THE EFFECT OF CODING ON RATE EQUALIZATION OF DIGITAL CHANNELS

S. BUTMAN
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California

Summary The pulse stuffing technique for rate equalization of digital channels is generalized in this article to the stuffing of a sequence of pulses (a word), which can be coded. The extra capacity needed for signaling the stuffed word decreases exponentially with the number of pulses in it, and may, in fact, be eliminated at a negligible increase in the error rate of the channel.

Introduction In digital communication networks it is often necessary to time-division multiplex a number of digital channels with different and fluctuating pulse rates into a single high rate link, and to invert this operation (demultiplex, or decommutate) at various receiving points down the line. In order for this process to function properly, it is necessary to have the pulses in each channel occur at specified instants in time so that they may be sampled in an order known to the receiving stations. This retiming process is called synchronization. It can be effected by reading the pulses of each channel into an elastic memory buffer, and reading out the contents of each buffer in sequence.

From the point view of an individual channel, synchronization is equivalent to rate equalization since the fluctuating buffer input rate $r(t)$ is converted to the synchronous buffer output rate s. Buffer overflow, which leads to loss of data, can be prevented by having $s \geq \max_t r(t)$. The difference

$$s - r(t) \geq 0 \tag{1}$$

is the buffer depletion rate, and it must be made up by inserting extra pulses (or time slots) into the output stream. These stuffed pulses must be identified and deleted at the receiving stations (after decommutation) in order to prevent decoding errors. The transmission of this identification information requires extra channel capacity.

Several synchronization schemes using pulse stuffing have been reported in [1-7]. In each case a pulse is inserted into the output stream as soon as the memory buffer is empty. The methods differ, however, in the manner of signaling the presence of the stuffed pulse. In added bit signaling [6,7], the signaling channel is created by adding an extra pulse for every N data pulses, this corresponds to an increment of 1/N in the total channel capacity. In statistical subcarrier signaling [4], each of the I most probable data words is assigned two symbols A_i and B_i = 1,2,....I, and the presence of a stuffed pulse is signaled by substituting A_i's for B_i's. In this case the amount of additional channel capacity set aside for signaling stuffed pulses is not clearly defined, however, it is evident that the introduction of extra symbols into the channel alphabet is equivalent to increasing the capacity.

The purpose of the present discussion is to generalize the concept of pulse stuffing to word stuffing, a word being defined as a sequence of k pulses, in order to reduce the capacity of the signaling channel. This requires that the buffer memory increase proportionately to the length of the stuffed word. The question of whether to use more memory and less signaling capacity or vice versa is a matter of economics of the particular situation.

Equalization Using Stuffed Words The operation of an elastic memory buffer is analogous to a reservoir which can be filled intermittently while being drained at a steady rate. Consider a buffer with a memory size of k pulses. When the contents of this buffer have been depleted to some reference, say zero, the buffer must be allowed to refill by a temporary pause in its readout. At the same time a predetermined sequence of k pulses, a word W, is inserted into the output stream in order to maintain the synchronous rate s. (Actually there is no such thing as not inserting pulses, since the absence of a signal is part of the code). The time interval involved in the word stuffing operation is k/s, and during this time interval

$$\begin{aligned} k' &= \int_0^{k/s} r(t)dt \\ &= k - \int_0^{k/s} [s - r(t)]\, dt \\ &\leq k \end{aligned} \qquad (2)$$

pulses for reading into the buffer. Because of the discrete nature of read-in and read-out mechanisms, k' is either the greatest integer in, or the nearest integer to the quantity on the right hand side of (2). If r is the average value of r(t), then the average value of k' is kr/s.

The average rate of occurrence of stuffed words is equal to the average depletion rate divided by the average number of pulses read in, that is, s(s-r)/kr. Thus, although the exact time of occurrence of a stuffed word is Unpredictable, stuffed words occur on the average only one kth as often as stuffed pulses, and require only one kth the signaling rate.

An even better approach is to signal the occurrence of a data word W rather than the stuffed word. Thus, in the absence of signaling, the receiver is arranged to detect and delete the sequence W, and signaling is used to inhibit deletion when there is a data sequence W. In this way, only the stuffed word W is removed, provided that there are no transmission errors. For a non-zero channel error probability, the situation becomes somewhat more complicated due to the possibility of errors of the first and second kind (non detection of W and false alarms), this matter will be taken up in the next section. If the data stream consists of m equiprobable pulses, the probability of a data word being W is m^{-k} and the average signaling rate is only $r/(km^k)$, which is smaller than the rate of stuffed words s(s-r)/kr when $k > \log_m [r^2/s(s-r)]$. Finally, it is of interest to observe that signaling may be discarded at the price of allowing the error probability to increase by about m^{-k}. This is accomplished by forcing the transmitter to change one (or more) pulses in the data sequence W. If the channel were error free, all data would pass through the W detector at the receiver, however, the data word that was W will have one pulse in error, causing the average error probability to increase by m^{-k}/k. The number of intentional errors will increase when redundancy is introduced to combat errors of the first and second kind.

