A note on the free distance for convolutional codes

A note on the free distance for convolutional codes.

by

A. J. Vinck,
J. G. van Wijk, and
A. J. P. de Paepe.

TECHNISCHE HOGESCHOOL EINDHOVEN
NEDERLAND
AFDELING DER ELEKTROTECHNIEK
VAKGROEP INFORMATIE- EN COMMUNICATIETHEORIE EI

EINDHOVEN UNIVERSITY OF TECHNOLOGY
THE NETHERLANDS
DEPARTMENT OF ELECTRICAL ENGINEERING
GROUP INFORMATION- AND COMMUNICATION THEORY EI

A note on the free distance for convolutional codes.

by

A.J. Vinck,
J.G. van Wijk, and
A.J.P. de Paepe.

This work was partly supported by the Netherlands Organization for the Advancement of Pure Research (Z.W.O.).

ISBN 90 6144 075 0
Juni 1977

ABSTRACT

The free distance [1] of a convolutional code determines to a large extend the error rate in the case of maximum likelihood (ML) decoding. This report describes a method to derive upperbounds on the bit error probability for a Viterbi decoder [2] with finite path register length. These bounds are compared with measurements for both the Viterbi- and for the Stack-algorithm [3]. An exhaustive search is carried out for several classes of R = k/n codes, including the classes mentioned by Schalkwijk et al [4,5]. These latter codes allow for exponentional hardware savings in the (syndrome) decoder. Tradeoffs between complexity and performance are considered.

I. INTRODUCTION

The complexity of a Viterbi decoder strongly depends on the path register memory. Heller and Jacobs [6], indicated that is does not pay to increase the decoder path register length above 4 to 5 times the encoder memory length.
Chapter II describes a method to derive upperbounds on the bit error probability for Viterbi decoders with finite path register length.
In chapter III, the above mentioned bounds are compared with measurements for the Viterbi- as well as for a Stack-decoder.
When using long constraint length codes, the derivations made in chapter II are not longer possible. Then, the minimum weight nonzero code word, or free distance is used.
The free distance d_{free} of a convolutional code is a good indicator of the error correcting performance of the code. Good algebraic nor analytic techniques for constructing codes with large d_{free} are known. Hence, one has to attemp search procedures.
As convolutional codes have an exponentional growth in the number of states with increasing length, computing free distance is an horrendeous job for long memory length codes. The algorithm used, is strongly related to the Fano algorithm, i.e. no excessive memory requirements.
The bias term in the Fano metric has been neglected, because this would lead to an algorithm that looks for long paths with a small Hamming weight, while most paths at free distance are short.
Another reason to omit the bias term is that it leads to a possible looping during the search procedure. The threshold setting is also different from that of the Fano decoding algorithm. We start with a threshold value equal to the best known upperbound on the free

distance, for the code being tested. If there is no path starting at the origin an remerging with the all zero path, without violating the threshold, then the free distance of the code is said to be equal to the original threshold value. The results of the exhaustive search are given in chapter IV.

II. BOUNDS ON THE ERROR PROBABILITY

The bit error probability for a Viterbi decoder is upperbounded by

$$P_b < \left. \frac{dT(D,N)}{dN} \right|_{N = 1, D = 2\sqrt{p(1-p)}} , \tag{1}$$

where T(D,N) is the generating function [7] of the encoder, and p the transition probability of the binary symmetric channel (BSC). The union bound (1) is obtained as follows.

