THE BELL SYSTEM TECHNICAL JOURNAL

DEVOTED TO THE SCIENTIFIC AND ENGINEERING ASPECTS OF ELECTRICAL COMMUNICATION

Volume 49 | April 1970 | Number 4

Performance of Burst-Trapping Codes

By S. Y. TONG

(Manuscript received November 6, 1969)

The performance of several burst-trapping codes has been estimated through computer simulation on three sets of field trial data over the switched telephone network (Alexander-Gryb-Nast, Townsend-Watts, Vestigial-Sideband). The results indicate that the codes are capable of giving better performance than interleaved codes based on the same coding redundancy and approximately the same storage requirement.

I. INTRODUCTION

An adaptive error control scheme called burst-trapping has been proposed as a means to combat errors in a channel where both random and burst disturbances occur.[1] Such a channel is called a compound channel. Error statistics derived from field trial data over telephone channels indicate that the switched telephone network is a prime example of a compound channel.

It is of interest to see how the proposed technique performs on telephone channels and to compare its performance with other known techniques. In this report, a computer simulation is used to determine the performance of such codes. A program has been written which simulates any burst-trapping code of rate $(b - 1)/b$, b an integer. The program was used on the three well-known sets of telephone error data: the

Alexander-Gryb-Nast (AGN) data,[2] the Townsend-Watts (TW) data[3] and the Vestigial-Sideband (VSB) data.[4]

The output of the simulation program gives both the bit and block error rate before and after the application of burst-trapping error control. The simulated block error rate with a large interleaving degree shows an excellent agreement with the result of an analytical approach presented in the companion paper.[5] Since two independent approaches are used, the agreement seems to indicate both techniques are accurate. The analytical technique gives the tail of distribution which cannot be produced by the simulation without a large amount of raw data. On the other hand, the simulation technique can show the effect of interleaving on performance. Thus it is a valuable design tool to determine the size of interleaving required which in turn determines the cost of the error control system. (Note that in Ref. 5, interleaving is assumed to be sufficiently large that each subcode essentially has statistically independent error blocks.) Thus these two techniques complement each other for the evaluation of performance of a burst-trapping code.

II. A REVIEW OF BURST-TRAPPING CODES

The reader is assumed to be familiar with Ref. 1 on burst-trapping techniques. We shall therefore confine ourselves to a brief review and will restrict our attention to codes with rate $R = \frac{2}{3}$, as this is the lowest rate that reveals the general operation. For a general description, see Ref. 1.

Let C be a linear systematic binary block code of length n with k information bits per block, such that $k/n = \frac{2}{3}$. Let I_i^1 be a $k/2$-tuple representing the first $k/2$ information bits in the ith transmitted block and let I_i^2 be similarly defined for the second $k/2$ information bits. Let P_i be the parity digits in the code word associated with I_i^1 and I_i^2. (Note that P_i is also a $k/2$-tuple.) Finally, define Q_i as

$$Q_i \triangleq P_i + I_{i-1}^1 + I_{i-2}^2$$

where addition is bit-by-bit, modulo-2. Then the transmitted message consists of the sequence of $k/2$-tuples, I_0^1, I_0^2, Q_0, I_1^1, I_1^2, Q_1, $\cdots$, I_i^1, I_i^2, Q_i, $\cdots$, and so on, as shown in Fig. 1.

At the decoder there are two modes of operation, the *random* mode and the *burst-trapping* mode. Let the minimum distance of C be d_m and assume that the decoder is designed to correct up to t errors per block where $t \leq [(d_m - 1)/2]$. Now assume the decoder has just received

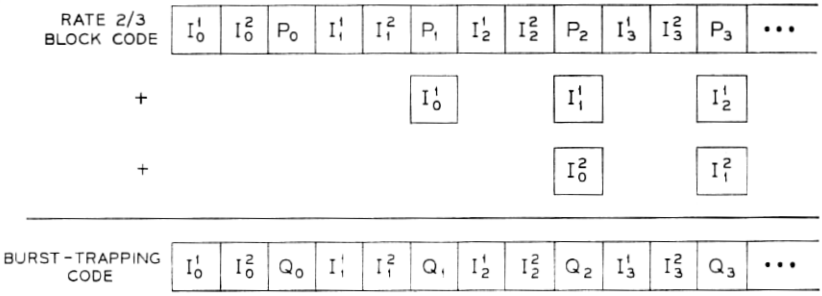

Fig. 1—Generation of $R = 2/3$ burst-trapping code.

