Oversampled M-sequences for Joint Data and Bit Epoch Estimation in DSSS Transmissions

Daniele Borio[1], *Member, IEEE*

***Abstract*— The Maximum Likelihood (ML) estimator for the bit synchronization epoch and data bit values in Direct-Sequence Spread-Spectrum (DSSS) transmission consists of determining the binary sequence that maximizes the correlation with the recovered data samples. This requires the exhaustive test of all sequences generated by the different bit combinations and alignments, resulting in a computationally intensive process.**

In this letter, the properties of maximum length sequences (m-sequences) are exploited for testing all the bit combinations and alignments in parallel, leading to a computationally efficient implementation of the ML estimator. The case where the data bits are modulated by an overlay or secondary code is also considered and the proposed algorithm is generalized to include the effect of this additional modulation.

***Index Terms*—Bit Estimation, Bit Synchronization, Direct-Sequence Spread-Spectrum (DSSS), Fast Fourier Transform (FFT), Maximum Length Sequences, M-sequences, Secondary Codes.**

I. Introduction

In this paper, a new algorithm efficiently implementing the joint Maximum Likelihood (ML) estimator for data and bit epoch in burst Direct-Sequence Spread-Spectrum (DSSS) transmissions is proposed. The joint ML estimator for the bit boundaries and data values requires the generation of all possible binary sequences as a function of the different bit combinations and alignments. The ML estimates are then derived from the sequence that maximizes the correlation with the incoming samples [1]. This type of approach requires a computational complexity exponentially growing with the number of bits to be estimated. For this reason, iterative procedures, alleviating the computational load problem at the expenses of some performance loss, have been proposed [1].

In this paper, the ML estimator is implemented as the circular correlation of the incoming data samples with a combining sequence. The substrings of the combining sequence span all the possible bit combinations and alignments, allowing their parallel test. The circular correlation is efficiently evaluated using the Fast Fourier Transform (FFT) algorithm.

In this paper, modified m-sequences [2] are oversampled and interpolated in order to generate the combining sequence required for the parallel test of all the bit combinations and alignments. Two cases are considered: in the first one the data bits are unmodulated and samples belonging to the same symbol assume a constant value. This corresponds, for example, to the GPS L1 Coarse Acquisition (C/A) signal where each bit lasts 20 periods of the primary spreading sequence that modulates the transmitted data. In the second case, the data bits are further modulated by an overlay code as foreseen by several modernized GPS signals.

The remainder of this paper is organized as follows: Section II reviews the signal model and the joint ML estimator for data and bit epoch. In Section III, the generation of the combining sequence is detailed whereas Section IV presents the fast correlation algorithm implementing the ML estimator. The algorithm is demonstrated using live GPS signals in Section V. Finally Section VI draws some conclusions.

II. Signal and System Model

The signal received by a DSSS receiver is at first down-converted and the primary pseudo-noise (PN) sequence is removed by the despreading process. The samples obtained after despreasing can be modeled as

$$r[n] = s_d[n-\tau] + \eta[n] \quad (1)$$

where s_d is the sampled data sequence and $\eta[n]$ is a residual noise term. τ is the delay relative to the beginning of a new data bit and introduced by the communication channel. The data sequence is given by [3]

$$s_d[n] = \sum_{k=-\infty}^{\infty} d_k p[n-kL] \quad (2)$$

where $\bar{d} = [..., d_{-m}, ..., d_{-1}, d_0, d_1, ..., d_m, ...]$ is the transmitted bit sequence and $p[\cdot]$ is a waveform of duration equal to L samples. If the bits are unmodulated, $p[\cdot]$ is a rectangular window whereas, if an overlay code is present, $p[\cdot]$ models the effect of this secondary modulation.

$\eta[n]$ is assumed to be a zero-mean, white Gaussian sequence with variance equal to σ^2. σ^2 depends on the input carrier-to-noise density ratio, C/N_0, on front-end filtering and decimation strategy and on the coherent integration time adopted by the despreading process. In this paper, it is assumed that $r[n]$ has been normalized such that s_d has unitary amplitude. This choice does not affect the results reported herein since the power relationship between signal and noise are preserved by scaling.