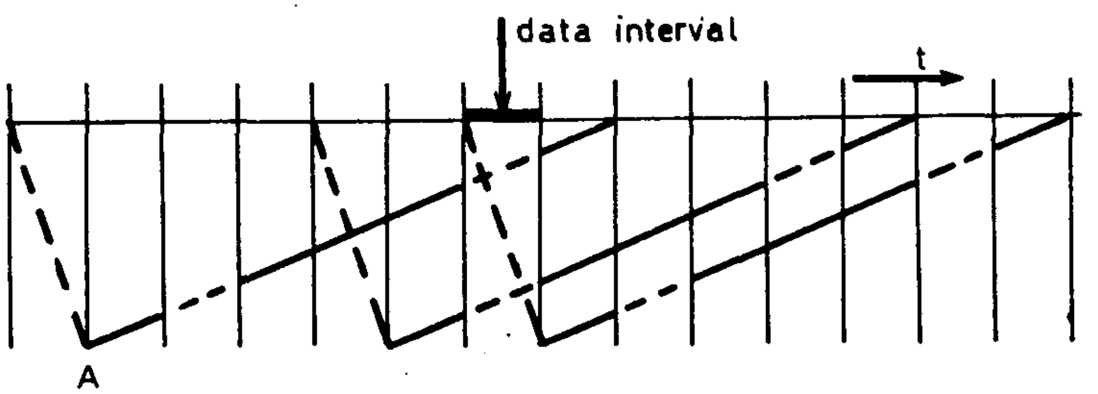

Fig. 1. Error-event A in the bit error causing phase for infite pathregister length.

Assume the all-zero codeword is sent. A certain error-event A with $D^k N^n$ can cause a bit error in the given data interval for n different values of the phase, see Fig. 1, where n=3. When observing only two binary codewords with Hamming distance k, the error probability P_k is upperbounded by

$$P_k < [2\sqrt{p(1-p)}]^k \quad . \tag{2}$$

Substituting the bound of (2) in the generating function

$$\left.\frac{dT(D,N)}{dN}\right|_{n=1} ,$$

the union bound (1) on the bit error probability is obtained. The above derivation is valid for a decoder with infinite path register length. When reducing this length to some finite value L, the situation is as indicated in Fig. 2. Note that we considered the same event A. The different phases are marked A, A' and A'', respectively.

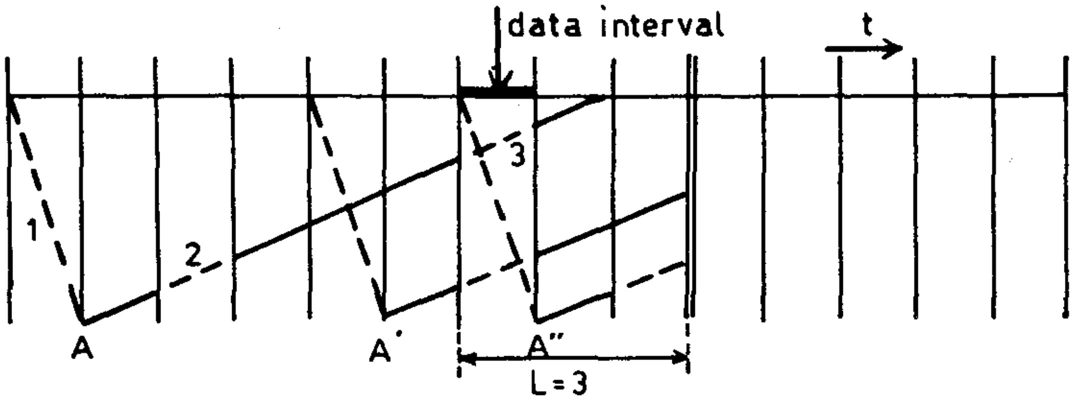

Fig. 2. Error-event A in the bit error causing phases for finite pathregister length.

Now the original event A can cause the same number of bit errors in the given data interval. However, the probabilities of the different phases of the event are no longer the same. For instance, the probability that bit error 1 lies in the given data interval is bounded by the two binary codeword error probability of the all zero sequence, and error event A''. Error event A'' is shorter than error event A, hence, its weight is less than or equal to the Hamming weight of event A. Hence, we have to find a new generating function for the error events. Then one can use techniques as given in [8].

III. CALCULATIONS AND MEASUREMENTS

The problem of finding a new generating function consists of two parts. First we have to know whether an error event causes a bit error in the given data interval or not. Secondly, if it does, we have to determine the weight of the event.