the 0th block. If $I_i^j = 0, i < 0, j = 1, 2$, then $Q_0 = P_0$ and the sequence I_0^1, I_0^2, Q_0 forms a code word in C. If the decoder "thinks" that there are t or fewer errors in the received block, then it operates in the random mode and attempts to correct these errors. If the decoder is successful then I_0^1 can be subtracted from Q_1 and I_0^2 from Q_2 through feedback as shown in Fig. 2. As long as decoding proceeds correctly in this fashion, then, upon the arrival of each successive block, the effect of all past information bits will be removed and the decoder will be left with a code word in C plus any channel errors which may have occurred. Thus, in the random mode, the decoder acts like an ordinary block decoder. Let us now assume that at the first block the decoder detects an error pattern of weight greater than t. It then enters the burst-trapping mode, inhibits feedback of the information bits and delays the estimation of I_1^1 and I_1^2 until the second and third blocks have arrived. The course of action then taken by the decoder is shown graphically in Fig. 3. If the second block is correctly received, then P_2 can be calculated

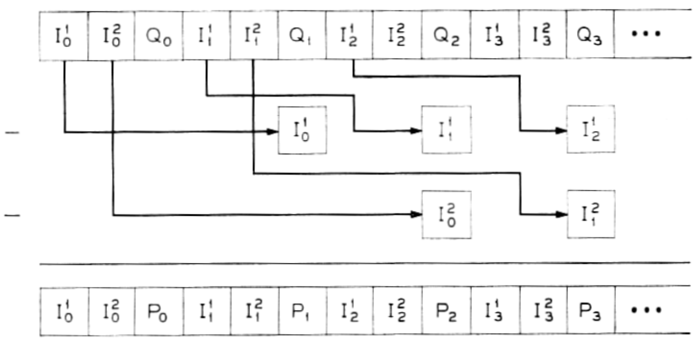

Fig. 2—Error correction in random code.

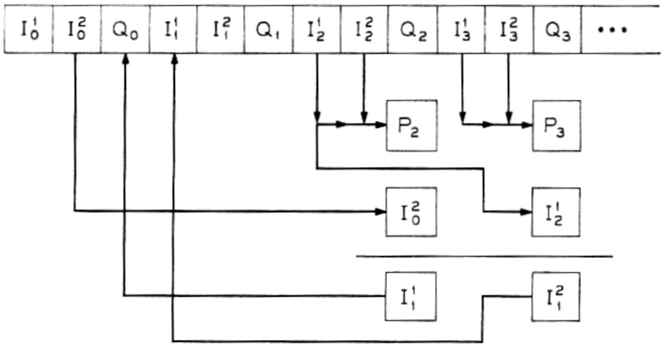

Fig. 3—Error correction in burst code.

from I_2^1 and I_2^2 and subtracted from Q_2. Assuming that the 0th block has been correctly decoded, I_0^2 may also be subtracted from Q_2, leaving I_1^1, which replaces the first $k/2$ discarded information bits in the first block. If the third block is also correctly received, then I_3^1 and I_3^2 may be used to calculate P_3 which, along with I_2^1, is subtracted from Q_3, leaving I_1^2 which, in turn, replaces the second $k/2$ discarded information bits in the first block. Thus I_1^1 and I_2^1 will be recovered error-free if and only if the second and third blocks are error-free. This is the guard space for correcting a burst in the first block. To summarize this discussion, a block error pattern which is detectable with C is correctable with the burst-trapping decoder if and only if the next two blocks are error-free. The probability of failing to detect an error can be made small by keeping t small. In such a case, most of the cosets of C are used for detection, with only a few being used for correction. Such a strategy is often successful on channels where the probability of error between bursts is quite low.

As with any feedback convolutional decoding algorithm, this discussion is valid only under the assumption that all previous blocks have been decoded correctly. In the event of a decoding error, it has been shown that error propagations are rare and are limited.[1] Now suppose Q_i is redefined as

$$Q_i \triangleq P_i + I_{i-\ell}^1 + I_{i-2\ell}^2 .$$

Then the code is said to be *block-interleaved to degree* ℓ. In the burst trapping mode, the decoder then recovers I_i^1 and I_i^2 from the $(i + \ell)$th and the $(i + 2\ell)$th blocks, respectively. Interleaving produces, in effect, ℓ distinct codes of the type described before and if a burst is limited to ℓ or fewer blocks, only one block in each code will be affected. It follows that a burst confined to ℓ successive blocks is correctable if

(i) no undetected errors occur in any of the ℓ blocks and (ii) the next 2ℓ blocks are error-free. The latter condition is not a necessary one since many burst patterns, particularly on telephone channels, will contain blocks with t or few errors. If this is true for, say, the kth block, then the $(k + \ell)$th and $(k + 2\ell)$th blocks need not be error free.

In the next section, the simulation results on bit error rates are given. It is shown that the burst-trapping codes, when used on telephone channels, seem to perform better than other known codes, with the same coding redundancy and same storage requirement.

