano scheme. However, it is found in practice that, more often than not, the two encoding schemes result in code-word lengths* that are identical, and, when differences do occur, they involve the low-probability code-words and have little or no effect on the overall compression ratio that can be obtained. Also, the Huffman encoding procedure is slightly more difficult to structure and to program than the Shannon–Fano scheme and, therefore, since the Shannon–Fano encoding scheme is near optimum it was decided to use it rather than the Huffman scheme in the investigation of the compression of satellite data.

The basic idea underlying the Shannon–Fano method is that the most frequently occurring source signals are encoded using short-length binary code words and the less frequently occurring signals are encoded using longer binary code words. A similar idea forms the basis of the familiar Morse code.

The Shannon–Fano encoding technique was applied to ESRO 1 data. To do this software packages were written which enabled the following operations to be performed:

1. The data were examined and an estimate, P_i, formed of the probability that the data source would generate as signal output the amplitude i. In the case of ESRO 1 data the output was in the form of 8-bit PCM words, and hence it was possible for i to have, at any one time, an integer value in the range 0 to 255.
2. The zero-order entropy, $H = \sum_{i=0}^{i=255} P_i \log_2 (1/P_i)$, was computed.
3. The Shannon–Fano encoding book was constructed using the information obtained in operation (1), and the data were encoded and the compression ratio computed.
4. The encoded data were decoded to obtain the original data, and the decoding time was evaluated.

The results obtained from applying the Shannon–Fano technique to a number of ESRO 1 data channels are given in **Table 1.** The compression ratio, which is defined as:

$$CR = \frac{\text{Number of binary digits in input sequence}}{\text{Number of binary digits in output sequence}},$$

varies from a maximum of 1·81:1 to a minimum of 1·44:1, with an average compression ratio of 1·60:1. The compression ratios obtained are seen to be close to the maximum ratios that the zero-order entropies indicate as possible. This closeness indicates that the Shannon–Fano technique is highly efficient as a means of encoding 8-bit outputs, rather than that the maximum possible compression has been achieved. The zero-order entropy indicates only the maximum compression that can be achieved when successive outputs are statistically independent. In the case of ESRO 1 data, successive outputs were found not to be statistically independent and higher compression ratios are thus possible. One method of increasing the degree to which the data could be compressed would be to use the Shannon–Fano technique to encode groups of source signals rather than to encode individual source signals. The difficulty with this technique is that the number of code words in the encoding book increases exponentially as the number of source signals encoded at any one time, and this thus places excessive demands on the available core storage. With ESRO 1 data, for example, the source can generate any one of 256 different possible outputs, and hence to encode pairs of outputs it would be necessary to have an encoding book consisting of $256^2 = 65{,}536$ binary code words.

Table 1 Shannon–Fano encoding applied to individual data samples

	Zero order entropy (bits/ 8-bit sample)	*Compression ratio*	*Compression based on zero-order entropy*
MIN	4·34	1·44:1	1·46:1
MAX	5·46	1·81:1	1·83:1
AVERAGE	5·07	1·60:1	1·58:1

Table 2 Shannon–Fano encoding applied to difference between adjacent data samples

	Zero-order entropy of differences (bits/8-bit sample	*Compression ratio*	*Compression ratio on zero-order entropy*
MIN	3·22	2·04:1	2·08:1
MAX	3·84	2·46:1	2·48:1
AVERAGE	3·52	2·24:1	2·28:1

In an attempt to take advantage of the statistical dependence between adjacent data samples, and to overcome the sensitivity of the Shannon–Fano technique to changes in data statistics† a modified application of the Shannon–Fano technique was investigated. This modification consisted of applying the technique to differences between the adjacent data samples. The results obtained when using the modification are shown in **Table 2.** From the table it is clear that the modification results in an increase in the compression that can be achieved.

2.2.2. *Compression by Run-Length encoding*

Run-length encoding has been used successfully as a means of reducing the bandwidth necessary for the transmission of television signals (Cherry *et al.*, 1967).

In this section a number of modified versions of the basic run-length encoding techniques are considered and the results obtained when using them to compress satellite data are given.

