Node and Frame Synchronization in the Big Viterbi Decoder

J. I. Statman, B. Siev, and J. Rabkin
Communications Systems Research Section

The Big Viterbi Decoder (BVD), currently under development for the Deep Space Network (DSN), uses a single module to perform three functions: node synchronization, frame-marker synchronization, and adjustment to nontransparent codes. In the present DSN, the first function is executed by the Maximum-Likelihood Convolutional Decoder, and the second and third functions reside in a Frame-Synchronization Subassembly. The BVD approach combines these functions to increase equipment effectiveness and capabilities.

I. Introduction

The Big Viterbi Decoder (BVD) is a programmable decoder for convolutional codes with constraint length K up to 15 and code rate $1/n$, where $n = 2, \ldots, 6$ [1]. It can decode virtually all codes within these parameters, and operates at 1.1 Mbit/sec, meeting all foreseeable Deep Space Network (DSN) support requirements.

As part of the BVD, the synchronization module performs the following three key functions (for brevity, "sync" is used for "synchronization"):

(1) Acquire and maintain node sync; i.e., generate a decoded bit clock from the received symbol clock, with a proper offset. For example, when a rate 1/6 code is used, the decoded bit clock is six times slower than the encoded symbol clock. The proper phasing of the two clocks is crucial to the operation of the decoder and must be established before correct decoding starts, and monitored thereafter.

(2) Acquire and maintain frame sync; i.e., detect the presence of frame markers in the data and pass the information to the Reed-Solomon (RS) decoder. Frame markers are inserted in the data stream in the DSN concatenated coding format to allow follow-on deinterleaving and RS decoding. This function is presently performed by the Frame-Synchronization Subassembly (FSS).

(3) Detect and correct inverted symbol polarity for nontransparent codes. The current (7,1/2) and (7,1/3) convolutional codes are transparent; i.e., if all the input symbols are inverted, the decoder operates correctly except that the resulting decoded bits are also inverted. The possible inversion of the input symbols is a by-product of the operation of the telemetry receiver and cannot be easily overcome. Transparent codes allow the decoder to operate even with such an inversion, delaying compensation for this inversion to postprocessing. A nontransparent code, such as that used on Galileo, requires real-time compensation for this inversion.

Performance of these functions can be improved when they are executed jointly, e.g., data from frame-sync assist in determining symbol inversion. The remainder of this article describes how these operations are integrated in the BVD.

II. Organization of the Sync Module

The sync module is organized as shown in Fig. 1. Three separate algorithms are implemented: the first measures the rate at which the decoder-accumulated metrics are growing (metric-growth rate), the second searches for the frame marker in the decoded bit stream (bit correlation), and the third searches for an encoded version of the frame marker in the encoded input stream (symbol correlation). The bit-correlation and symbol-correlation algorithms can be used to perform any of the module's three functions. The metric-growth-rate algorithm determines only node sync or, in the case of nontransparent codes, symbol inversion. Table 1 shows the relationships between the algorithms and the functions.

Selection of the right mix of algorithms and their respective parameters depends on bit signal-to-noise ratio (E_b/N_0), desired acquisition time (T_{acq}), and required probabilities for detection and false alarm during acquisition and monitoring. The following is a short description of the three separate algorithms and the method by which they are combined.

A. Metric-Growth-Rate Method

The metric-growth-rate method is based on the fact that, for practical E_b/N_0, the accumulated metric-growth rate is lower when the decoder operates with correct node sync than when the decoder is out of node sync [2]. A difference in metric-growth rate also occurs, despite correct node sync, if the input symbols are inverted and the code is nontransparent. Metric-growth rate is measured indirectly, through the renormalization signal, as shown in Fig. 2. The renormalization (renorm) signal is generated in the BVD whenever the accumulated metrics are at risk of overflowing their accumulators and hence must be reduced. The frequency of this signal is proportional to the average rate of metric growth.

Two similar sets of circuitry are available: the first set measures the number of decoded bits between renorm pulses, resulting in a count (e.g., 1500 bits/renorm); the second set measures "renorm rate" by dividing the number of renorm pulses by the total number of decoded bits during a specified measurement interval, resulting in a fraction (e.g., 3 percent renorms per decoded bit). In either case, the measured value is compared to a threshold to determine whether the decoder is in node sync and, for a nontranparent code, that the symbols are not inverted. Simulations indicate that the first measurement is preferred because it provides a reliable result in a shorter time. The second measurement is subject to additional uncertainty due to the fractional renormalization cycles included in the specified measurement interval.