III. SIMULATION RESULTS

AGN data was collected over the nation-wide switched network in 1959 with an FM data set operating at 600 b/s and 1200 b/s. Data from a total of about 1000 calls were recorded. The distribution of the average bit error per call is reproduced in Fig. 4. As will be shown later, the burst-trapping codes used in the test correct all the errors in at least 95 percent of the calls. Since the distribution of calls above 90 percent level is almost identical for both 600 and 1200 b/s calls, it was decided to combine the two types of calls together to simplify the discussion.

TW data was collected at 2000 b/s. A DATAPHONE 201A modem

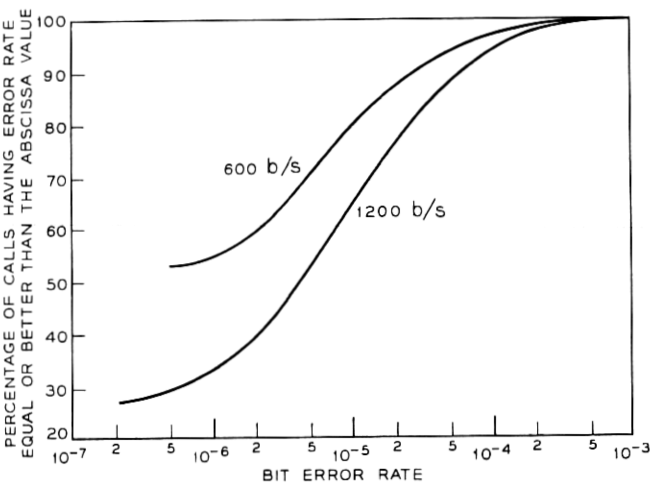

Fig. 4—Alexander-Gryb-Nast raw data statistics. Distribution of calls with respect to bit error rate and speed. Total number of calls—1000.

was used. A total of 502 usable calls* was recorded. The statistics of the calls are reproduced in Fig. 5.

VSB data used in this report consists of 85 selected calls. The statistics of the calls are shown in Fig. 6.

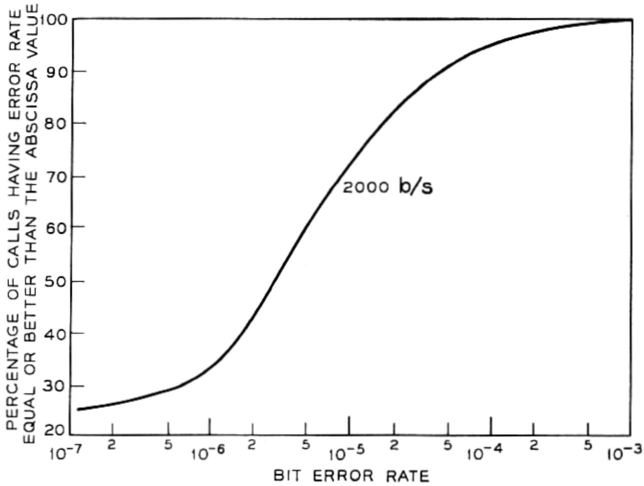

Fig. 5—Townsend-Watts (TW) raw data statistics. Distribution of calls with respect to bit error rate. Total number of calls = 502.

To simplify discussion, a quintuple (n, k, d_m, t, ℓ) is used to describe each burst-trapping code. Recall that a burst-trapping code is built on the basis of a (n, k, d_m) block code, where n is the block length, k is the number of information bits per block and d_m is the minimum distance of the block code. The parameter t denotes the number of random errors the burst-trapping decoder is designed to correct and ℓ is the interleaving degree. Roughly, with a (n, k, d_m, t, ℓ) burst-trapping code one may either correct t random errors in n bits or up to an $n\ell$-bit burst provided that a guard space of $n^2\ell/(n - k)$ bits exists. The storage requirement is $nk\ell/(n - k) + n + \ell$ bits. For more details, please refer to Ref. 1.

Figure 7 compares the performance of two rate $\frac{1}{2}$ codes on VSB data: a (24, 12) triple-error-correcting extended Golay code interleaved to

---

* Reference 3 indicates 548 completed calls. However, Ref. 6 indicates 503 completed calls of which 136 were error-free and 367 contained errors. The data available on magnetic tape and used in this study are consistent with the number of calls in Ref. 6 with the exception that one call was uninterpretable. Therefore, we shall use the calls from Ref. 6 with the exception of the aforementioned call leaving our sample to be of 502 calls of which 136 were error-free. (Courtesy of Messrs. F. X. Brophy and M. M. Buchner, Jr.)

