Decoder Synchronization For Deep Space Missions

Joseph]. Statman
Kar-Ming Cheung
'J'octet H. Chauvin
lay Rabkin
Mignon L. Belongie

Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California, 91109 USA
Tel: (818) 3S4-2926 FAX: (818) 354-5994

Abstract

The Consultative Committee for Space. Data Standards (CCSDS) recommends that space communication links employ a concatenated err or-correcting channel-coding system in which the inner code is a convolutional (7,1/2) code. and the outer code is a **(2.55,223)** Reed-Solcmml~ code. The traditional implementation is to perform the nodc synchronization for the Viterbi decoder and the frame synchroni zation for the Reed-Solomon decoder as separate, sequential operations, This article discusses a unified synchronization technique that is required for deep space missions that have data rates and signal-to-noise ratios (SNRs) that arc extremely low. This technique combines frame synchronization in the bit and symbol domains and traditional accumulated-metric growth techniques to establish a joint frame and nodc synchronization. A variation on this technique is used for the Galileo spacecraft on its Jupiter-bound mission.

I. Introduction

The traditional approach to decoding the channel crlor-cone.cling, coding in the space. communication links [1] is for the implementation to follow the CCSDS functional model [2] shown in Figure 1; i.e., establish a concatenated decoder consisting, of two distinct stages: a Viterbi decoder and a Reed-Solomon decoder, with no feedback between the two stages. Each of the two decoders requires appropriate synchronization: the Viterbi decoder requires node synchronization (the grouping of n-tuplets of soft symbols that correspond 10 a single information bit) and the Reed-Solomon decoder requires frame synchronization (the detection of the transport frame and the extraction of Reed-Solomon words). In most applications, data received pier to the accomplishment of synchronization is lost; however, as long as the symbol-signal-to-noise ratios (SSNR, *Es* / *No*) is relatively high, the synchronization time is short and the data loss is often ignored.

For deep Space communications environment, such an approach is often deficient. *Es* /No could be low, hence the synchronization time, measured in number of bits, is longer. As the data rate decreases, the synchronization time, fixed in terms of number of bits, can result in 10ss of a significant percent of total data, Also, the sequential nature of the synchronization process compounds the data loss. To speed the synchronization process and reduce the. data loss we introduce here a joint synchronization technique that is being applied in the ground support for the Galileo Deep Space Mission to Jupiter [3].

Section 2 introduces the core algorithms and the joint synchronization approach. Section 3 disc-asses the application of joint synchronization in sever al scenarios, and Section 4 presents two specific cases where the joint synchronization approach is applied.

11, Description of Synchronization Algorithms.

The structure of the general joint synchronization dc.coder is shown in Figure 2. It consists of the traditional series of processing functions, namely a Viterbi dccodcr, a de-interleaver, and a Reed-Solomon decoder, preceded by a soft symbol buffer. The processing functions arc controlled by a joint synchronizat ion function, which in turn relics on several core synchronization algorithms. In this article, we select three such core

synchronization algorithms: a frame-marker correlator in the symbol domain, a frame-marker correlator in the bit domain, and an accumulated-metric growlll-rate indicator. It is worth noting that decoders often contain other synchronization indicators that can be integrated into a joint synchronizer using techniques similar to those described below.