The joint ML estimators uses $m \cdot L$ samples from (1) to determine the delay τ and the $m+1$ bits that can occur when m data periods are observed. The joint ML is given by [3]:

$$\left[\hat{\tau}, \hat{\bar{d}}\right] = \arg\max_{\tau, \bar{d}} \sum_{n=0}^{Lm-1} r[n] s_d\left[n-\tau\right] \quad (3)$$

i.e., $[\hat{\tau}, \hat{\bar{d}}]$ are obtained as the delay and bit sequence that make $s_d[n-\tau]$ maximize the correlation with the incoming signal

1) Department of Geomatics Engineering, University of Calgary. 2500 University Dr NW Calgary, Alberta T2N 1N4 Canada. Email: dborio@ucalgary.ca

$r[n]$. It is noted that the $L2^{m+1}$ sequences, $s_d[n-\tau]$, have to be generated for testing all the possible bit combinations and alignments. In order to reduce the number of sequences, the following property can be exploited. When the data sequence in (1) is observed over a finite duration, $m \cdot L$, it can be rewritten as

$$s_d[n-\tau] = d_0 \sum_{k=0}^{m} b_k p[n-kL-\tau] \tag{4}$$

where $b_i = d_0 d_i$ accounts for bit changes with respect to the first received bit. It is noted that $b_0 = d_0^2 = 1$. In this way, the joint estimator (3) can be evaluated in two steps. At first τ and the bit transitions $\bar{b} = [b_1, b_2, ..., b_m]$ are estimated as

$$\left[\hat{\tau}, \hat{b}\right] = \arg\max_{\tau, \bar{b}} \left| \sum_{n=0}^{Lm-1} r[n] s_d[n-\tau] \right| \tag{5}$$

then, the estimate of d_0 is obtained as

$$\hat{d}_0 = \text{sign}\left\{ \sum_{n=0}^{Lm-1} r[n] s_{\tilde{d}}[n-\hat{\tau}] \right\} \tag{6}$$

where $s_{\tilde{d}}[n-\hat{\tau}]$ is the sequence obtained for $d = \tilde{d} = [1, \hat{b}_1, \hat{b}_2, ...\hat{b}_m]$ and $\tau = \hat{\tau}$. In this way, the number of substrings is reduced by a factor 2.

In the next section, a way for generating a sequence whose substrings correspond to all the possible $s_d[n-\tau]$ in (5) is discussed.

It is noted that (5) can be adopted also in the case in which the receiver is not able to recover the signal phase and $\eta[n]$ is a complex Gaussian process. Under the assumption of data bits rotated by a constant phase, (5) is the joint ML estimator for the delay τ and the bit transitions $\bar{b}$. The operator, $|\cdot|$, in (5) has to be interpreted as the absolute value of a complex number.

III. Oversampled m-sequences

An m-sequence of $2^m - 1$ elements contains all the binary strings of length m with the exception of the sequence of all zeros [2]. By extending the longest subsequence of "1" by one element a modified m-sequence, y, is obtained. y contains 2^m binary substrings of length $m+1$. Moreover, if s is a substring of y of length $m+1$, then $\bar{s}$, the complementary sequence of s, is not a substring of the modified m-sequence [4]. The complementary sequence, $\bar{s}$, is obtained by negating all the elements of s, for example $s = [0, 0, 1, 1, 0, 1]$ implies $\bar{s} = [1, 1, 0, 0, 1, 0]$. An example of modified m-sequence for $m = 3$ is given by

$$y = [1, 1, 1, 1, 0, 1, 0, 0]. \tag{7}$$

It is noted that none of the subsequences in (7) has its complementary in y. For this reason, y can be used for testing all the possible bit combinations of $m+1$ elements. Since the transmitted bits are encoded in bipolar form, y has to be mapped into a new sequence, $y_b[n]$, according to the following relationship

$$y_b[n] = \sum_{i=0}^{2^m-1} (2y[i]-1)\delta[n-i] \tag{8}$$

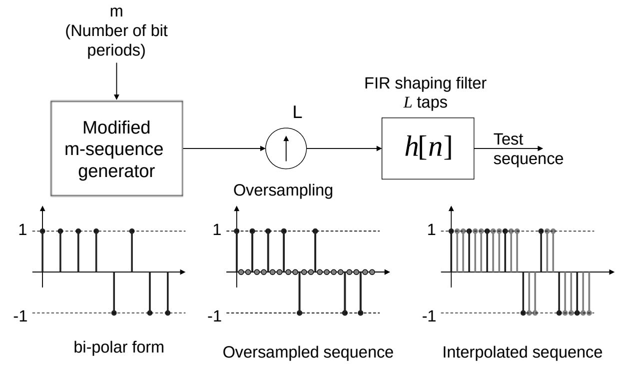

Fig. 1. Generation of the combining sequence for testing all the possible bit combinations and alignments.