When observing the state diagram of a R = 1/n convolutional encoder, one can see that transitions to odd states correspond with an information bit equal to one. Hence, one only has to know the weight of error events bypassing an odd state. Given any odd state, we are able to calculate the generating function for paths from state zero to that particular odd state. This can be done with the usual techniques. The second part of the error event consists of paths leaving the particular odd state, and returning to the all zero state within L-1 steps, or to any other state in exactly L-1 steps. The generating function of the second part depends on the decoding length L, and can easily be determined by computer. Multiplying both functions, gives the generating function of all error events causing a bit error in the given data interval, for a particular odd state. Summation over all odd states leads to the following result.

$$T(D) = \sum_{j=0}^{2^{\nu-1}-1} G_{0,2j+1}(D)\, G^{L}_{2j+1,0}(D) \quad , \tag{3}$$

where $G_{0,2j+1}(D)$ is the generating function for paths from state zero to state 2j+1, and $G^{L}_{2j+1,0}(D)$ the generating function from state 2j+1 to state zero in 1,2,...,L-1 steps, or state i; $i=1,2,\ldots,2^{\nu-1}$, in L-1 steps. By substituting (2) in (3), an upperbound

$$P_b < T(D) \Big|_{D = [2\sqrt{p(1-p)}]} , \qquad (4)$$

on the bit error probability for decoders with finite path register length is obtained. These calculations can also be done for $R = k/n$ codes, by defining the states in a proper way.

The bound (4) has been evaluated for a code with generators $(1+X+X^2, 1+X^2)$ and for a code with generators $(1+X+X^3+X^4, 1+X+X^4)$. Figures 3, and 4 give the measured bit error rate P_b for various values of the path register length L as a function of the channel transition probability. Note that the bounds are in close agreement with the measurements.

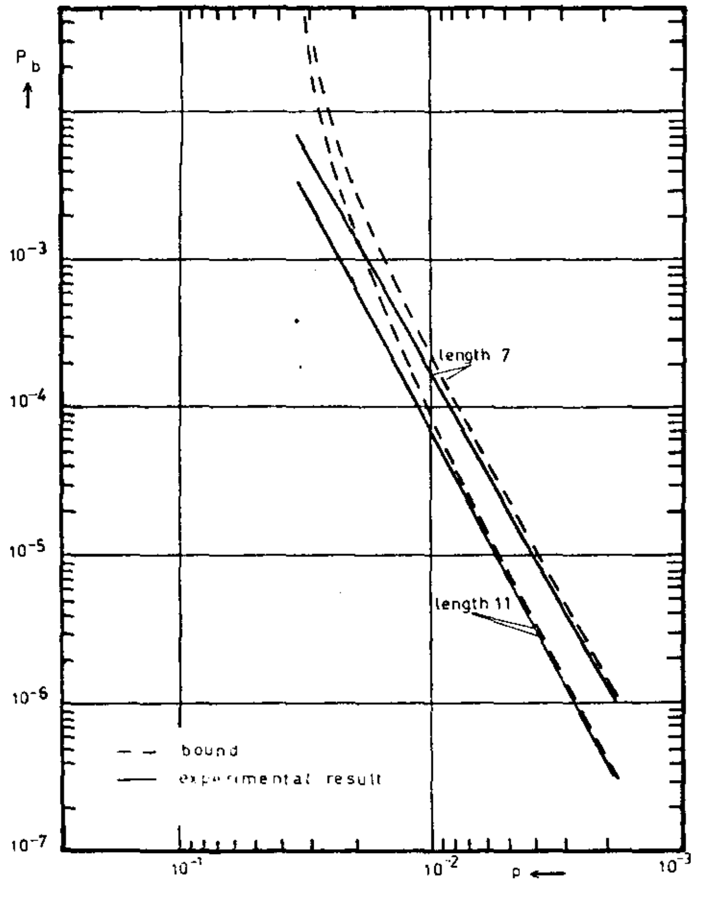

Fig. 3. Bit error rate P_b versus transition probability p for code with generators $(1+X+X^2, 1+X^2)$