The basic idea of run-length encoding is that if a number of adjacent data samples have the same value (amplitude) then they can be considered to constitute a run, and the information contained in the run can be either transmitted or stored using only the amplitude of the first sample, and the length of the run. It is not necessary to transmit or store all samples in the run.

Modifications to the basic run-length encoding technique are associated mainly with the way in which the 'length-of-run' information is encoded. When studying the compression of ESRO 1 data three methods of encoding the length of run information were investigated. In the first method the lengths of all naturally occurring runs were encoded using a Shannon–Fano code, in the second method naturally occurring runs were broken down into sequences of permissible sub-runs and

*With Shannon-Fano and Huffman compression schemes a source symbol, or, in more complex encoding, a group of source symbols are encoded into and are therefore represented by a sequence of binary digits called a 'code-word'. The number of binary digits in the code word is referred to as the 'code-word length' and the complete set of binary code words used to represent the source symbols is termed 'the encoding book'.

†It can be seen (Fano, 1961) that the structure of the encoding book depends on the probability with which the various source signals occur. It thus follows that if the probabilities change then the encoding book should be changed. Failure to do so may result in data expansion rather than compression!

Table 3 Natural runs divided into permissible sub-runs

Length of run	*Sequence of permissible sub-runs used to represent run*
1	1
3	2, 1
8	5, 2, 1
30	9, 9, 9, 2, 1
100	9, 9, 9, 9, 9, 9, 9, 9, 9, 9, 9, 1

NB Runs of lengths 1, 2, 5 and 9 were adopted as permissible runs

Table 4 Compression by run-length encoding with zero aperture

	Run-length encoding					
	Method 1		*Method 2*		*Method 3*	
	Comp. ratio	*Assoc. RMS error*	*Comp. ratio*	*Assoc. RMS error*	*Comp. ratio*	*Assoc. RMS error*
MIN	1·12	0	1·07	0	0·987	0
MAX	1·71	0	1·53	0	1·26	0
AVERAGE	1·32	0	1·23	0	1·09	0

the lengths of the sub-runs were encoded using a Shannon-Fano code. In the third method the naturally occurring runs were again broken down into sub-runs, and the lengths of the sub-runs were encoded using a set of fixed-length 2-bit code words.

The way in which naturally occurring runs were broken down into sequences of permissible sub-runs is shown in **Table 3.** The set of permissible run lengths of 1, 2, 5 and 9 were chosen in an arbitrary manner since, at present, the answers to the questions of: 'What constitutes an optimum set of fixed run lengths?' and 'How should the set be determined from and how is it related to the probability distribution governing the naturally occurring run lengths?', are not known. In a general sense these are quite fundamental questions that should be examined further since in practice the number of naturally occurring runs will be large, and, therefore, run-length encoding using naturally occurring runs will require large encoding and decoding books which tend to be prohibitively complex. On account of this code-book complexity it is highly likely that any hardware implementation of run-length encoding will involve the use of a limited set of permissible run-lengths, and it is therefore necessary to know how to determine the set.

The results obtained when applying the three versions of run-length encoding to compress ESRO 1 data are shown in **Table 4.** From Table 4 it can be seen that the first method is the most efficient, and that the second method is slightly better than the third method. If the results of Table 4 are compared with those of Tables 1 and 2 it will be noted that the compression that can be achieved by run-length encoding appears to be somewhat disappointing. The reason for the poorness of the result is that the ESRO 1 data contained very few long runs and consisted mainly of runs of length equal to one. The data used in the study were specially selected by ESOC to provide a stringent test of the compression techniques investigated and analysis revealed that between 69 per cent and 79 per cent of all runs were of length one.

In an attempt to increase the length of the runs, and thereby increase the extent to which the data could be compressed, the technique of 'finite aperture sampling' was used. With finite aperture sampling, samples are considered to be the same if they differ by less than a predetermined amount. In the tests on the ESRO 1 data the two sizes of aperture, ± 1 PCM quantisation unit and ± 7 PCM quantisation units, were used*. These apertures correspond respectively to $\pm 1/256 = 0{\cdot}4$ per cent and $\pm 7/256 = 2{\cdot}7$ per cent of peak signal value; and it follows that, when using an aperture of size ± 1 PCM unit, the maximum error that can occur between the original and the reconstructed data is 0·4 per cent of the peak signal value. When sampling with an aperture of ± 7 PCM units this maximum error increases to 2·7 per cent.