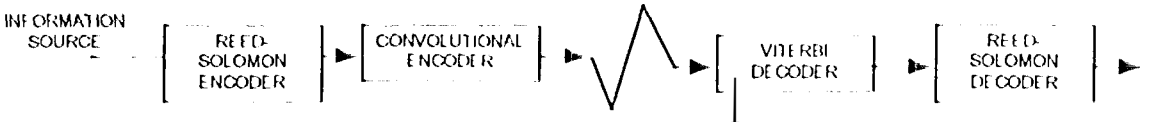

Figure 1, Erlor-correcting Encxrcic.rfllc.coder Traditional Configuration

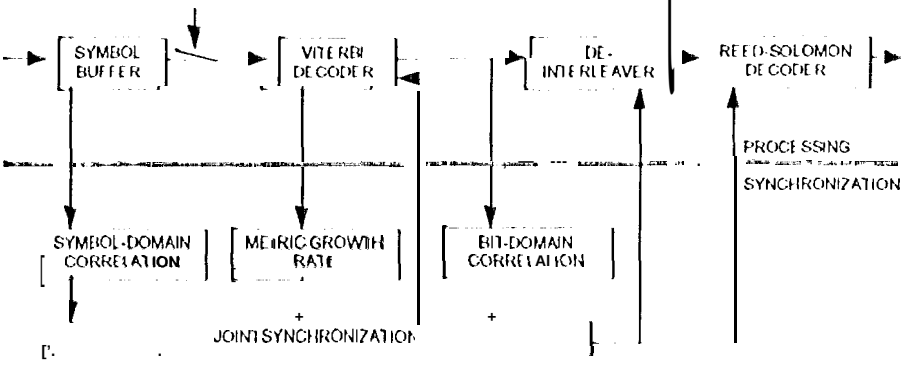

Figure 2. Joint Synchronization Decoder Structure

11,1 Bit-Domain Correlation

This algorithm examines the Viterbi-decoded bits at the output of the Viterbi decoder for presence of the frame market pattern. When the pattern is detected, it indicates that the node synchronization hypothesis is "true" and, by definition, frame synchronization was accomplished.

Let the transport frame be of length M, including a frame marker of length N al the beginning of each frame. Let the frame marker be $(d_i), \quad i = 0 \ldots N-1, \quad d_0(i) = \pm 1)$. Then, the transmitted data stream can be represented as

$$D(i) = \begin{cases} d_0(i \bmod M), & i \bmod M < h' \\ data, & i \bmod M \geq N \end{cases}$$

The bit-domain correlation algorithm computes the running correlation, $J_{bits}(k)$, between the frame marker and the received signal $s(i)$

$$J_{bits}(k) = \frac{1}{N} \sum_{i=0}^{h'-1} d_0(i) * s(i+k)$$

where $s(i)$ is $D(i)$ contaminated by the effects of transmission, reception, and decoding. The expected value of $J_{bits}(k)$ *is 1* when the received sequence is perfectly aligned with the frame marker. Itlsc.where., for a properly selected frame marker, its auto correlation properties assure that the expected value of $J_{bits}(k)$ is near zero. In practice, there arc two methods to implement the bit-domain correlation algorithm:

(a) Testing for threshold, where the value of k is compare.d against a threshold. When the value of $J_{bits}(k)$ exceeds the threshold for the first time at k_0, the hypothesis that $s(i + k_0)$ is the beginning of the frame marker is declared true.

(b) Testing for maximum, where the value $J_{bits}(k)$ is maximized over all $0 < k < M$, regardless of a threshold. I .et k_0 be the location where $J_{bits}(k)$ reaches its highest value over the search range, then the hypothesis that $s(i + k_0)$ is declared true.

1 n the special interest of *transparent* convolutional codes (i.e., if the soft symbols arc inverted, the Viterbi-decoded bits arc inverted as

well) $J_{bits}(k)$ may take the values of + 1 for "correct symbol phase correlat ion" or - 1 for "inverted symbol phase correlation" indicating that the Viterbi decoder is synchronized but the soft symbols arc inverted and must be returned to the correct phase (i.e.., reinverted) prior to de-interleaving. For transparent codes, implementing the bit-domain correlation requires searching for both a maximum and a minimum of $J_{bits}(k)$, or alternatively employing high and low thresholds.

The bit-correlation approach is powerful and usually highly reliable. 1 1 s main disadvantage is that it requites hypothesis testing at al I possible offsets of the soft symbol n-tuplets, as well as the correct and inverted symbol phases (only for nontransparent convolutional code). '1'bus, for a (15,1/6) non-transparent convolutional code, a large number of attempts (12 attempts in the worst case, 6 attempts on the average) are required before synchronization is accomplished.

II.2 Symbol-Domain Correlation

This algorithm examines the soft symbols prior to the Viterbi decoder for presence of encoded version of the frame marker pattern. When the pattern is detected, it provides both the node and frame synchronization prior to any decoding operation. in this case the symbol correlation function is

$$J_{sym}(k) = \text{'h. \& 1)} \sum_{i=K-1}^{N} \frac{1}{n} Fe(d_0(i)) * S(i+k)$$

where $Fe(d_0(i))$ is the n-tuplet of soft symbols corresponding to a single bit of the frame. marker, $S(i+k)$ is an n-tuplet of received symbols, and the operatotion on the two n-tuplets is a dot-product. The main disadvantage of this algorithm is that correlation can be performed only over part of the frame marker. For a frame marker of N bits and convolutional code of length K, only the symbols corresponding to the last N- (A' - 1) bits are known in the symbol domain - the symbols that correspond to the first K - 1 bits of the frame marker arc corrupted by the unknown previous contents of the encoder. Depending cm N, K, and Es / No, the partial correlation may degrade the correlation SNR sufficiently to make synchronization difficult.

