Selection of Parity Check Equations for the Iterative Message-Passing Detection of M-Sequences

Mathieu Des Noes, Valentin Savin, Laurent Ros, Jean-Marc Brossier

Selection of Parity Check Equations for the Iterative Message-Passing Detection of M-Sequences

M. des Noes, V. Savin, L. Ros and J.M. Brossier

***Abstract*—We consider the joint detection and decoding of m-sequences. The receiver has to decide whether an m-sequence is received and possibly to decode its initial state. To do so, it implements an iterative message-passing decoding algorithm that operates on a parity check matrix, built upon a number of reference parity-check equations satisfied by the m-sequence. This matrix concatenates several elementary parity check matrices which are derived from reference equations. Unlike the conventional decoding case, the detection problem imposes to consider false alarms that may occur when the decoder is only fed with noise. While absorbing sets are known to be responsible for the error floor phenomenon of iterative message-passing decoders, we show that they may have a beneficial effect on the detection performance, in that they may prevent the decoder to produce false alarms. We further compute the number of hybrid cycles of length 6 and 8 in the Tanner graph of the decoder and use the minimization of this number as criterion to derive an algorithm for selecting the reference parity check equations. This algorithm was found to be efficient for minimizing the probability of false alarm and decreases also the probability of wrong detection in the very small SNR region. This has been achieved at the cost of a reduction of the probability of correct detection.**

I. Introduction

Maximum length sequences (m-sequences) form a family of binary sequences with excellent correlation properties [1], widely used in wireless communications and positioning systems for synchronization, channel estimation or transmission in multipath channels [2][3][4].

The conventional method to synchronize with an m-sequence is to correlate the received signal with a replica of the searched sequence [5]. If a correlation peak is observed and is above a given threshold, the receiver is synchronized. An alternative method consists of performing detection by decoding the received signal. In fact, an m-sequence is a codeword of a cyclic linear code C defined by the characteristic polynomial $p(x)$ of the m-sequence [6]. It is thus possible to detect a transmitted sequence using a suitable decoder. This solution was originally proposed in cryptography for fast correlation attacks on stream ciphers [7][8][9]. It has been applied more recently in wireless communications and localization domains [10][11][12][13][14][15]. The task of the decoder is to simultaneously decide whether an m-sequence is received, and if so, to decode its initial state [16]. This corresponds to a joint detection and decoding problem [17]. Exploiting the parity check equations satisfied by the m-sequence, an iterative message-passing algorithm can be implemented to decode the received signal [18]. More specifically, [19] and [20] propose an iterative message passing algorithm based on a redundant graphical model (RGM) for the decoding of m-sequences or Gold sequences. The parity check matrix E concatenates K elementary parity check matrices. Each elementary matrix is generated by consecutive cyclic shifts of one parity check equation, referred to as the *reference parity check equation*. In order to ensure good error correction capabilities, the Hamming weight of this equation (number of non-zero coefficients), denoted by *t*, must be low. Indeed, it is known that the probability of correct decoding decreases with increasing *t* [7]. Consequently, one wants to use reference parity check equations having the smallest weight. We note that a parity check equation corresponds to a codeword of the dual code $C^{\perp}$. In case of an m-sequence, the dual code is the Hamming code *H* generated by the characteristic polynomial $p(x)$, and is known to contain many codewords of weight t=3 [21]. As a result, we will only consider parity check equations of weight $t = 3$ because it is the most favorable for decoding performance. Eventually, the properties of E depend solely on the K reference parity check equations.

In this article, we address the following problem : given a set of reference parity check equations of weight $t = 3$, how to select the K ones that will give the "best performance" ? The "best performance" corresponds in an ideal situation to a maximal probability of correct detection P_{CD}, and a minimal probability of wrong detection (P_{WD}) and false alarm P_{FA}. These performance criteria are defined in Section II-B. Unfortunately this objective is impossible to reach [22], one has to make a trade-off between these 3 parameters. In our study, we focus on a strategy aiming at minimizing P_{FA}. To the authors' best knowledge, this problem has not been addressed in the literature yet. Our strategy is motivated by the decisive impact of P_{FA} on the mean acquisition time of direct sequence spread spectrum systems [5]. A false alarm occurs if the decoder finds a codeword while there is only noise at the input. For instance this situation may happen if the sequence is transmitted occasionally (e.g. TDMA system) or when attempting to detect the scrambling code of the UMTS-FDD system [14].