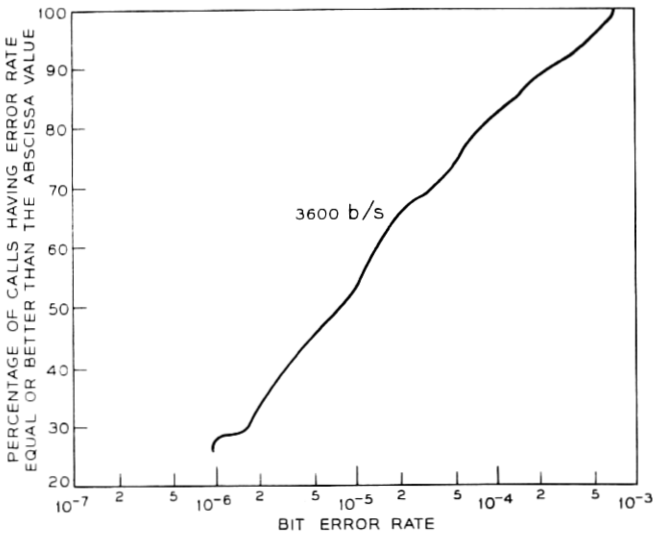

Fig. 6—Raw data statistics for multilevel Vestigial-Sideband (VSB) modem—with respect to bit error rate. Total number of calls = 85.

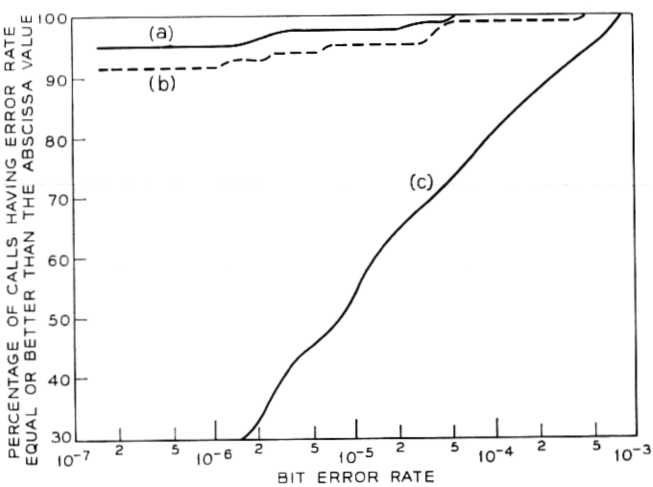

Fig. 7—Comparison of rate 1/2 codes. (a) (24, 12, 8, 1, 330) burst-trapping code. Using (24, 12) Golay code corrects single random error and burst up to 7920 bits. Storage = 4214 bits. (b) Interleaved (24, 12) extended Golay code corrects three errors. Interleaving degree = 175. Corrects bursts up to 525 bits. Storage = 4200. (c) VSB raw data.

degree 175 and a burst-trapping code based on the same Golay code but with single-error-correction only. As shown in the figure, their storage requirements are about the same; yet the (24, 12, 8, 1, 330) burst-trapping code out-performs the interleaved Golay code at a significantly lower cost (single-error correction vs. triple-error correction).

Figure 8 compares the performance of two rate $\frac{2}{3}$ codes: a (57, 38) triple-error-correcting shortened BCH code interleaved to degree 75

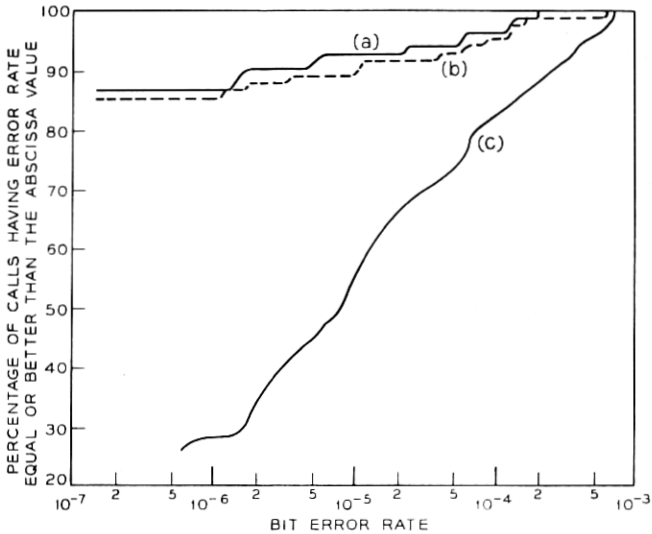

Fig. 8—Comparison of two rate 2/3 codes. (a) (57, 38, 8, 2, 50) burst-trapping code. Using (57, 38) BCH code corrects 2 random errors and bursts up to 2850 bits. Storage = 3957 bits. (b) Interleaved (57, 38) BCH code corrects three errors and bursts of 275 bits. Storage = 4275 bits. Interleaving degree = 75. (c) VSB raw data.

and a (57, 38, 8, 2, 50) burst-trapping code. Figure 9 also compares two rate-$\frac{2}{3}$codes: an interleaved (degree 25) (255,170) compound code* that corrects seven random errors and bursts of 1000 bits (after interleaving) and a (126, 84, 14, 6, 25) burst-trapping code. It is seen, that the burst-trapping code outperforms the compound code at smaller storage requirement and lower logic cost (six error vs. seven error correcting). A more striking result is shown in Fig. 10, where the performance of the (255, 170) compound code is compared with a (39, 26, 6, 1, 117) burst-trapping code, which performs better than the compound code; yet only single-error-correction is required (vs. seven error

* Taken from page 137 Table 1 of Ref. 7.