Errors of the First and Second Kind Let p be the error probability for a binary symmetric channel (m = 2). The probability of not detecting the stuffed word, P_1 , is then

$$P_1 = 1 - (1-p)^k \tag{3}$$

and it increases with k. It can, however, be decreased by redundant coding of W. Thus, the receiver is instructed to take any sequence of k pulses with a Hamming distance d or less from W as the stuffed word itself. This, of course, increases the number of data words that will be mistaken for W, unless they are intentionally put in error at the transmitter, or, unless the signaling rate is increased. The number of errors that the transmitter must introduce to change the distance from i to d+1 is d+1-i, and the number of words at a distance i is $\binom{k}{i}$. Therefore, the intentional average probability of error, P_I, is

$$P_I = 2^{-k} \left[\sum_{i=0}^{d} (d+1-i)\binom{k}{i} \right] /k \tag{4}$$

or, the average rate of the inhibit signal is rP_I. However, the probability of an error of the first kind is now only

$$P_I = \sum_{i=d+1}^{k} \binom{k}{i} p^i (1-p)^{k-i} \tag{5}$$

A false alarm, or an error of the second kind, is the event that a sequence of k data pulses originally at a distance $h \geq d+1$ is in the course of transmission altered to a sequence whose distance from the stuffed sequence is $\leq d$. The probability of error of the second kind, P_2, increases with d, its minimum value being equal to P_1 when d is zero. In order to calculate P_2, note that all but h of the pulses in a sequence at distance h from W are the same as in W. Therefore, the distance decreases by i when there are i errors in the h unlike pulses, and increases by j when there are j errors in the remaining k-h pulses. Since there are $\binom{h}{i}$ ways of making i errors in the h unlike pulses, and $\binom{k-h}{j}$ ways in the remaining k-h pulses, the probability of changing the distance from h to h+j-i is

$$p\left\{h \rightarrow h+j-i\right\} = \binom{h}{i}\binom{k-h}{j} p^{i+j}(1-p)^{k-i-j} \quad . \tag{6}$$

and the probability of having $j-i \leq d-h$ is

$$p\left\{j-i \leq d-h\right\} = \sum_{i=h-d}^{h} \sum_{j=0}^{i+d-h} \binom{h}{i}\binom{k-h}{j} \cdot p^{i+j}(1-p)^{k-i-j} \tag{7}$$

Finally, since the number of words at distance h from W is $\binom{k}{h}$ except that for h = d+1 there are an additional $\sum_{h=0}^{d} \binom{k}{h}$ words that were originally at a distance $\leq d$, the probability of a false alarm is

$$P_2 = 2^{-k}\left[p\,\{j-i \leq 1\} \sum_{h=0}^{d+1} \binom{k}{h} + \sum_{h=d+2}^{k} \binom{k}{h} p\,\{j-i \leq d-h\}\right] \tag{8}$$

Performance The total increment in the average error probability is then

$$\Delta p = s(s-r)P_1/r^2 + P_2 + P_I \tag{9}$$

For a given value of k, the above expression will have a minimum with respect to d because P_1 decreases with d while P_2 and P_1 increase. In many practical situations $\Delta p \approx P_I$. For example, consider a channel with $0 \leq (s-r)/r \leq 1$ and an error probability $p = 10^{-6}$. Then, if k = 30 and d = 1, P_1 and P_2 are negligible compared to P_I which is $\sim 2^{-30} \approx 10^{-9}$, so that Δp is only .1% of p; and this is with no signaling.

It might be argued that errors of the first or second kind are the most catastrophic, since their occurrence upsets the data flow, causing, possibly, serious difficulties in decoding. It is, therefore, of interest to minimize the average rate of occurrence of undetected stuffed words or false alarms. The relevant expression, when s ≈ r, so that $s(s-r)/r^2 \approx (s-r)/r$, is

$$u = [(s-r)P_1 + rP_2]/k \tag{10}$$

where u is the average rate of occurrence of errors of the first or second kind. Table I presents u in events/year, for a channel with $p = 10^{-6}$, $s = 1.544 \times 10^6$ pps and $s-r = 200$ pps, as a function of k and d. The minimum value of u for each choice of k is underlined, indicating the best choice for d.

d \ k	10	20	30	40
0	<u>60,500</u>	6,500	6,500	6,500
1	5,500,000	<u>1,100</u>	<u>1.4</u>	1.3
2		5,500	20	<u>.04</u>
3				4.0

Table I. Yearly occurrence of errors of the first or second kind as a function of stuffed word length k and redundancy d for a channel with $P = 10^{-6}$, $s = 1.544 \times 10^6$ pps and s-r = pps.

Conclusion This article investigated the performance of rate equalization schemes based on the concept of pulse stuffing. It was shown that by increasing the length of the sequence of stuffed pulses, a previously neglected quantity, it is possible to reduce, and even to eliminate, the need for an auxiliary signaling channel. The analysis presented for the binary symmetric channel is a worst-case solution because each data sequence was assumed to be equally likely. If non-uniform statistics prevail, the stuffed sequence should be the least probable data word. Finally, it should be noted that word stuffing is particularly suitable in equalizing channels with large fluctuations in rate.

References

1. J. S. Mayo, “PCM Netword Synchronization,” U. S. Patent 3136861, 1964.

2. J. S. Mayo, “Experimental 224 Mb/s PCM Terminals,” Bell Sys. Tech. J., Vol. 44, pp. 1813-1841, November, 1965.

3. F. J. Witt, “An Experimental P24 Mb/s Digital-Multiplexer-Demultiplexer Using Pulse Stuffing Synchronization,” Bell Sys. Tech. J., Vol. 44, pp. 1843-1885, November 1965.

4. J. W. Pan, U. S. Patent Pending.

5. A. A. Geigel and F. J. Witt, "Elastic Stores in High Speed Digital Systems," NEREM Rec., Vol. 6, 1964.

6. V. I. Johannes, "Multiplexing of Asynchronous Digital Signals Using Pulse Stuffing with Added-Bit Signaling," IEEE Trans. on Comm.Tech. Vol. COM-14, No. 5, pp. 562-568, October 1966.

7. J. S. Mayo, "An Approach to Digital System Networks," IEEE Trans. on Comm. Tech., Vol. COM-15, No. 2, pp. 307-310, April 1967.