Based on preliminary simulation results, we have observed that P_{FA} may range from 0.1 to 10^{-6} according to the selected parity check equations for the decoding of an m-sequence. This has a huge impact on the mean acquisition time. These surprising observations motivated us to provide an explanation. We found that absorbing sets [23] are responsible for these results. These topological structures of the decoding graph are already known to be responsible for error floors of LDPC codes [24][25]. They are fixed points of Gallager B decoding algorithm and prevent the decoder to converge in certain circumstances. When the decoder is only fed with noise, it shall not converge to a codeword otherwise it produces a false alarm. This non-convergence is obtained if the decoder is

trapped by an absorbing set. This situation is likely to happen if the Tanner graph contains many small absorbing sets. Since the selected parity check equations define the properties of the decoding graph, the existence of small absorbing sets depends on this selection. More precisely, some configurations may generate "hybrid" cycles that are responsible for the destruction of small absorbing sets and hence increase the occurrence of false alarms. The definition of a hybrid cycle is given in Section III-C. The identification of the small absorbing sets and the hybrid cycles allows us to derive an algorithm for selecting reference parity check equations. Its goal is the elimination of these hybrid cycles. The first step of the algorithm requires to count the number of cycles of length 6 and 8. To do so, we implemented the method proposed by Halford and Chugg in [26] to derive analytical formulas. Then, lower bounds on the number of cycles of length 6 and 8 are determined. The difference between the actual number of cycles and the lower bounds gives an evaluation of the number of hybrid cycles of length 6 and 8. The selection of the K reference parity check equations is based on the minimization of these numbers.

The paper is organized as follows. Section II details the generation of m-sequences, recalls some of their coding properties and describes the iterative message passing algorithm used to decode them. Section III presents a method for identifying small absorbing sets according to the reference parity check equations. Section IV details the proposed algorithm for the selection of parity check polynomials. The number of cycles of length 6 and 8 are evaluated as a function of the system parameters (K and t) and the parity check matrix. Using these expressions, a selection algorithm is derived. Section V shows the impact of the selected parity check polynomials on the detection performance and the benefit of our algorithm. Eventually, Section VI concludes this paper.

Notation: the index of a sequence is computed modulo its length N: $y(k) = y(k \mod N)$. The modulo 2 binary addition is noted with symbol $\oplus$. The notation $A = B \setminus C$ means A is the set of elements of B that are not in C. E^T is the transpose of matrix E and $tr(L)$ is the trace of the square matrix L.

II. Iterative decoding of m-sequences

In this section, we first define maximum length sequences, and then explain how they can be regarded as codewords of an error correcting code.

A. Generation of m-sequences

An m-sequence is generated by using a linear feedback shift register (LFSR) sequence generator such as depicted in Fig. 1. The feedback taps are given by the characteristic polynomial $p(x) = \sum_{k=0}^{r} p_k x^k$, with $p_0 = p_r = 1$. Moreover, for an m-sequence, $p(x)$ is a primitive polynomial of degree r, in which case the period of the generated sequence $\mathbf{y}$ is $N = 2^r - 1$ [27].

Let $u_y(i)$ be the content of the i^{th} shift-register of sequence $\mathbf{y} = (y(0), \cdots, y(N-1))$. The state of sequence $\mathbf{y}$ is the vector $\mathbf{u}_y = (u_y(0) \cdots u_y(r-1))$. There are 2 types of LFSR generators: Galois and Fibonacci feedback generator [27]. Both can be used to generate the same m-sequence. Using the Fibonacci generator of Fig. 1, the initial state of a sequence $\mathbf{y}$ is given by its first r chips: $u_y(0) = y(0), \cdots, u_y(r-1) = y(r-1)$.

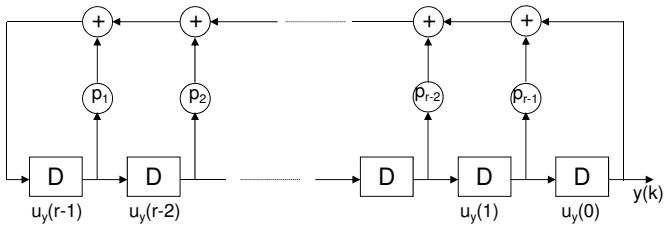

Figure 1. Fibonacci feedback generator.