The extent to which the satellite data could be compressed when using run-length encoding with finite aperture sampling is shown in **Table 5** and **Table 6.**

On consideration of Tables 5 and 6 a number of important points are immediately apparent. The first point to note is that Method 1 is clearly more efficient than either of the two other methods. This is to be expected since with Methods 2 and 3 a restriction is placed on the maximum permissible run-length. A second point to note is that, as regards compression efficiency, there is little difference between Methods 2 and 3. As regards practical implementation, Method 1 is more difficult to implement than Methods 2 and 3, and Method 3 is easier to implement than Method 2. The difficulty associated with Method 1 stems from the fact that the number of different

Table 5 Compression by run-length encoding with a ± 1 PCM quantisation unit aperture

	Run-length encoding					
	Method 1		*Method 2*		*Method 3*	
	Comp. ratio	*Assoc. RMS error (pcm units)*	*Comp. ratio*	*Assoc. RMS error (pcm units)*	*Comp. ratio*	*Assoc. RMS error (pcm units)*
MIN	1·74	0·57	1·58	0·87	1·52	0·87
MAX	2·42	0·48	2·07	0·65	2·00	0·65
AVERAGE	2·04	0·56	1·80	0·78	1·72	0·78

Table 6 Compression by run-length encoding with a ± 7 PCM quantisation unit aperture

	Run-length encoding					
	Method 1		*Method 2*		*Method 3*	
	Comp. ratio	*Assoc. RMS error (pcm units)*	*Comp. ratio*	*Assoc. RMS error (pcm units)*	*Comp. ratio*	*Assoc. RMS error (pcm units)*
MIN	3·26	2·65	2·65	2·87	2·60	2·87
MAX	9·90	3·00	4·94	2·76	4·77	2·76
AVERAGE	5·66	2·73	3·52	2·82	3·45	2·82

*The apertures were obtained by discarding the appropriate binary-digit differences in the 8-bit representation of data samples.

naturally occurring runs may be very large and hence the associated Shannon–Fano code book (which has to be stored) may also be very large. With Methods 2 and 3 the size of the code book is determined by the number of different run-lengths that are permitted. Method 3 is simpler than Method 2 because the encoding and decoding procedures are easier to perform with fixed-length than with variable-length code words.

From Tables 5 and 6 it can be seen that average compression ratios of about 3·5:1 are possible with Methods 2 and 3 and that with Method 1 the average compression ratio is approximately 5:1. Although these results are not startling, they are encouraging, especially when it is realised that the data used in the study were of a much more fluctuating nature than that likely to be encountered in general.

2.2.3. *Compression by Hadamard transformation encoding*

Fairly recently (Pratt *et al.*, 1969) the suggestion has been made that signal transformation techniques such as the Fourier and Hadamard transformations may be useful as a means of reducing the bandwidth necessary for the transmission of television pictures. In this section the application of the Hadamard transformation to data compression is described and the compression results obtained when applying the technique to ESRO 1 data are presented.

The basic idea underlying compression by the Hadamard transformation technique is that by the process of transformation some of the redundancy may be removed from the signal, and, therefore, if the signal is stored in the form that it has in the transform domain less storage capacity may be necessary.

A more detailed understanding of the functioning of the Hadamard transformation, and of the reasons why it may be used to achieve data compression, can be gained from an examination of the process and from consideration of a simple example. In applying the Hadamard transformation technique the data are broken down into blocks of N^2 samples and, for each block, the samples which are denoted by $x_1, x_2, \ldots, x_{N^2}$ are arranged as an $N \times N$ data matrix

$$[D(X)] = \begin{bmatrix} x_1 \ldots x_{N^2-N+1} \\ \vdots \quad\quad \vdots \\ x_N \quad\quad x_{N^2} \end{bmatrix} .$$

The Hadamard transform matrix, $G(U)$ is then obtained using the matrix equation

$$[G(U)] = [H] \cdot [D(X)] \cdot [H] \tag{1}$$

where $[H]$ is an N by N Hadamard matrix and $[D(X)]$ is the data matrix.