B. Bit-Correlation Method

The bit-correlation method can be applied when the original frame contains known data (e.g., frame markers). In many DSN missions,[1,2] as well as in the Consultative Committee for Space Data Systems (CCSDS) standard data, such markers are present. The node-sync hypothesis is tested through checking for the presence or absence of the frame marker, because when the decoder is not in node sync, the decoded output is unrelated to the frame marker. Operation of the bit correlator is shown in Fig. 3. For each decoded bit time, the correlation between the decoded-bit sequence and the frame-marker sequence is evaluated. The resulting "number of agreements" is compared against two programmable thresholds: for nontransparent codes, a high correlation value (above T_H) indicates both positive node sync and correct symbol polarity; for transparent codes, both high (above T_H) and low (below T_L) correlation values indicate node sync. For the latter, high correlation indicates correct symbol polarity and low correlation indicates inverted symbol polarity.

C. Symbol-Correlation Method

The symbol-correlation method is a variation on the bit-correlation method. Instead of examining the decoded bit stream for the presence of a frame marker, the encoded symbol stream is tested for an encoded version of the frame marker. A detailed analysis is provided in [3].

III. Some Observations

Each of the three algorithms described here has advantages and disadvantages. The approach taken in the BVD is to use a combination of the three methods so that overall system performance is optimized.

A. Requirements

The required BVD performance[3] is derived from the particular environment of deep-space missions. The parameters of interest are total acquisition time including

[1] *DSCC Subsystem Functional Requirements, Telemetry Subsystem, 1992–1995*, TDA Document 824-35 (internal document), Jet Propulsion Laboratory, Pasadena, California, May 1, 1989.

[2] *DSN System Functional Requirements, Telemetry System, 1992–1995*, TDA Document 821-35 (internal document), Jet Propulsion Laboratory, Pasadena, California, April 15, 1990.

[3] *DSCC Subsystem Functional Requirements, Telemetry Subsystem, 1992–1995*, op. cit.

confirmation (T_{acq}), the probability of detection, and the probability of false alarm. In general, the DSN desires extremely low probability of false loss of sync during a track, while it is willing to compromise the requirement for a short initial acquisition time. Because of the existing partitioning of the telemetry subsystem, the BVD has not been levied with frame-sync and symbol-inversion requirements, hence, the discussion here is focused on the node-sync requirements. It is important to remember that these other functions can be done in the BVD (in fact, symbol inversion must be done in the BVD for nontransparent codes) when appropriate interfaces are provided.

Initial acquisition consists of a search period and a confirmation period. It is usually conducted when the antennas are at low elevation; hence, the recovered symbol signal-to-noise ratio (SNR) is low and reasonable data loss is expected. A typical DSN requirement (derived from the Maximum-Likelihood Convolutional Decoder [MCD] Assembly requirement) is for 99-percent probability of acquisition within 5000 bits, for $E_b/N_0 > 0.5$ dB, using the NASA standard (7,1/2) convolutional code. (For Galileo data rates of 134.4 kbit/sec and 115 kbit/sec, this is less than 0.05 sec.) Following this search, acquisition must be confirmed to a probability significantly higher than 99 percent to assure correct node sync. This confirmation process is not specifically defined.[4] To fulfill these requirements, a combination of the three algorithms will be needed.

During tracking, the DSN requirement is for no false loss of node sync for 24 hours whenever $E_b/N_0 > 0.5$ dB. The edict here is clear: a false assertion that node sync is lost during a track is a severe offense causing loss of valuable data and should be avoided. However, the flip side of the coin is that if node sync is truly lost, the BVD must detect it and correct for it as soon as possible. The ability of the BVD to look at three (almost) independent node-sync methods allows it to perform this delicate balance in monitoring node sync during tracking.

B. Trade-offs

1. Symbol-correlation versus bit-correlation methods. The BVD allows frame-marker correlation to be performed both in the symbol and bit domains. In the symbol domain, if the frame marker length is N_{sync} and the constraint length is K, then only $N_{sync} - K + 1$ bits of frame marker are usable in the correlation. The encoded symbols corresponding to the other $K - 1$ bits are dependent on the previous contents of the encoder shift register, unrelated to the frame marker. For example, if $N_{sync} = 32$ and $K = 15$, 14 of the 32 bits of the frame marker are useless. This is a disadvantage compared to the bit-correlation approach in which all N_{sync} bits can be used in the correlation. To recover enough SNR for the correlation, integration over two or more frames may be needed at low symbol SNR. The BVD hardware allows this integration.