The Fibonacci feedback generator of Fig. 1 can be seen as a linear encoding operation that encodes the information word $\mathbf{u}_y$ into the codeword $\mathbf{y}$. Hence, the set of all the m-sequences generated by $p(x)$ forms a linear code *S* of dimension r and length N. The r information bits are loaded into the registers of Fig. 1. The generator is run N times to produce the codeword $\mathbf{y}$. A parity check matrix of the code *S* is a matrix $\mathbf{E}$ such that $\mathbf{E}\mathbf{y}^T = 0$, for any m-sequence $\mathbf{y}$. Hence, any row of $\mathbf{E}$, say $\mathbf{g} = (g_0, ..., g_{N-1})$, is a codeword of the dual code of *S*, which is known to be the Hamming (cyclic) code *H* of dimension $N-r$ and length N, generated by $p(x)$ [6]. We refer to $\mathbf{g}$ as a parity check equation of *S*, or by a slight abuse of language, as a parity check equation of the m-sequence. Using the polynomial notation $g(x) = g_0 + g_1 x + \cdots + g_{N-1} x^{N-1}$, it follows that $g(x)$ is a parity check equation of *S* if and only if $g(x)$ is a multiple of $p(x)$ modulo $x^N - 1$.

A reference parity check equation is a parity check equation $g(x)$ with constant term equal to 1 ($g_0 = 1$). We define Υ_t as the set of reference parity check equations $g(x)$ having weight *t*. Let us also define $K_t = \text{card}(\Upsilon_t)$ the number of these equations. Other parity check equations are derived by cyclic shifts $(xg(x), \cdots, x^{N-1}g(x))$ of reference parity check equations. K_t is obtained by enumerating the number of codewords of the Hamming code *H* having weight *t* and a constant term equal to 1. The decoding performance is very sensitive to the weight of the parity check equations [7]. It is thus advisable to exploit only equations having a small weight *t*. For $t = 3$, we have $K_3 = 2^{r-1} - 1$ [21]. The number of reference parity check equations is fortunately very large for practical configurations. As a result, there is no need to search for equations with $t > 3$ since they will degrade the decoding performance. In the sequel, we will only consider the decoding of m-sequences using parity check equations of weight $t = 3$. In addition, the problem of finding parity check polynomials of weight $t = 3$ is not addressed in this paper as it has been deeply analyzed in cryptography for the implementation of fast correlation attacks [28][29]. It is assumed the receiver has already computed the set Υ_3 collecting such polynomials. They are referred to as trinomials throughout the rest of the paper.

B. Iterative message-passing detection

The goal is to implement a receiver capable to simultaneously detect the presence of an m-sequence $\mathbf{y}$ and estimate its initial state. To do so, it has to decide for one of the two following hypothesis:

$$\begin{array}{l} H_1 :\ R(i) = (-1)^{y(i)} + w(i) \\ H_0 :\ R(i) = w(i) \end{array} \tag{1}$$

$\mathbf{R} = (R(0), \cdots, R(M-1))$ is the received signal and M is the number of samples used by the decoding process. If $M < N$, this corresponds to an incomplete observation of sequence $\mathbf{y}$. The additive noise $w(i)$ is real zero mean white gaussian with variance σ^2.

Under hypothesis H_1, the receiver observes the m-sequence $\mathbf{y}$ with additive noise, while it only receives noise under H_0. This situation may happen if the sequence is transmitted occasionally (e.g. TDMA system) or when attempting to detect the uplink scrambling code of the UMTS-FDD [14] or CDMA2000 [13] system. Under hypothesis H_1, the receiver is assumed to be synchronized with the time frame. Hypothesis H_0 applies otherwise. The decision for one of the two hypothesis is based on the result of decoding the vector $(R(0), \cdots, R(M-1))$ [16]. The receiver decides for hypothesis H_1 if the decoder finds a valid codeword and H_0 otherwise. The codeword also gives the initial state of sequence $\mathbf{y}$.