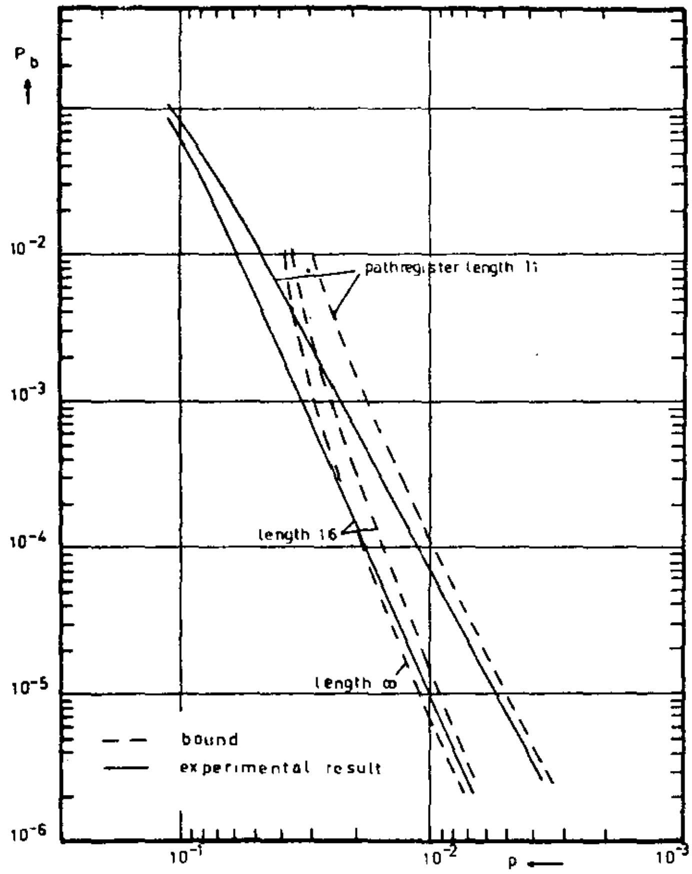

Fig. 4. Bit error rate P_b versus channel transition probability p for the code with generators $(1+X+X^3+X^4, 1+X+X^4)$

IV. FREE DISTANCE PROPERTIES OF SOME CLASSES OF BINARY CONVOLUTIONAL CODES

First, we investigated the whole class of binary $R = \frac{1}{2}$ codes, to find the best systematic - and non-systematic code for encoder lengths up to 15.

Earlier results only used the first term of v.d. Meeberg's bound [8]. Our program uses as many terms as needed to yield a unique code for each $K = 3,4,\ldots,15$, where K-1 is equal to the highest degree ν of the encoder connection polynomials. Tables 1, and 2 list the best systematic- and non-systematic codes, respectively.

K	C_1	C_2	d_f	B
4	1	13	4	1
5	1	33	5	8
6	1	53	5	3
7	1	123	5	1
8	1	267	7	17
9	1	647	7	6
10	1	1447	7	3
11	1	3053	7	1
12	1	5723	9	25
13	1	12237	9	14
14	1	27473	9	5
15	1	51667	11	95

Table 1. List of best systematic codes.

K	C_1	C_2	d_f	B
3	5	7	5	5
4	15	17	6	2
5	23	35	7	16
6	43	65	7	1
7	115	147	9	14
8	233	305	9	1
9	523	731	11	14
10	1113	1705	11	1
11	2257	3055	13	19
12	4325	6747	15	164
13	11373	12161	15	33
14	22327	37505	16	2
15	43543	71135	17	56

Table 2. List of best non-systematic codes.

The columns C_1 and C_2 give the connection polynomials in octal notation. The column d_f lists the free distance of the codes. Column B gives the total number of bit errors in error events of weight d_f if d_f is even, or of weight d_f and d_f+1 if d_f is odd, as in the latter case error events of distance d_f and d_f+1 have the same probability of occurance in a binary comparison with the no error sequence, see v.d. Meeberg [8].

A well known subclass for binary $R = \frac{1}{2}$ convolutional codes is the class of quick-look in (QLI) codes [9]. For this class the encoder polynomials differ in exactly one coefficient. Such codes permit recovery of the information sequence from the received data sequences using a single modulo-two adder. Table 3 lists QLI-codes for K up to 14.

	$\ell = 1$		2		3		4		5		6		Bound on d_f
K	C_1	d_f	C_1	df	C_1	df	C_1	df	C_1	df	C_1	df	
3	5	5											5
4	15	6											6
5	31	7	23	7									7
6	55	8	43	7									8
7	151	9	133	9	107	8							10
8	215	9	213	9	225	9							10
9	455	10	473	10	463	10	453	10					12
10	1071	11	1161	11	1227	11	1055	11					12
11	2255	12	3113	12	2623	12	2255	12	3117	12			14
12	6055	13	6113	13	4163	12	4147	12	5415	13			15
13	14135	14	11433	13	11225	13	10453	13	17611	13	14657	13	15
14	22755	15	31323	14	21227	14	32217	14	21735	14	26037	14	16

Table 3. Q.L.I. codes

The place, from the right hand side, where the generator polynomials differ, is given by ℓ, $\ell = 1,2,\dots,\left\lfloor\frac{K}{2}\right\rfloor$. The codes listed in Table 3 are not unique. It can be seen that the $\ell=1$ clas of QLI-codes is slightly superior. The number of computations required to determine the free distance of a code with $\ell=1$ was much smaller than for a code with $\ell = \left\lfloor \frac{K}{2} \right\rfloor$. As the computer search uses a Fano like algorithm, this fact points towards a better distance profil for the $\ell=1$ class of QLI-codes.