As an example, suppose four data samples x_1, x_2, x_3 and x_4 are arranged as 2×2 matrix

$$[D(X)] = \begin{bmatrix} x_1 & x_3 \\ x_2 & x_4 \end{bmatrix} ,$$

and the Hadamard transformation is taken according to Equation 1 by using the 2×2 Hadamard matrix $\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$.

If this is done the Hadamard transformation is found to be

$$[G(U)] = \begin{bmatrix} u_1 & u_3 \\ u_2 & u_4 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} x_1 & x_3 \\ x_2 & x_4 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} =$$
$$\begin{bmatrix} (x_1 + x_2 + x_3 + x_4) & (x_1 + x_2 - x_3 - x_4) \\ (x_1 - x_2 + x_3 - x_4) & (x_1 - x_2 - x_3 + x_4) \end{bmatrix} .$$

It will be noted that the element u_1 is comprised of the sum of the data samples and that the elements u_2, u_3, and u_4 are made up of sums and the differences of the data samples. It thus follows that if the data are slowly varying, that is, samples do not change appreciably over the set, then the element u_1 will be large and all other elements will be relatively small. It is this fact which makes data compression possible. If the data samples are slowly varying over the set then compression can be achieved if only the large elements of the transform matrix $[G(U)]$ are stored, and all other elements are set to zero or some small value.

An alternative interpretation of the Hadamard transformation method of data compression can be given in terms of an orthogonal function representation of data sequences. It is possible to show (Turner, 1973) that the transformation of Equation 1 yields exactly those coefficients that are obtained when the data sequence $x_1, \ldots, x_{N^2}$ is expanded in terms of N^2 Walsh functions. This can be seen more clearly by considering again the example of above. If the normalised Walsh functions $\mathrm{Wal}(a_1) = 1/4(1, 1, 1, 1)$; $\mathrm{Wal}(a_2) = 1/4(1, -1, 1, -1)$; $\mathrm{Wal}(a_3) = 1/4(1, 1, -1, -1)$; $\mathrm{Wal}(a_4) = 1/4(1, -1, -1, 1)$ are used as orthogonal basis vectors and the data sequence x_1, x_2, x_3, x_4 is expanded as a sum of these Walsh functions, that is,

$$(x_1, x_2, x_3, x_4) = a_1 \,\mathrm{Wal}(a_1) + a_2 \,\mathrm{Wal}(a_2)) + a_3 \,\mathrm{Wal}(a_3) + a_4 \,\mathrm{Wal}(a_4) ,$$

then it is found that the coefficients, which are obtained by making use of the orthogonality properties of Walsh functions, are

$$a_1 = (x_1 + x_2 + x_3 + x_4);$$
$$a_2 = (x_1 - x_2 + x_3 - x_4);$$
$$a_3 = (x_1 + x_2 - x_3 - x_4)$$

and

$$a_4 = (x_1 - x_2 - x_3 + x_4) .$$

If the data are slowly varying then $\mathrm{Wal}(a_1)$ will be the major contributing term in the expansion representing the data sequence, and the associated coefficient, a_1, will be large and all other coefficients relatively small. It will be noted that the coefficients a_1, a_2, a_3 and a_4 are exactly the same coefficients that were obtained as elements of the transform matrix $[G(U)]$.

If $[\widetilde{G(U)}]$ is used to denote the stored compressed data matrix then an estimate, $[\widetilde{D(X)}]$, of the original data can be obtained from the stored matrix $[\widetilde{G(U)}]$ by using the equation

$$[\widetilde{D(X)}] = \frac{1}{N^2} [H][\widetilde{G(U)}][H] \tag{2}$$

It should be noted that if $[\widetilde{G(U)}] = [G(U)]$ then $[\widetilde{D(X)}] = [D(X)]$, that is, the data can be recovered without any loss of fidelity.