The symbol-correlation method is the only one allowing detection of the frame marker prior to BVD operation. This allows several benefits:

(1) The frame marker can be detected without testing for the possible phases of node sync and polarities of symbols ($2n$ possibilities for a nontransparent code), resulting in a shorter T_{acq}.

(2) Symbol inversion can be detected immediately for both transparent and nontransparent codes.

(3) Frame-marker detection capability is provided even for data that are not convolutionally encoded.

These benefits can easily outweigh the longer integration time used in this method, resulting in a minimal overall acquisition time.

2. Correlation versus metric-growth-rate methods. Typical behavior of T_{acq}, measured in data bits, versus E_b/N_0 is shown in Fig. 4. The correlation-based approaches require minimum fixed acquisition time (i.e., at least one full frame) even at high E_b/N_0. Furthermore, acquisition time is quantized to an integer number of frames. As E_b/N_0 is reduced, acquisition time does not vary until integration over an additional frame is needed. A rule of thumb is that, since quality of acquisition depends on total SNR, integration time (and acquisition time) approximately doubles for each 3-dB decline in E_b/N_0. In contrast, acquisition based on metric-growth rate can be accomplished very quickly (e.g., with several hundred bits [2]) for high SNR, but integration time grows rapidly as E_b/N_0 declines. Since the BVD measures metric-growth rate through renormalization signals, the acquisition time is quantized in steps corresponding to the distance between renormalization signals. The transition from reliance on correlation methods to reliance on metric-growth-rate methods depends on the specific convolutional code and required probabilities of detection and false alarm.

3. Node-sync thresholds. Three hypotheses are tested:

(1) Is node sync detected/initialized, conditioned on "cold start?"

[4] Ibid.

(2) Is node sync confirmed, conditioned on being detected/initialized?

(3) Is node sync still correct, conditioned on node sync being confirmed?

All three methods employ threshold values against which the node-sync hypotheses are tested. These thresholds depend on E_b/N_0, the specific convolutional code and frame marker, and the desired probabilities of detection and false alarm. Tabulated values for these thresholds (especially in the metric-growth method) will be developed initially from simulations, but can be computed later on a breadboard BVD (simulations require massive computational power).

4. Overall algorithm for node-sync function. The overall node-sync algorithm integrates the three methods in a manner that is programmable by the operator. Precise definition of the default algorithm can be changed prior to BVD deployment in the DSN. Nevertheless, a candidate algorithm can be expressed as follows:

(1) The detection/initialization hypothesis will be "locked" if testing of a single method (the one that is most suitable based on E_b/N_0 and the presence or absence of frame markers) is "locked." This will be accomplished using relatively low thresholds corresponding to the 99-percent probability of correct initialization.

(2) Confirmation will be declared "locked" if testing with all three methods results in detection of node sync. The thresholds used for this step will be more restrictive, corresponding to a probability of correct detection of, for example, 99.99 percent. The confirmation hypothesis may also be evaluated by checking for repeats of "locked" results in the individual methods.

(3) The node-sync-tracking hypothesis will be declared "no-lock" only if all three methods provide a "no-lock" result.

IV. Conclusions

The synchronization module of the Big Viterbi Decoder that performs node sync, frame-marker sync, and adjustment of symbol inversion has been described. The various methods used to accomplish these functions, their advantages and disadvantages, and their overall integration in the module were presented. Numerical results need to be obtained through simulations and actual BVD testing.

References

[1] J. Statman, G. Zimmerman, F. Pollara, and O. Collins, "A Long Constraint Length Viterbi Decoder for the DSN," *TDA Progress Report 42-95*, vol. July–September 1988, Jet Propulsion Laboratory, Pasadena, California, pp. 134–142, November 15, 1988.

[2] U. Cheng, "Node Synchronization of Viterbi Decoders Using State Metrics," *TDA Progress Report 42-94*, vol. April–June 1988, Jet Propulsion Laboratory, Pasadena, California, pp. 201–209, August 15, 1988.

[3] K.-M. Cheung and J. I. Statman, "A Node and Frame Synchronization Scheme in the Big Viterbi Decoder Based on Channel Symbol Measurements," *TDA Progress Report 42-103* (this issue), vol. July–September 1990, Jet Propulsion Laboratory, Pasadena, California, pp. 203–214, November 15, 1990.