The principle for the decoding of an m-sequence is to build a sparse parity check matrix E and then to apply an iterative message passing decoding algorithm on the induced bipartite graph [10][18]. For practical reasons, the decoder implements a Min-Sum algorithm (MS) [30] which provides an approximation of the maximum likelihood decoding of the sequence. It does not require the estimation of the noise variance σ^2.

Let $G = (V \cup F, \Xi)$ be the bipartite graph derived from the parity check matrix E. The set of variable nodes is noted V, F is the set of check nodes and Ξ is the set of edges connecting variables and check nodes. A check node is a parity check equation defined by a row of matrix E. A variable is connected to a check node if it is used by the corresponding equation. In order to improve the decoding performance, an usual design strategy is to consider redundant graphical models [11][19]. An augmented parity check matrix is built by concatenating K elementary parity check matrices:

$$E = \left[E_0^T\ E_1^T \cdots E_{K-1}^T\right]^T \tag{2}$$

The Tanner graph of the decoder is modified by the concatenation of these elementary matrices and so are the performances. If the combination of elementary matrices is well chosen, the number of small absorbing sets defined in Section III is reduced, and the probability of correct decoding is increased. Each matrix E_a is generated with a reference parity check polynomial $g_a(x) = \sum_{k=0}^{r_a} g_{a,k} x^k$ $(a = 0, \cdots, K-1)$:

$$\mathbf{E}_a = \begin{bmatrix} g_{a,0} & \cdots & \cdots & g_{a,r_a} & 0 & \cdots & \cdots & 0 \\ 0 & g_{a,0} & \cdots & \cdots & g_{a,r_a} & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & 0 & g_{a,0} & \cdots & \cdots & g_{a,r_a} \\ g_{a,r_a} & \cdots & \cdots & \cdots & 0 & g_{a,0} & \cdots & g_{a,r_a-1} \\ \vdots & \ddots & \cdots & \cdots & 0 & \ddots & \ddots & \vdots \\ g_{a,1} & \cdots & g_{a,r_a} & 0 & \cdots & \cdots & 0 & g_{a,0} \end{bmatrix} \tag{3}$$

$g_a(x)$ belongs to Υ_3, hence E_a has a row weight equals to $t = 3$. It is assumed that the receiver observes the sequence over its entire length $M = N$. As a consequence, E_a is circulant. This assumption will be helpful to derive the algorithm for selecting the K parity check polynomials. Simulation results will show that the algorithm provides also very good results when only a part of the sequence is observed ($M < N$).

E is thus a $KN \times N$ sparse matrix. The concatenation of K elementary matrices defines the overall bipartite graph on which the decoding algorithm is applied. While the row weight remains unchanged ($t = 3$), the K reference polynomials $g_0(x), \cdots, g_{K-1}(x)$ determine the structure of cycles appearing in the graph G, as well as specific topological configurations, named absorbing sets, which are known to have a significant impact on the decoding performance. This will be detailed in the next section.

The overall decoding operation is modeled as a function $dec(.)$ that produces a status indicator I_c and the estimated initial state $\hat{\mathbf{u}}_y$:

$$\{I_c, \hat{\mathbf{u}}_y\} = dec(R(0), R(1), \cdots, R(M-1)) \tag{4}$$

I_c is the indicator function of the decoder. It outputs a 1 if the decoder finds a valid codeword y, i.e. $\mathbf{E}\mathbf{y}^T = 0$. If the decoding is successful (i.e. $I_c = 1$), according to the Fibonacci representation of Fig. 1, the first r bits of the codeword represent the content of the shift registers at initialization. This produces the vector $\hat{\mathbf{u}}_y$.

The performance of the decoder is measured by the probability of correct detection P_{CD}, missed detection P_{ND}, wrong detection P_{WD} and false alarm P_{FA} defined as follows :

$$\begin{array}{l} P_{CD} = P(Ic = 1 \text{ and } \hat{\mathbf{u}}_y = \mathbf{u}_y | H_1) \\ P_{WD} = P(Ic = 1 \text{ and } \hat{\mathbf{u}}_y \neq \mathbf{u}_y | H_1) \\ P_{ND} = P(Ic = 0 | H_1) \\ P_{FA} = P(Ic = 1 | H_0) \end{array} \tag{5}$$

One notes that P_{CD}, P_{ND} and P_{WD} satisfy the following relation: $P_{CD} + P_{ND} + P_{WD} = 1$. As a result, we will only measure P_{CD} and P_{WD}.