In applying the Hadamard transformation technique to ESRO 1, data, blocks of 16 data samples were used and a

Table 7 Methods of encoding the transform matrix $[G(U)]$

Matrix compression method no.	*Encoding scheme*
1	12 bits to encode element a 3 bits to encode each of elements $b, \ldots, p$.
2	12 bits to encode element a 3 bits to encode each of elements b, f, e 2 bits to encode each other element.
3	12 bits to encode element a 2 bits to encode each of elements $b, \ldots, p$.
4	12 bits to encode element a 3 bits to encode each element b, f, e Neglect other elements.

Table 8 Compression ratios and associated RMS errors obtained with Hadamard transformation encoding

	1		*2*	
	Comp. ratio	*RMS error (pcm units)*	*Comp. ratio*	*RMS error (pcm units)*
MIN	2·25	6·50	2·85	6·67
MAX	2·25	23·50	2·85	23·60
AVERAGE	2·25	15·27	2·85	15·42
	3		*4*	
	Comp. ratio	*RMS error (pcm units)*	*Comp. ratio*	*RMS error (pcm units)*
MIN	3·05	6·71	6·10	6·77
MAX	3·05	23·70	6·10	23·70
AVERAGE	3·05	15·49	6·10	15·42

number of different methods of compressing the transform matrix,

$$G(U, V) = \begin{bmatrix} a & b & c & d \\ e & f & g & h \\ i & j & k & l \\ m & n & o & p \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_1 & x_2 & x_3 & x_4 \\ x_5 & x_6 & x_7 & x_8 \\ x_9 & x_{10} & x_{11} & x_{12} \\ x_{13} & x_{14} & x_{15} & x_{16} \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}$$

were examined. The methods used are shown in **Table 7** and the overall data compression obtained from the application of these various matrix compression schemes are shown in **Table 8.**

If the results of Table 8 are compared with those of Table 6 it will be seen that Matrix Compression Methods 1, 2 and 3 give overall data compression ratios that are not significantly different from those obtained by run-length encoding, but that the RMS errors are much worse than those resulting from run-length encoding. In the case of Matrix Compression Method 4, the overall data compression ratios are generally higher than those obtainable by run-length encoding. At present, tests are being carried out using larger apertures to determine the data compression ratios that are obtainable with run-length encoding for RMS errors similar to those that arise with Hadmard transformation encoding and matrix compression method 4.

3. Conclusions

In this paper, various techniques of data compression have been considered, and the results obtained when applying them to ESRO 1 satellite data are given.

An inspection of the Tables shows that, in general, run-length encoding tends to give the best compression results, particularly when apertures are employed. If apertures cannot be used then the modified Shannon–Fano technique gives the best compression.

Acknowledgements

The work reported on in this paper was supported by the European Space Operations Centre (ESOC) and was carried out for Oxford Computer Services (OCS). The author gratefully acknowledges the permission of ESOC and OCS to publish this paper.

References

SHANNON, C. E. (1948). A Mathematical Theory of Communication, *BSTJ*, Vol. 27, pp. 379-423 (Pt. 1), pp. 623-656 (Pt. 2).
BERGER, T. (1971). *Rate Distortion Theory*, Prentice Hall Book Co. Ltd.
HUFFMAN, D. A. (1952). A Method for the Construction of Minimum Redundancy Codes, *Proc. IRE*, Vol. 40, pp. 1098-1101.
FANO, R. M. (1961). *Transmission of Information*, Wiley Book Co. Ltd., and MIT Press.
CHERRY, E. C., *et al.* (1967). Results of a Prototype Television Bandwidth Compression Scheme, *Proc. IEEE*, Vol. 55, No. 3, pp. 356-364.
PRATT, W. K. *et al.* (1969). Hadamard Transform Image Coding, *Proc. IEEE*, Vol. 57, No. 1, pp. 58-68.
TURNER, L. F. (1973). *Some Properties of the Hadamard Transformation and its Relation to Data Compression*, Internal Report, Dept. of Elec. Eng., Imperial College.