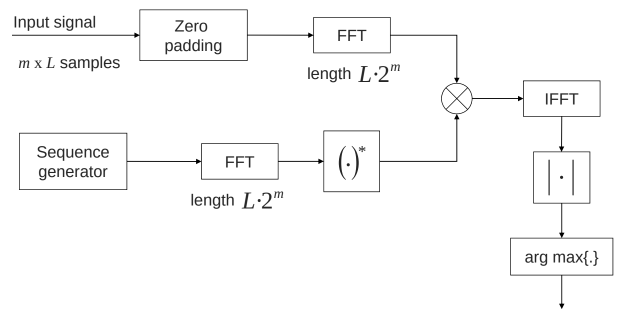

Fig. 2. Efficient implementation of the joint ML estimator using circular correlation. The circular correlation is implemented using the FFT algorithm.

where $y[i]$ denotes the ith element of y and $\delta[n]$ is the Kronecker delta. The combining sequence for the parallel test of all possible bit combinations and alignments is obtained by oversampling y_b by a factor L. The oversampled sequence is then interpolated using a Finite Impulse Response (FIR) filter:

$$y_h[n] = y_b[n/L] * h[n] \tag{9}$$

where the convention $y_b[n/L] = 0$ for $n/L \notin \mathbb{Z}$ is adopted. $h[n]$ is the impulse response of the shaping filter and $*$ denotes convolution. When the bits are unmodulated, $h[n]$ is given by the rectangular window

$$h[n] = \sum_{i=0}^{L-1} \delta[n-i]. \tag{10}$$

The process for the generation of the combining sequence is shown in Fig. 1 where the modified m-sequence (7) is generated in bipolar form, oversampled by a factor $L = 3$ and interpolated by the filter (10).

When the data bits are modulated by an overlay code, the shaping filter is given by

$$h[n] = \sum_{i=0}^{L-1} c_o[i]\delta[n-i] \tag{11}$$

where $c_o[i]$ is the ith coefficient of the overlay sequence.

IV. Fast Correlation Algorithm

$y_h[n]$ contains all the possible bit combinations without without complementary elements and can be used for the fast evaluation of (5). More specifically, all the possible correlations in (5) can be evaluated in parallel by evaluating the circular correlation of a zero-padded version of the input sequence $r[n]$ with $y_h[n]$. The circular correlation is efficiently implemented using the FFT algorithm as detailed in Fig. 2. The absolute values of the correlation in (5) are then obtained as

$$c_{corr}[k] = \left| \text{IFFT}\left[\text{FFT}\left(r[k], L2^m\right) \cdot \text{FFT}\left(y_h[k]\right)^* \right] \right| \tag{12}$$

where k is a composite index that accounts for both bit combinations and alignments. $(\cdot)^*$ denotes complex conjugate, whereas $\text{FFT}\left(r[k], L2^m\right)$ indicates that the FFT has been evaluated on a zero-padded version of $r[k]$ of $L2^m$ elements. The index

$$\hat{k} = \arg\max_k c_{corr}[k] \tag{13}$$

is then used for determining $\hat{\tau}$ and the estimate of the bit transitions:

$$\hat{\tau} = \hat{k} \mod L \tag{14}$$

$$\hat{b} = [1, a_0 y_b[\hat{s}], a_0 y_b[\hat{s}+1], ..., a_0 y_b[\hat{s}+m-1]] \tag{15}$$

where $\hat{s} = \left\lceil \frac{\hat{k}}{L} \right\rceil$ and $a_0 = y_b[\hat{s}-1]$. $\lceil \cdot \rceil$ is the ceiling operator. The normalization by a_0 is required for guaranteeing the condition $b_0 = 1$. More specifically, the subsequences of $y_b[n]$ can start either with 1 or -1 and the normalization by a_0 enforces the estimate of b_0 to be equal to 1. d_0 is then estimated using (6), where $s_{\tilde{d}}[n - \hat{\tau}]$ is reconstructed from (14) and (15).

V. Sample Results

In this section, some sample results obtained by processing live GPS L1 C/A signals are reported. The GPS L1 C/A signal is modulated by a 1023 chip primary PN sequence of duration equal to 1 ms. The data bits are unmodulated and last $L = 20$ primary code periods. The 1 ms correlator outputs obtained by despreading the GPS signal have been used for testing the proposed algorithm. The absolute value of the circular correlation (12) is reported in Fig. 3 as a function of the composite index k. The input signal was characterized by an estimated $C/N_0 = 30$ dB and $m = 10$ bit periods were considered. The algorithm is able to correctly determine the delay, τ, and the transmitted bit sequence. The 1 ms correlator outputs are compared with the reconstructed signal in Fig. 4, showing the good match between the two sequences. In Fig. 4, the reconstructed signal has been scaled in order to better show the agreement with the transmitted sequence.