Table 1. Sync module relationships

Code	Metric-growth rate	Bit correlation	Symbol correlation
Transparent			
Node sync	Yes	Yes	Yes
Frame sync	No	Yes	Yes
Symbol inversion	No	Yes	Yes
Nontransparent			
Node sync	Yes	Yes	Yes
Frame sync	No	Yes	Yes
Symbol inversion	Yes	Yes	Yes

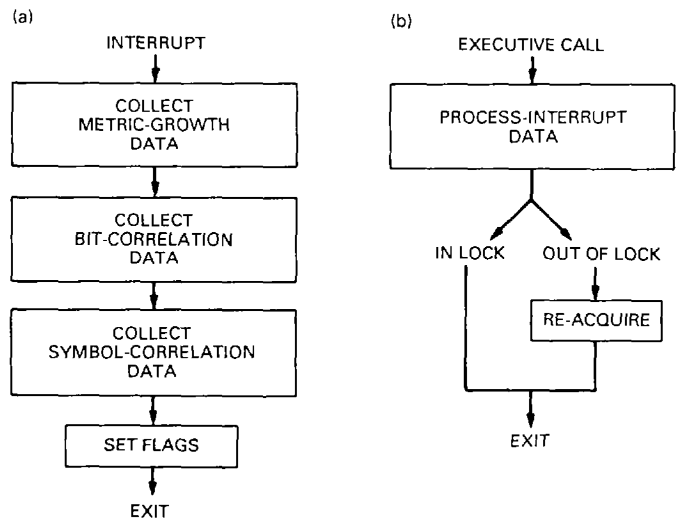

Fig. 1. The sync module: (a) software interrupts, and (b) software regular processing.

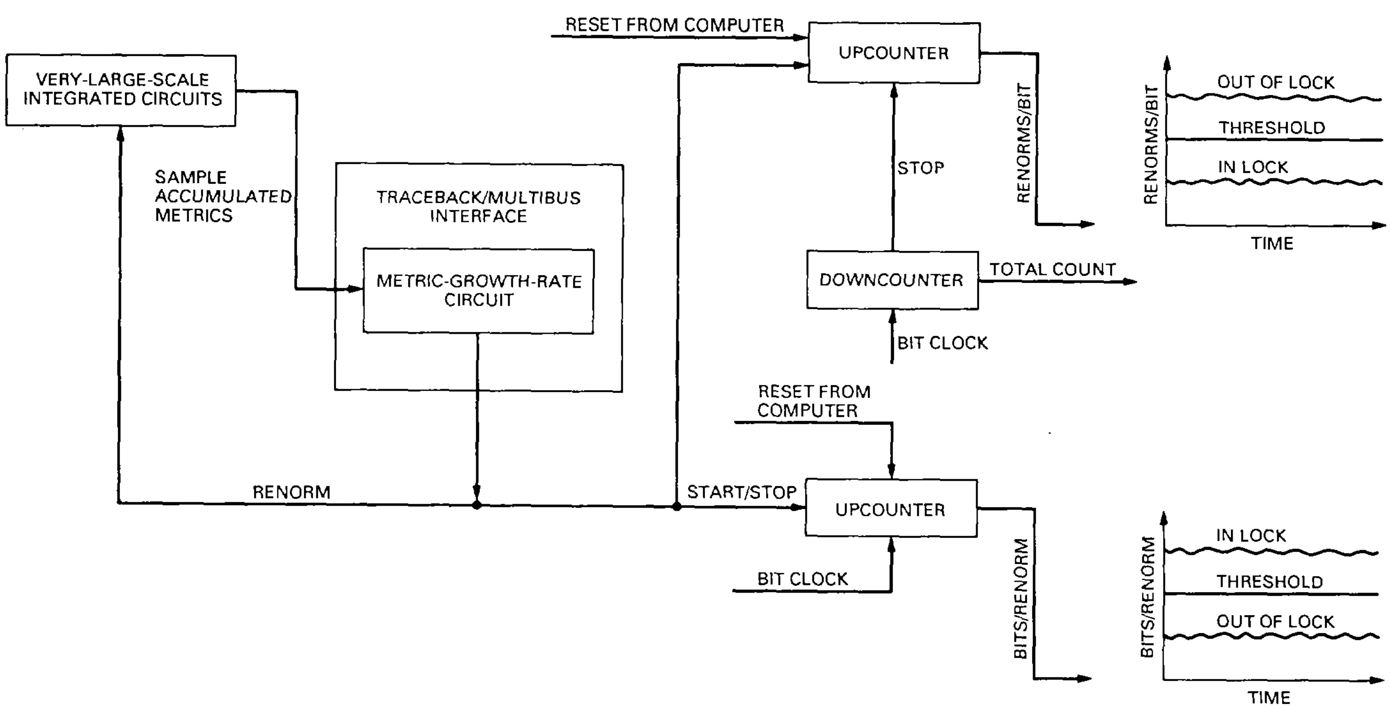

Fig. 2. The circuit for metric-growth rate.