Our goal is to define a rule for the selection of the K trinomials that will minimize P_{FA}. Indeed this parameter has a determinant impact on the mean acquisition time in a serial search [5]. A false alarm induces a penalty corresponding to the amount of time needed to discover the acquisition error.

We found that absorbing sets [23] have a decisive impact on false alarms. These topological structures of the decoding graph are fixed points of Gallager B decoding algorithm

and prevent the decoder to converge if some conditions are satisfied. When the decoder is only fed with noise, it shall not converge to a codeword otherwise it produces a false alarm. This non-convergence is obtained if the decoder is *trapped* by an absorbing set. This situation is likely to happen if the Tanner graph contains many small absorbing sets. Since the selected parity check equations define the properties of the decoding graph, the existence of small absorbing sets depends on this selection. We will first propose a method for identifying absorbing sets as a function of the selected parity check trinomials used by the decoder. This method is then used to explain how these absorbing sets simultaneously reduce P_{FA} and increase the probability of error $P_e = P_{ND}+P_{WD} = 1-P_{CD}$. Finally we will derive our algorithm for selecting parity check polynomials.

III. Absorbing sets

Absorbing sets have been introduced to explain error floors observed at high Signal to Noise Ratio (SNR) when decoding some LPDC codes over the AWGN channel [25]. They correspond to specific topological structures which characterize the behavior of the algorithm when it does not converge to a codeword.

Definition 1 ([31]): For a subset D of V, let $O(D)$ (resp. $E(D)$) be the set of neighboring vertices of D in F with odd (resp. even) degree with respect to D. Given an integer pair (a, b), an (a, b) absorbing set is a subset D of V such that:

1) D contains a variables and $O(D)$ contains b check nodes.
2) Every variable in D has strictly more neighbors in $E(D)$ than in $O(D)$.

Definition 2 ([31]): We say that an (a, b) absorbing set D is a fully absorbing set, if in addition, all variable nodes in $V \setminus D$ have strictly more neighbors in $F \setminus O(D)$ than in $O(D)$.

A fully absorbing set is a special type of trapping sets [24][32][33], and is a fixed point of the Gallager B decoding algorithm [34]. For instance, if the all zero codeword is sent and the bits corresponding to the variables of the absorbing set are erroneous ('1' instead of '0'), the decoder will not be able to correct them and thus will not converge. It is widely recognized that small fully absorbing sets are responsible for error floors in the decoding of LDPC codes [31][35]. In conventional situations, one wants to eliminate these absorbing sets in order to lower the error floor. In our context, we take the opposite direction, we want to ensure the presence of fully absorbing sets to avoid false alarms. In fact, when the decoder is only fed with noise (hypothesis H_0), one does not want the decoder to converge to a valid codeword. This desired situation happens if it is trapped by a fully absorbing set. This configuration occurs if all parity check equations of $E(D)$ are satisfied while those of $O(D)$ are not. If the number of fully absorbing sets is sufficiently large, the probability to be trapped becomes large when the input vector is large. On the other hand, when there is a valid codeword at the input, these absorbing sets may block the decoder and the probability of detection will decrease. This corresponds to the usual trade-off between probability of detection and false alarm in conventional detection theory [22].

We will first describe a method for constructing small absorbing sets corresponding to the parity check matrix defined by (2). Then we will validate by means of simulations that our assumption about the impact of absorbing sets on false alarms is correct. We will eventually explain how the selected parity check trinomials may destroy these absorbing sets, and thus allow the advent of false alarms.

A. Construction of absorbing sets

Let us first consider a circulant parity check matrix generated by the trinomial $g_a(x) = 1 + x^{i_a} + x^{r_a}$. Fig. 2 illustrates a cycle of length 6 connecting variables $y(k)$, $y(k + i_a)$ and $y(k+i_a-r_a)$. Parity check equation $F_a(k)$ corresponds to the k^{th} row of matrix E_a: $F_a(k) : y(k)\oplus y(k+i_a)\oplus y(k+r_a) = 0$. According to the preceding definition, $(y(k), y(k + i_a), y(k - r_a + i_a))$ is a $(3, 3)$ absorbing set. Each variable is connected to two check nodes belonging to $E(D)$ and one in $O(D)$. If two variables connected to $O(D)$ coincide, it is no longer a fully absorbing set. This leads to the following proposition.