Complementary codes have been investigated by Bahl and Jelinek [10]. One remark should be made. The code with generator 120643, is catastrophic, and could be replaced by 121055.

As was shown in [4], syndrome decoding is an alternative to Viterbi decoding. If the degree $\nu=2\ell$, $\ell = 1,2,\ldots$, of the encoder polynomials is even and $C_{1,k} = C_{2,k}$, $0 \le k \le \nu$, $k \ne \nu/2$, the complexity of the syndrome decoder is proportional to $(\sqrt{3})^{\nu}$. In general, for codes with $C_{1,0} = C_{2,0} = C_{1,\nu} = C_{2,\nu} = 1$, and $C_{1,j} = C_{2,j}$, $C_{1,\nu-j} = C_{2,\nu-j}$, $1 \le j \le \ell-1$, $\ell \le \nu/2$, the number of redundant states is equal to

$$2^{\nu-2\ell}\,(4^{\ell}-3^{\ell}) \ , \ 1 \le \ell \le \nu/2 \quad . \tag{5}$$

However, when making these constraints on the code generators, it is questionable if the optimal free distance can be reached. Therefore, we have investigated the class of $R=\frac{1}{2}$ convolutional codes, with the above mentioned constraints. Table 4 gives the results for encoder lengths up to 14.

For a Viterbi decoder, the number of different decoder states is equal to 2^{ν}, where $\nu = K-1$. Table 5 gives the number of different decoder states needed to implement syndrome decoders with $\ell=1,2,\ldots,\left\lfloor \frac{K}{2} \right\rfloor$.

ℓ =	1	2	3	4	5	6	Bound on d_{free}
K							
3	5						5
4	6						6
5	7	7					7
6	8	8					8
7	10	9	8				10
8	10	10	10				10
9	12	11	10	10			12
10	12	12	12	11			12
11	14	14	12	12	12		14
12	15	14	14	14	14		15
13	15	15	15	15	15	13	15
14	16	16	16	16	15	14	16

Table 4. Free distance for codes with the constraint that $c_{1,0} = c_{2,0} = c_{1,\nu} = c_{2,\nu} = 1$, and $c_{1,j} = c_{2,j}$, $c_{1,\nu-j} = c_{2,\nu-j}$, $1 \leq j \leq \ell-1$, $\ell \leq \nu/2$.

ℓ =	1	2	3	4	5	6	Viterbi
K							
3	3						4
4	6						8
5	12	9					16
6	24	18					32
7	48	36	27				64
8	96	72	54				128
9	192	144	108	81			256
10	384	288	216	192			512
11	768	576	432	384	243		1024
12	1536	1152	864	768	486		2048
13	3072	2304	1728	1536	972	729	4096
14	6144	4608	3456	3072	1944	1458	8192

Table 5. Number of different decoder states needed to implement syndrome decoders with $\ell = 1,2,\ldots, \left\lfloor \frac{K}{2} \right\rfloor$.

In practice, one is interested in the minimum number of different states needed to implement a decoder for a convolutional code with a given free distance. This number of states is given in Table 6

free distance	5	6	7	8	9	10	11	12	13	14	15
Minimum # of states											
Viterbi	4	8	16	32	64	128	256	512	1024	1024	2048
Syndrome	3	6	9	18	36	48	144	192	486	486	972

Table 6. States needed to implement decoders with the given free distance.

for the Viterbi- as well as for the syndrome decoder.

From Table 6 observe that given the free distance the Viterbi decoder roughly needs twice as many states as does the syndrome decoder.

Binary R = k/n codes.

According to the criterion mentioned in Chapter IV, we investigated codes for R = 1/3 and R = 1/4, respectively. Again, these codes were found by an exhaustive search. The unique results are given in Tables 7, and 8.