T'd assess the performance of synchronization using symbol-domain correlation, let us compute the probability of false detection. Let the soft symbol be modeled as having a value of $\pm m + n_{sym}$ where $\pm m$ has equal probability of being $+m$ and $-m$ and n_{sym} is $N(0, \sigma_{sym})$[1]. Let us assure that the frame marker has been selected such that its auto correlation is near ideal, i.e. no significant secondary correlation peaks exist. Let us further assume that the correlation between the frame marker and a noiseless set of received symbols contains 110 significant secondary peaks, Then, when the received symbols arc not aligned with the frame marker, $J_{sym}(k)$ can be modeled as $N(0, O)$,

$$\sigma = \sigma_{sym} / \sqrt{N - (K - 1)},$$

while when the received symbols arc aligned with the frame marker, $J_{sym}(k)$, is modeled as $N(m, o)$. The probability of selecting an incorrect peak is given by

$$P_{FD} = \frac{1}{2\pi\sigma^2} \int_{-\infty}^{+\infty} e^{-\frac{x^2}{2\sigma^2}} \left(\int_{-\infty}^{x} e^{-\frac{(y-\mu)^2}{2\sigma^2}} dy \right) dx$$

Figure 3 shows this probability as a function of the conflation SNR, $20\log(m/o)$. As an example, let N = 32 bits, K = 15, and the minimal bit SNR (dictated by the required bit error rate) E_B / No= *0.7* dB. Then the SNR at the output of the correlator is

$$0.7 + 10 * \log_{10}(32 - (15 - 1)) = 13.2\,dB$$

and the probability of selecting a false correlation peak is (from the plot) --{). . For a frame size of 16,384 symbols, false detection is 13 times more likely than correct detection of the frame marker! Note that due to the sharp slope of the curve, increasing N ICI 64 will increase the SNR sufficiently to assort correct detection with high probability.

1[.3 Accumulated Metric Growth Rate

This algorithm examines the accumulated metric al all the Viterbi decoder states, indicating the level of "mismatch" between the

1 Zero-mean, normally-distributed random variable with standard deviation of σ_{sym}

received soft symbol stream and the bit stream associated with the specific state. Even though the accumulated metric varies from slate to state, its peak-to-peak variation is bounded by

(constraint length - 1) * (max branch metric).

Because of this bound, implementors often monitor the growth rate of a single selected accumulated metric where at high *Eb* / No there is a clear distinction between in-node-synchronization and out-of-node-synchronization conditions. The distinction between the two conditions becomes more blurry as *Eb* / No decreases; the thresholds for detecting the in-lock and out-of-lock hypotheses must be chosen carefully 10 meet the probability-of-detection and false alarm requirements. Unfortunately, threshold selection must be accomplished empirically, as the growth rate measurement dots not lend itself 10 analytic expressions.

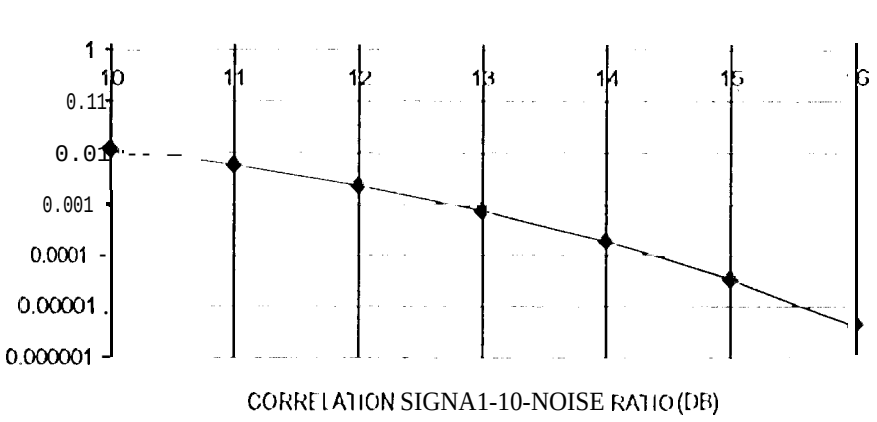

Figure 3. Probability of False Detection vs Correlation SNR

III. Unified Synchronization

The objective of the unified synchronization approach below is to realize the attractive benefits of the symbol-domain correlation; namely, accomplish node and frame synchronization prior to the Viterbi dc.coder, while mitigating the algorithm's degraded performance at low SNR.