Proposition 1: The absorbing set described by Fig. 2 is a fully absorbing set if $r_a \neq 3i_a$ and $r_a \neq 3i_a/2$.

Knowing that $r_a > i_a$, there are only two possibilities for two variables to coincide. If $r_a = 3i_a$ then $y(k-i_a+r_a)$ coincides with $y(k+2i_a)$. If $r_a = 3i_a/2$ then $y(k+2i_a-2r_a)$ coincides with $y(k - i_a)$.

Two other absorbing sets are obtained by changing k by $k-i_a$ or $k+r_a-i_a$. As a result, each parity check matrix E_a contains a total of N $(3, 3)$ absorbing sets. They are fully absorbing sets if $r_a \neq 3i_a$ and $r_a \neq 3i_a/2$.

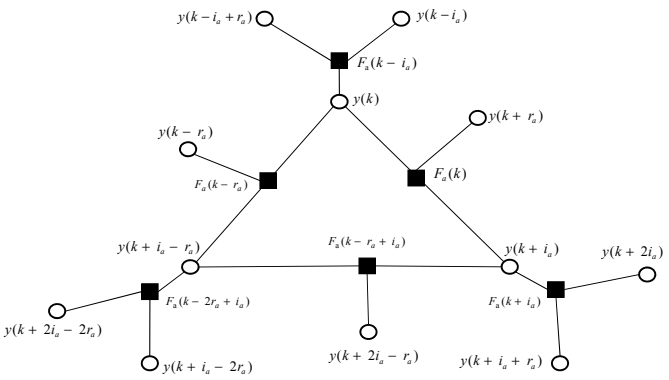

Figure 2. Cycle of length 6

Let us now consider a row circulant parity check matrix E generated with K trinomials according to (2) and (3). The method for constructing small absorbing sets is based on an incremental process. They are built upon initial $(3, 3)$ absorbing sets such as the one depicted in Fig. 2. Let $D_K(k)$ be an absorbing set obtained when using K parity check trinomials. Index k indicates that variable $y(k)$ belongs to $D_K(k)$. Other absorbing sets are obtained by a cyclic shift of index k. The absorbing set $D_{K+1}(k)$ is built upon $D_K(k)$ as it will be detailed now. The proposed construction method also assumes that every parity check equation connecting two

variables of $D_K(k)$ is not connected with a third variable of $D_K(k)$. The violation of this assumption leads to a destruction of the absorbing sets, as it will be detailed in Section III-C.

Two configurations are distinguished whether K is even or odd. If K is even ($K = 2Q$ for some integer Q), each variable of $D_K(k)$ is connected to $6Q$ parity check nodes and must be connected to *strictly* more parity checks of $E(D_K(k))$ than $O(D_K(k))$. As a result, it must be connected to at least $3Q + 1$ parity check nodes belonging to $E(D_K(k))$. For the same reason, if K is odd ($K = 2Q + 1$), each variable must be connected to $3Q + 2$ parity check equations of $E(D_K(k))$.

Let us denote $D_K(k) = \{y(k), y(k+\alpha_1), \cdots, y(k+\alpha_{a_K-1})\}$ the constructed absorbing set with K trinomials. It contains a_K variables, including $y(k)$. If we add a new elementary parity check matrix defined by trinomial $g_c(x) = 1+x^{i_c}+x^{r_c}$ to the decoder, matrix E_c contains cycles of length 6 such as the one depicted in Figure 2. The variables of this cycle are $\{y(k), y(k+\beta_c), y(k+\gamma_c)\}$, where $(\beta_c, \gamma_c) \in \{(i_c, i_c - r_c); (-i_c, -r_c); (r_c - i_c, r_c)\}$. We will now expose our construction method.

Theorem 1: If $D_{2Q+1}(k)$ is an absorbing set, then $D_{2Q+2}(k)$ defined by:

$$D_{2Q+2}(k) = D_{2Q+1}(k) \cup D_{2Q+1}(k+\beta_c) \cup D_{2Q+1}(k+\gamma_c) \tag{6}$$

is an absorbing set.