K	d_{free}	c_1	c_2	c_3	#paths / # bit errors on d_{free} , d_{free} + 1	if necessary # paths / # bit errors on d_{free} + 2 / d_{free} +
3	7	5	5	7	1 / 1	
4	9	11	15	17	2 / 3	
5	11	25	27	33	3 / 6	4 / 13
6	13	47	55	67	3 / 6	
7	15	123	135	157	5 / 13	
8	15	211	327	353	1 / 1	6 / 17
9	17	455	623	727	1 / 1	12 / 46
10	19	1075	1127	1663	2 / 3	16 / 61
11	21	2127	3323	3751	4 / 7	21 / 89
12	23	4257	5575	6263	8 / 20	30 / 151
13	23	10663	15275	17051	1 / 1	10 / 35

Table 7. Best R = 1/3 codes in octal notation.

K	d_{free}	c_1	c_2	c_3	c_4	# path / # bit errors on d_{free} / d_{free} + 1	if necessary # path / # bit errors on d_{free} + 2 / d_{free} + 3
3	9	5	5	7	7	1 / 1	2 / 4
4	13	13	13	15	17	3 / 6	3 / 10
5	15	23	25	35	37	3 / 5	2 / 7
6	17	47	53	71	75	2 / 3	3 / 8
7	19	117	133	145	165	1 / 1	4 / 11
8	21	225	271	323	357	1 / 1	4 / 10
9	23	427	565	633	751	1 / 1	3 / 7
10	27	1133	1271	1517	1675	7 / 16	2 / 8
11	29	2327	2471	3133	3575	6 / 14	4 / 14

Table 8. Best R = 1/4 codes in octal notation.

As was shown [5] that syndrome decoding also is an alternative for Viterbi decoding in the R = k/n case. Tables 9, and 10 give the distances obtainable for R = 1/3 and R = 2/3 codes in the relevant class $\Gamma^{(n-k)}_{n,h,\ell}$.

k	ℓ = 1	2	3	4	bound on d_{free}
3	7				8
4	9				10
5	11	10			12
6	12	12			13
7	14	14	13		15
8	16	16	15		16
9	18	17	16	16	18
10	20	19	18	18	20

Table 9. d_{free} obtainable for R = 1/3 codes with metric/pathregister savings.

K	ℓ = 1	2	3	4	bound on d_{free}
3	3				3
4	4				4
5	5	5			5
6	6	6			6
7	6	6	6		7
8	8	7	6		8
9	8	8	8	6	8
10	8	8	8	8	9

Table 10. d_{free} obtainable for R = 2/3 codes with metric/pathregister savings.

REFERENCES

[1] D.J. Costello, "A construction technique for random-error-correcting convolutional codes", IEEE Trans. Information Theory, vol. IT-15, pp. 631-636, September 1969.

[2] A.J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm", IEEE Trans. Information Theory, vol. IT-13, pp. 260-269, April 1967.

[3] F. Jelinek, "A fast sequential decoding algorithm, using a stack", IBM J. Res. Develop., vol. 13, pp. 675-685, November 1969.

[4] J.P.M. Schalkwijk and A.J. Vinck, "Syndrome Decoding of Binary Rate-½ Convolutional Codes", IEEE Trans. on Communications, vol. COM-24, pp. 977-985, September 1976.

[5] J.P.M. Schalkwijk, A.J. Vinck, and K.A. Post, "Syndrome decoding of binary rate-k/n convolutional codes", to be published.

[6] J.A. Heller and I.M. Jacobs, "Viterbi decoding for satellite and space communication", IEEE Trans. on Commun. Technol., vol. COM-19, pp. 835-848, October 1971.

[8] L. van de Meeberg, "A tightened upper bound on the error probability of binary convolutional codes with Viterbi decoding", IEEE Trans. Inform. Theory, vol. IT-20, pp. 389-391, May 1974.

[9] J.L. Massey and D.L. Costello, Jr., "Nonsystematic convolutional codes for sequential decoding in space application", IEEE Trans. on Comm., vol. COM-19, pp. 806-813, October 1971.

[10] L.R. Bahl and F. Jelinek, "Rate-½ convolutional codes with complementary generators", IEEE Trans. Inform. Theory, vol. IT-17, pp. 718-727, Nov. 1971.