CASE 1 - Independent flames, minimal restriction on latency.

1.et us first observe that many communications links consist of transport flames that are independently encoded as a result of the fact that the frame sync marker is longer than the constraint length of the convolutional code, there.fore serving as an effective bat rier between the frames, resetting the encoder to a known state. 1.et us also assume that there arc minimal restrictions on latency[2]. In this case, a simple decoder architecture is possible[3], shown in Figure 4.

In this architecture, no decoding, is initiated until the input stream has been separated into frames, using f[alnc-marker correlation in the symbol domain. Once the frame detection is verified, the frames are processed independently; the function marked "Viterbi and Reed-Solomon decoders" is replicated as many times as needed to meet the required data rate and latency. This architecture is especially suitable for implementation with parallel processors and for case.s where a "pool" of resources is available to cover the needs of many users with diverse needs.

CASE 2 - Independent frames, restriction on latency.

2 Latency is the time delay between the signal arrival at the antenna and the time a fully decoded transport frame is available at the output of the decoder

3 E. Greenberg, JPL Internal Memorandum 3171-93-20, Dated April 6, 1993

The process of frame detection often requires perfor ming the correlation over multiple frames and bridging "gaps" that eliminate frame markers, therefore introducing a substantial, and sometimes unacceptable, latency. in this case, the architecture can be modified as shown in Figure 5, utilizing the symbol-domain frame-markc.r correlator only for tentative frame identification. Decoding of frames starts immediately following this tentat ive detection. 1 lowever, results from the other synchronization algorithms arc used to verify the synchronization and arc fed back to allow correction of the synchronization, if needed,

The extent of input buffering and feedback depends on trade-off between the latency requirement and the implementation restrictions on data rate and available storage.

CASE 3 - Non-independent frames or tight latency requirements.

This case occurs either when the frame marker is shorter than the convolutional code constraint length (hence the. encoding is not independent from frame to frame) or when the tight latency requirement dictates that Viterbi decoding must be initiated even prior to frame synchronization. The resulting decoder architecture is shown in Figure 6 - the Viterbi decoder relics on its internal measures, e.g. acctlll~lllalcd-ll~ ctric-p,l o\\ll]-late, to achieve node synchronization, but receives feedback from the symbol-domain and bit-domain frame-marker correlators. In this architecture multiple Viterbi decoders could be employed to expedite the. detection of the node synchronization over multiple symbol phase offsets.