Proof: If we want $D_{2Q+2}(k)$ to be an absorbing set, one must add 2 connections in $E(D_{2Q+2}(k))$ to each variable belonging to $D_{2Q+1}(k)$. This is obtained by the duplication and translation of $D_{2Q+1}(k)$ defined by (6). We have to check that every variable in $D_{2Q+2}(k)$ has strictly more neighbors in $E(D_{2Q+2}(k))$ than in $O(D_{2Q+2}(k))$. To do so, we will enumerate the number of connections of $y(k)$ in $E(D_{2Q+2}(k))$. Since $y(k) \in D_{2Q+1}(k)$, $y(k)$ has at least $3Q+2$ connections in $E(D_{2Q+1}(k))$. In addition, $y(k)$ belongs to the cycle of length 6 $\{y(k), y(k+\beta_c), y(k+\gamma_c)\}$, which adds 2 connections to $E(D_{2Q+2}(k))$. As a consequence, $y(k)$ has at least $3Q + 2 + 2 = 3(Q + 1) + 1$ connections in $E(D_{2Q+2}(k))$ which is strictly larger than the number of connections to $O(D_{2Q+2}(k))$. As a result, $D_{2Q+2}(k)$ is an absorbing set. ■

Theorem 2: Let us define parameter $\delta_c \in \{i_c, -i_c, r_c, -r_c, r_c - i_c, i_c - r_c\}$ as any index of one of the 6 variables $y(k + \delta_c)$ connected to $y(k)$ by a parity check equation of matrix E_c.

If $D_{2Q}(k)$ is an absorbing set, then $D_{2Q+1}(k)$ defined by:

$$D_{2Q+1}(k) = D_{2Q}(k) \cup D_{2Q}(k+\delta_c) \tag{7}$$

is an absorbing set.

Proof: If we want $D_{2Q+1}(k)$ to be an absorbing set, one must add only one connection in $E(D_{2Q+1)}(k))$ to each variable belonging to $D_{2Q}(k)$. This is obtained by the duplication and translation of $D_{2Q}(k)$ defined by (7). Since $y(k) \in D_{2Q}(k)$, $y(k)$ has at least $3Q + 1$ connections in $E(D_{2Q}(k))$. In addition, $y(k)$ is connected to $y(k + \delta_c)$ by a the parity check equation of the form $F_c(k + ...)$. As a consequence, $y(k)$ has $3Q + 2$ connections in $E(D_{2Q+1}(k))$ which is strictly larger than the number of connections to $O(D_{2Q+1}(k))$. As a result, $D_{2Q+1}(k)$ is an absorbing set. ■

Absorbing sets are constructed by the alternative application of Theorems 1 and 2. The initial $(3, 3)$ absorbing set $D_1(k)$ is generated according to Fig. 2. Fig. 3 shows an example of the construction of $D_{2Q+2}(k)$ with $\beta_c = r_c - i_c$ and $\gamma_c = r_c$. Fig. 4 shows an example of the construction of $D_{2Q+1}(k)$ with $\delta_c = i_c$.