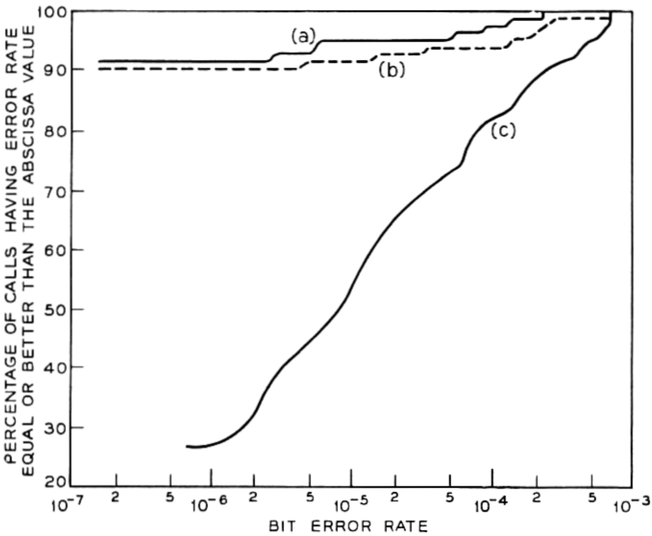

Fig. 9—(255, 170) compound code vs. (126, 84, 14, 6, 25) burst-trapping code. (a) (126, 84, 14, 6, 25) burst-trapping code using (126, 84) BCH code corrects 6 random errors and bursts up to 3150 bits. Storage required = 4376 bits. (b) Interleaved (255, 170) compound code that corrects 7 random errors and 1000-bit burst errors with interleaving degree 25. Storage required = 6375 bits. (c) VSB raw data.

correction required for the compound code). Note that the compound code chosen is a very powerful one. The comparison simply shows that this type of code might not be suitable for use on telephone channels.

Figure 11 shows the effect of the trade-off between random-error-correction and burst-error-correction for a typical burst-trapping code. In this case, a (57, 38) BCH code was used to correct 0, 1, 2 and 3 errors with correspondingly decreasing burst-error-detecting capability and consequently lower burst-correcting capability.

The (57, 38, 8, 0, 50) is simply a burst-error-correcting code. Its performance reflects the fact that telephone channels do not produce burst errors only. The (57, 38, 8, 3, 50) code is essentially a random-error-correcting code. Such a code, as expected, does not do well for telephone channels either. The (57, 38, 8, 1, 50) single-error-correcting and the (57, 38, 8, 2, 50) double-error-correcting burst-trapping codes perform better. This seems to confirm the assertion that telephone channels can be characterized as a type of compound channel.

Figures 12 through 16 show the performance of various burst-trapping codes on AGN and TW data. It is seen that more than 93 percent of

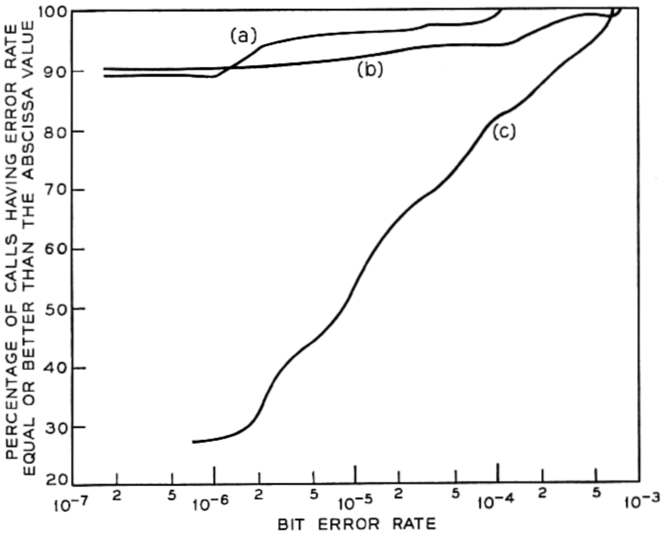

Fig. 10—(255, 170) compound code vs. (39, 26, 6, 1, 117) burst-trapping code. (a) (39, 26, 6, 1, 117) burst-trapping code which corrects single error and bursts up to 4563 bits. Storage required = 6357 bits. (b) Interleaved (255, 170) compound code which corrects 7 random errors and 1000-bit burst errors with interleaving degree 25. Storage required = 6375 bits. (c) VSB raw data.