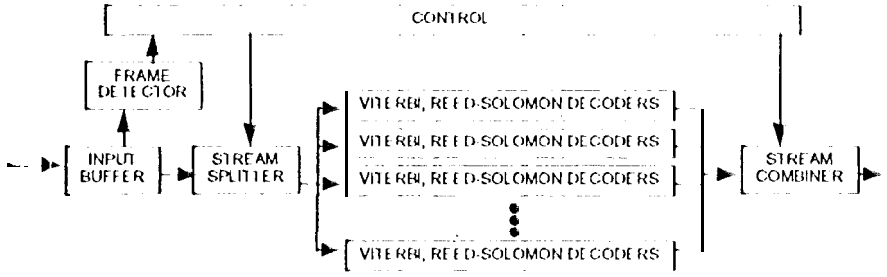

Figure 4. IJc.coder Architecture without Synchronization Feedback

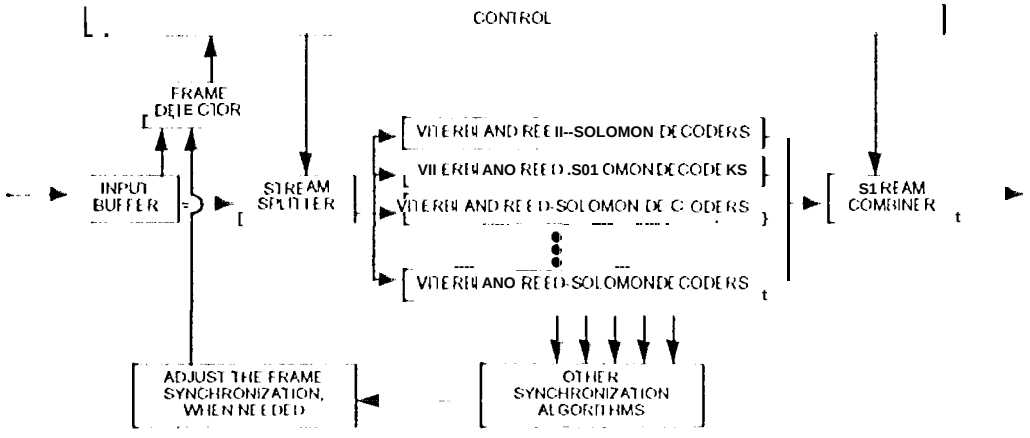

Figure 5. Decoder Architecture , with Feedback for Synchronization Correction

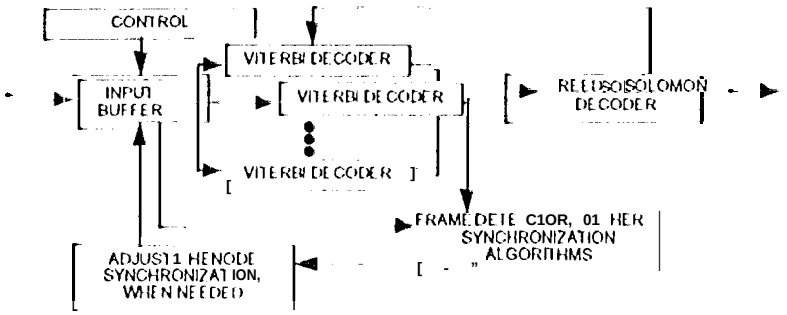

Figure 6. Decoder Architecture, Viterbi Decoder is Self-Synchronizing

IV. Examples

The joint synchronization approach has been applied in two decoder systems currently at JPL. The first decoder, denoted Maximum Likelihood Decoder III (MCD III) [4,5] is a fully-programmable, $(K,1/n)$ Viterbi decoder with $3 \le K \le 15$ and $2 \le n \le 6$. It is implemented in a fully-parallel architecture using 64 identical custom VLSI devices, capable of operating at bit rates of up to 1.1 Mbits/s. The MCD III incorporates all three synchronization algorithms described in Section ?, resulting in a flexible synchronization architecture.

A more recent example is the Feedback Concatenated Decoder (FCD) developed specifically for the Galileo mission [6]. This decoder is implemented in software on a 4-CPU SUN workstation. It is capable of performing (14,1/4) Viterbi and 4-redundancy (255,m) Reed-Solomon decoding as well as the associated re-decoding at 160 bits/s. As an integrated decoder, it is oriented toward symbol-domain frame synchronization, while attempting to minimize latency (case 2 above).

V. Conclusions

We have presented a method of performing joint frame and node synchronization for a concatenated decoder. This approach enables a design that shortens the acquisition time and allows for parallel implementation of the decoder resource-demanding tasks, overall improving the decoder efficiency.

Acknowledgment

The research reported in this paper was conducted at the Jet Propulsion Laboratory, California Institute of Technology, under contract with the National Aeronautics and Space Administration.

References

[1] S. Lin and D. Costello, "Error Control Coding," Prentice Hall Series on Computer Applications in Electrical Engineering, 1983.

[2] Consultative Committee on Space Data Standards, Document 101-0-11-3, (Blue book issue 3), May 1992.

[3] J. Statman, "Optimizing the Galileo Space Communication Link," this volume.

[4] J. Statman, G. Zimmerman, F. Pollara and O. Collins, "A Long Constraint Length VLSI Viterbi Decoder for the DSN," TDA Progress Report 42-95, Jet Propulsion Laboratory, Pasadena, California, pp. 186-200, February 15, 1992.

[5] K. M. Cheung, L. Swanson and S. Arnold, "Node Synchronization Schemes for the Big Viterbi Decoder," TDA Progress Report 42-108, Jet Propulsion Laboratory, Pasadena, California, pp. 134-142, December 15, 1988.

[6] T. Chauvin and K. M. Cheung, "A Parallel Viterbi Decoder for Shared Memory Multiprocessor Architecture," 3rd SIAM Conf. on Linear Algebra in Signals, Systems, and Control, Seattle, WA. August 1993.