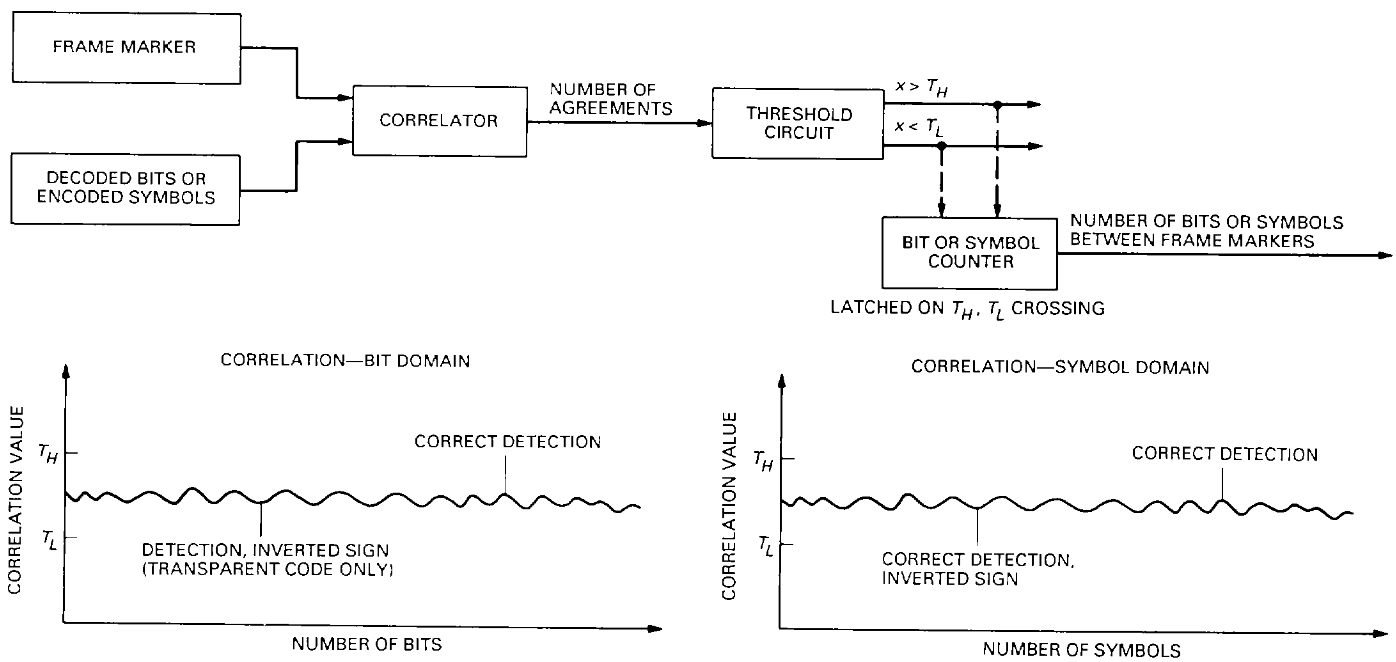

Fig. 3. The circuit for frame-marker correlation.

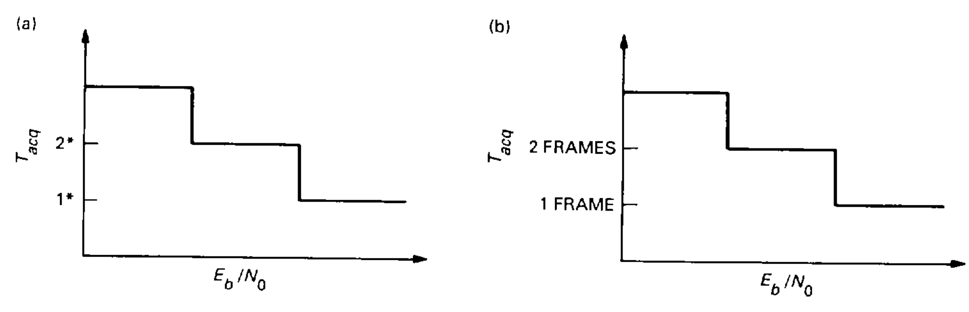

Fig. 4. Behavior of algorithms: (a) metric-growth-rate method, and (b) correlation method.