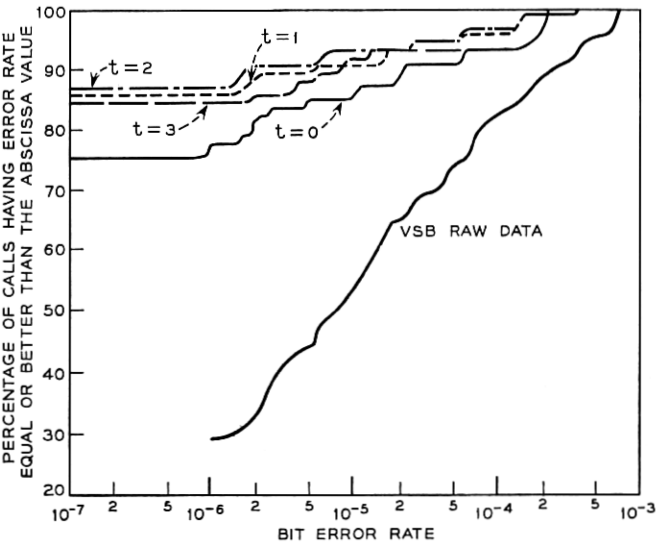

Fig. 11—Effect of trade off on VSB channels. Burst-trapping code using (57, 38) code with different mix of random and burst-error-correcting capability. Interleaving degree $L = 50$. Storage = 3957 bits.

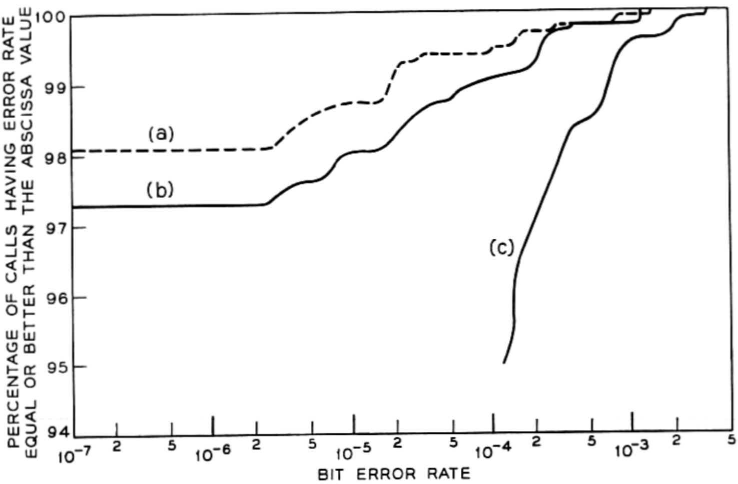

Fig. 12—Performance of two burst-trapping codes over AGN data. (a) (57, 38, 8, 2, 50) burst-trapping code corrects burst up to 2650 bits. Storage required = 3957 bits. (b) (57, 38, 8, 2, 26) burst-trapping code corrects bursts up to 1482 bits. Storage required = 2085 bits. (c) AGN raw data.

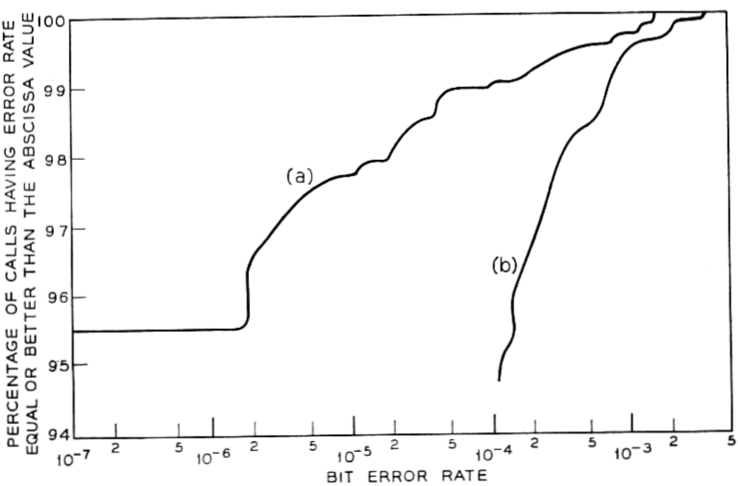

Fig. 13—Performance of (30, 20, 5, 1, 50) burst-trapping code. (a) (30, 20, 5, 1, 50) burst-trapping code corrects single error and bursts up to 1500 bits. Storage required = 2130 bits. (b) AGN raw data.

