The Problem of Synchronization of Noisy Video

R. J. McEliece
Communications Systems Research Section

The problem of acquiring and maintaining TV line synchronization with the use of a PN prefix on each scan line, in the presence of random noise is discussed, and the recommendation is made that on the Mariner Mercury-Venus flight, a PN of length 63 be used.

I. Introduction

The currently proposed method of detecting a loss of TV line synchronization on the *Mariner* Mercury-Venus flight is to prefix each line with a 31-bit PN sequence, and to assume that a loss of synchronization has occurred if the PN appears to have suffered 9 or more bit errors. The underlying assumption is, of course, that a loss of sync will result in a misframing of the PN sequence, so that what is supposed to be the PN has no close relationship to it at all. At high signal-to-noise ratios this method behaves satisfactorily, but at the low signal-to-noise ratios which could occur at Mercury, the PN sequence can actually suffer 9 or more random bit errors with a significant probability. To be precise: if the TV data is uncoded and if the signal-to-noise ratio drops to a value which gives a bit error probability of 0.1, then the probability that a correctly synched PN prefix will suffer 9 or more errors is 0.003. Since each TV picture contains 700 scan lines, each picture will suffer, on the average, 2.1 cases of this "spurious line drop." The 0.01 bit error probability case will also be discussed, since even 0.01 is higher than previous design points (5×10^{-3}) for synching.

It is the object of this note to point out that this problem of spurious line drop can be overcome in a simple fashion without sacrificing protection against a true loss of synchronization. In addition we shall propose a method of reacquiring line synchronization once it has been irretrievably lost, which is practical for real-time operation at 117 kbps, the highest bit rate which has been suggested for use on the mission.

II. Analysis

Let us first of all remark that, as stated, the problem admits of several very easy solutions. Two of these are:

(1) Relax the running threshold somewhat; e.g. allow up to 10 bit errors before becoming alarmed. At a bit error probability of 0.1, this would reduce the probability of spurious line drop to 0.0001.

(2) Wait until two lines in a row fail the PN test. At a bit error probability of 0.1, this would reduce the probability of spurious line drop to about 10^{-5}.

However, let us remember that once line sync has been irretrievably lost, we must face the prospect of reacquiring it. And at the extremely high bit rates which will be in use, the only practical method is to search all 31-bit segments of the incoming data until one of the segments looks "enough" like the PN sequence that we can be reasonably sure that we have located the start of a new

TV line. It is here that serious problems occur. For if our criterion for recognizing the PN sequence is too stringent, we may miss seeing it because of the noise in the channel. Conversely, if the criterion is not stringent enough, among the 7056 possible frameup of a TV line a 31-bit sequence which is not the PN sequence might pass the test. We call these two alternatives *failure to detect* and *false detection*, and have computed Table 1 for prefixes of lengths 31, 47, and 63.

In order to compute Table 1 we made the assumption that a TV line was a sequence of 7056 0's and 1's chosen independently and with probability ½. The parameter T represents the largest acceptable number of errors in a PN sequence. It is easy to see that with these assumptions the probability of false detection is not dependent on the bit error probability. On the other hand, we see that the probability of failure to detect the PN is highly sensitive to the BEP.

We draw several conclusions from Table 1.

(1) For $N = 31$, at $P = 0.1$ the best algorithm for reacquiring sync once it has been lost is to look for a 31-bit sequence which looks like the PN with up to 4 errors allowed. This procedure will relocate sync about 75% of the time within one line of the sync loss. However, 12% of the time the algorithm will lock onto the wrong synchronization, and 20% of the time it will make no decision at all during the first line. This is unacceptable.

(2) For $N = 31$ and $P = 0.01$ we should choose T to be 2, and then sync will be reacquired within one line of its loss more than 99% of the time. Notice, however, that if we keep $T = 4$, we get a 12% rate of false detection. Incidently, $N = 31$ is used on the high-rate telemetry system with a bit error probability of 5×10^{-3}, and the length 31 need not increase for 0.01 operation.

(3) Thus, to get the best performance at $N = 31$, we would change T as the signal-to-noise ratio changes, and in the worst case ($p = 0.1$) we will fail, 25% of the time, to correctly re-establish sync.

(4) If the prefix is longer than 31, however, performance can be much improved: at $N = 47$ we get a worst-case failure rate of about 5%; at $P = 0.01$ we get a failure rate of 10^{-4} or less; at $N = 63$, a worst-case rate of less than 0.01; and at $P = 0.01$ a failure rate of $< 10^{-5}$. Notice that at $N = 63$, if we take $T = 12$, we get a failure rate of less than 1% at any BEP which is 0.1 or less.

III. Conclusion

We conclude that the problem of spurious line drop-out is not serious and can easily be eliminated with almost no change in existing procedures. However, we also conclude that the problem of reacquiring line sync once it has been lost presents much greater difficulties, especially at high bit rates (which prohibit sophisticated search schemes) and low signal-to-noise ratios (which cause the PN prefix to be maimed somewhat). In particular we have shown that the best algorithm for finding line sync in the first line will fail 20% of the time, and 12% of the time it will produce false synchronization. This is not adequate performance. However, a PN prefix of length 63 can be used to reduce these probabilities to 0.008 and 0.002 at a BEP of 0.1, and this is acceptable. Thus we recommend this prefix.

Table 1. Probabilities of false detection and failure to detect for algorithm which looks for a prefix of length N and allows $\leq T$ errors (uncoded TV line of 7056 bits)

N	T	Probability of false detect	Probability of failure to detect PN BER = 0.1	Probability of failure to detect PN BER = 0.01
31	1	0.0001	0.830	0.03
	2	0.002	0.611	0.003
	3	0.017	0.376	0.0003
	4	0.12	0.193	10^{-5}
	5	0.70	0.083	10^{-6}
	6	1.0	0.03	10^{-7}
	7	1	0.009	10^{-7}
	8	1	0.003	10^{-7}
47	5	0.0001	0.328	10^{-5}
	6	0.0006	0.186	10^{-6}
	7	0.004	0.093	10^{-8}
	8	0.02	0.041	10^{-9}
	9	0.09	0.016	$<10^{-10}$
	10	0.35	0.005	$<10^{-10}$
	11	1.0	0.001	$<10^{-10}$
63	8	$<10^{-5}$	0.175	$<10^{-8}$
	9	2×10^{-5}	0.094	$<10^{-9}$
	10	0.0001	0.046	$<10^{-9}$
	11	0.0006	0.021	$<10^{-9}$
	12	0.002	0.008	$<10^{-10}$
	13	0.01	0.003	$<10^{-10}$
	14	0.04	0.001	$<10^{-10}$
	15	0.13	0.0004	$<10^{-10}$

N = PN length.

T = number of errors permitted.

Probably of false detection is independent of bit error rate, but probability of failure to detect PN is not.

These probabilities are also independent of which N-bit prefix is selected. A PN sequence is recommended because it provides maximum protection against sync "slips" of just a few bits.