In the lower part of Fig. 4, the algorithm is further tested by adding simulated noise to the correlator outputs. Although the C/N_0 is reduced by 10 dB, the algorithm is able to recover the transmitted sequence.

VI. Conclusions

In this paper, an efficient implementation of the joint ML estimator for data and bit epoch is proposed and tested using live GPS signals. The proposed algorithm exploits the properties of m-sequences for generating combining strings allowing the parallel evaluation of all possible bit combinations and alignments. The algorithm is extended to the case where the data bits are modulated by an overlay code. The proposed technique can be easily adapted to the presence of residual phase errors in the despreaded data samples.

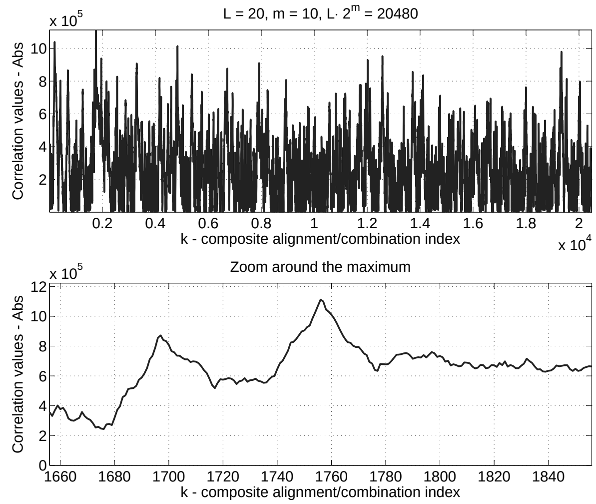

Fig. 3. Absolute value of the circular correlation (12) for different bit combinations and alignments. Live GPS data, $C/N_0 = 30$ dB-Hz, $L = 20$ and $m = 10$.

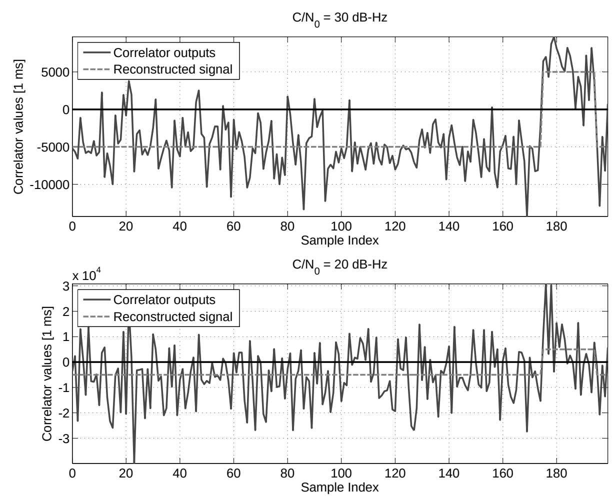

Fig. 4. 1 ms correlator outputs and reconstructed signal using the proposed algorithm. Upper part: original signal, $C/N_0 = 30$ dB. Lower part: signal with additional simulated noise, $C/N_0 = 20$ dB-Hz.

References

[1] S.-C. Rezeanu and R. E. Ziemer, “Fast-acquisition algorithm for joint maximum-likelihood data and bit synchronization epoch estimation in burst direct-sequence spread-spectrum transmission,” *IEEE Commun. Lett.*, vol. 2, no. 2, pp. 33 –3 5, Feb. 1998.

[2] S. W. Golomb and G. Gong, *Signal Design for Good Correlation: For Wireless Communication, Cryptography, and Radar*. Cambridge University Press, Jul. 2005.

[3] S.-C. Rezeanu, R. E. Ziemer, and M. A. Wickert, “Joint maximum-likelihood parameter estimation for burst DS spread-spectrum transmission,” *IEEE Trans. Commun.*, vol. 45, no. 2, pp. 227–238, Feb. 1997.

[4] D. Borio, “Fast m-sequence transform and secondary code constraints for composite GNSS signal acquisition,” *Submitted to IEEE Trans. Aerosp. Electron. Syst.*, Mar. 2009.