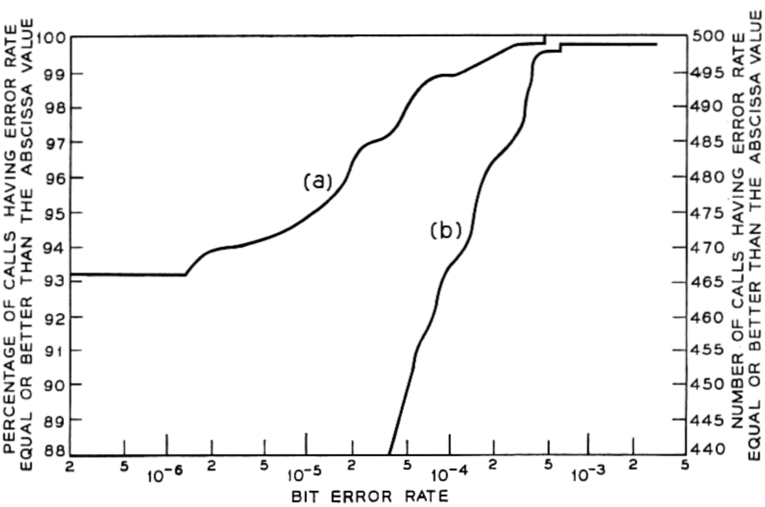

Fig. 14—Performance of (57, 38, 8, 2, 26) burst-trapping code over TW data. (a) (57, 38, 8, 2, 26) burst-trapping code corrects double errors and bursts up to 1482 bits. Storage required = 2085 bits. (b) TW raw data.

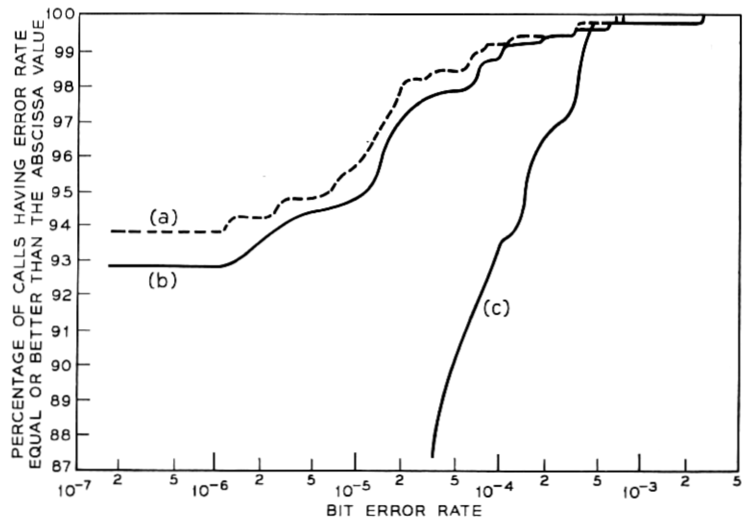

Fig. 15—Performance of two burst-trapping codes over TW data. (a) (39, 26, 6, 1, 80) burst-trapping code corrects burst up to 3120 bits. Storage required = 4379 bits. (b) (39, 26, 6, 1, 40) burst-trapping code corrects burst up to 1560 bits. Storage required = 2199 bits. (c) TW raw data.

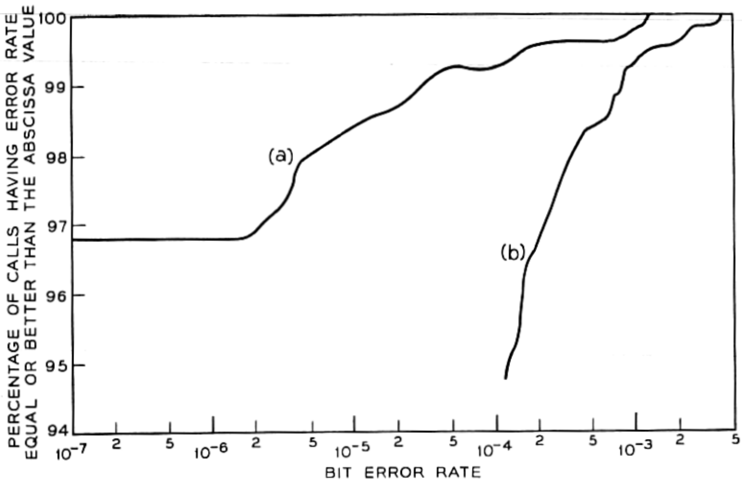

Fig. 16—Performance of (39, 26, 6, 1, 80) burst-trapping code. (a) (39, 26, 6, 1, 80) burst-trapping code corrects single error and bursts up to 3120 bits. Storage required = 4359 bits. (b) AGN raw data.

calls will be error-free if any of these codes is used for error control. All can be implemented at very modest cost.

Finally, we compare the performance of a (24, 20) convolutional single-error-correcting code interleaved to degree 127 with a burst-trapping code. The interleaved (24, 20) code is planned to be used in an optional error control unit for the 203 data set.[8,9] The performance of the (24, 20) code has been simulated by W. K. Pehlert with VSB data.[10]

Figure 17 curve (a) shows the performance of a (90, 75, 6, 1, 8) burst-trapping code with the same data; this code requires about the same storage* and has the same rate but performs better than (24, 20) code. Figure 17 curve (c) shows a more contrasting comparison. The (90, 75, 6, 1, 6) burst-trapping code requires only three-quarters as much storage yet still performs about the same as (24, 20) code. Before one attempts to draw any conclusion, however, it must be emphasized that the result is based on a relatively small sample size; hence wide variations are expected.

* We assume the (24, 20) code uses minimum storage, that is, 3048 bits although the actual implementation uses about 3600 bits of storage in order to simplify decoding logic.

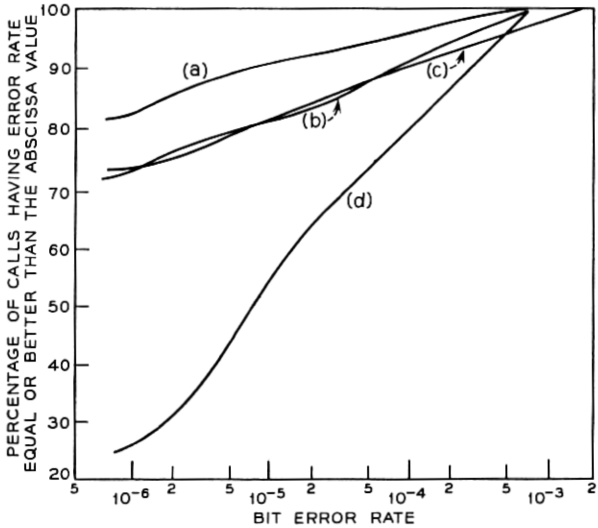

Fig. 17—Comparison of rate 5/6 error-correcting codes. (a) (90, 75, 6, 1, 8) burst-trapping code corrects burst up to 720 bits. Storage required = 3000 bits. (b) (24, 20) single error correcting code interleaved to degree 127. Corrects bursts up to 127 bits. Storage required = 3000 bits. (c) (90, 75, 6, 1, 6) burst-trapping code corrects bursts up to 540 bits. Storage required = 2250 bits. (d) VSB raw data.

IV. CONCLUSION

We have shown, through computer simulation, the performance of some burst-trapping codes, and have compared such codes with other known codes using telephone channel error statistics. It seems that several conclusions can be drawn, based on the available data.

(*i*) There are always a few percent of the calls that are so bad that it is difficult to improve them through forward error-control.

(*ii*) The performance of burst-trapping codes compares favorably with interleaved codes.

(*iii*) For a given level of performance, burst-trapping decoders are generally simpler to implement than decoders for other schemes that have been proposed for telephone channels.

V. ACKNOWLEDGMENTS

Thanks are due to H. O. Burton and E. R. Kretzmer for their valuable comments. W. K. Pehlert provided the simulated (24, 20) code performance data. The brief description of burst-trapping code is taken from a private memorandum co-authored by D. D. Sullivan and myself.

REFERENCES

1. Tong, S. Y., "Burst-Trapping Techniques for a Compound Channel," IEEE Trans. on Information Theory, *IT-15*, No. 6 (November 1969), pp. 710–715.
2. Alexander, A. A., Gryb, R. M., Nast, D. W., "Capabilities of the Telephone Network for Data Transmission," B.S.T.J., *39*, No. 3. (May 1960), pp. 431–476.
3. Townsend, R. E., and Watts, R. N., "Effectiveness of Error Control in Data Communication over the Switched Telephone Network," B.S.T.J., *43*, No. 6 (November 1964), pp. 2611–2638.
4. Farrow, C. W., and Holzman, L. N., "Nationwide Field Trial Performance of a Multilevel Vestigial-Sideband Data Terminal for Switched-Network Voice Channels," 1968 IEEE International Conference on Communications, Conference Record 68C20-COM (June 1968), pp. 782–787.
5. Pehlert, Jr., W. K. "Analysis of a Burst-Trapping Error Correction Procedure," B.S.T.J., this issue, pp. 493–519.
6. Townsend, R. L., unpublished work.
7. Hsu, H. T., Kasami, T., Chien, R. T., "Error-Correcting Codes for a Compound Channel," IEEE Trans. on Information Theory, *IT-14*, No. 1 (January 1968), pp. 135–139.
8. Mecklenburg, P., unpublished work.
9. Kretzmer, E. R., "Telephoning Data Faster," Bell Laboratories Record, *47*, No. 7 (August 1969), pp. 239–243.
10. Pehlert, Jr., W. K., unpublished work.