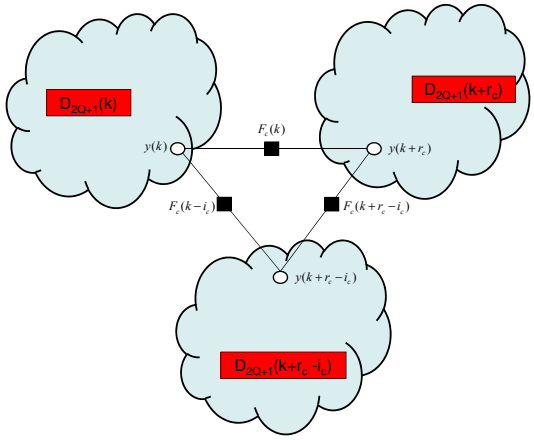

Figure 3. Construction of $D_{2Q+2}(k)$ upon $D_{2Q+1}(k)$

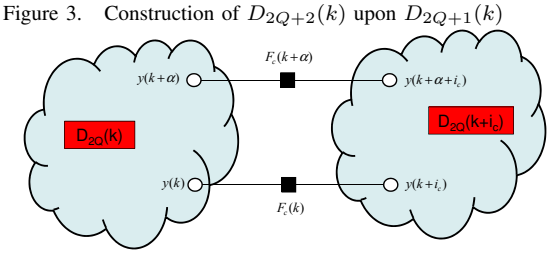

Figure 4. Construction of $D_{2Q+1}(k)$ upon $D_{2Q}(k)$

Fig. 5 illustrates a $(9, 18)$ absorbing set constructed from the initial cycle of Fig.2, according to (6). When $K = 2$, there are 2 parity check trinomials : $g_a(x) = 1+x^{i_a}+x^{r_a}$ and $g_b(x) = 1+x^{i_b}+x^{r_b}$. The 3 triangles colored in red identify the initial cycle of length 6 $D_1(k) = \{y(k), y(k + i_a), y(k + i_a - r_a)\}$ and its 2 shifted versions $D_1(k + r_b)$ and $D_1(k + r_b - i_b)$. The black lines show the connections between variables of $D_2(k)$ with parity check equations of the form $F_b(k + ...)$. Each variable is connected to 4 check nodes of degree 2 and 2 check nodes of degree 1. In Fig. 5, check nodes of degree 1 are not displayed since they do not connect variables belonging to the absorbing set. We validated by computer simulations that there is no smaller absorbing sets. There are 3 absorbing sets of size $(9, 18)$ derived from each cycle of length 6 containing variable $y(k)$. Since there are 3 possible cycles of length 6, there is eventually a total of 9 absorbing sets of size $(9, 18)$ containing $y(k)$. Since each one contains 9 variables, there are N such $(9, 18)$ absorbing set in the graph.

The size of the constructed absorbing sets is computed iteratively. First, we observe that $a_{2Q} = 3a_{2Q-1}$ and $a_{2Q+1} = 2a_{2Q}$. Since $a_1 = 3$, one obtains:

$$\begin{aligned} a_{2Q} &= 2^{Q-1}3^{Q+1} \\ a_{2Q+1} &= 2^{Q}3^{Q+1} \end{aligned} \tag{8}$$

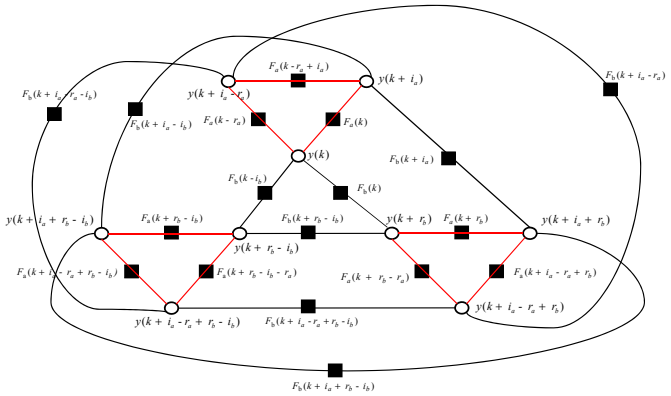

Figure 5. Absorbing set for $K = 2$ parity check polynomials

Table I
ABSORBING SET BLOCKING PROBABILITY - M-SEQUENCE 2415.

$i_a = 25, r_a = 49$		$i_a = 34, r_a = 65$		$i_a = 37, r_a = 76$	
Q	$P_{\text{block}}(Q)$	Q	$P_{\text{block}}(Q)$	Q	$P_{\text{block}}(Q)$
0	0.012	0	0.01	0	0.015
1	0.686	1	0.66	1	0.685
2	1	2	1	2	1
3	1	3	1	3	1

Table II
ABSORBING SET BLOCKING PROBABILITY - M-SEQUENCE 4445.

$i_a = 4, r_a = 49$		$i_a = 22, r_a = 73$		$i_a = 56, r_a = 93$	
Q	$P_{\text{block}}(Q)$	Q	$P_{\text{block}}(Q)$	Q	$P_{\text{block}}(Q)$
0	0.0	0	0.0	0	0.002
1	0.452	1	0.484	1	0.492
2	0.999	2	1	2	0.999
3	1	3	1	3	1

B. Influence of absorbing sets on false alarms

The variables of a fully absorbing set $D_K(k)$ are blocked if parity check equations belonging to $E(D_K(k))$ are satisfied while those of $O(D_K(k))$ are not [25]. As shown by (8), the size of the smallest absorbing set a_K increases exponentially with the number of elementary parity check matrices. Thus, it will be more difficult to satisfy this constraint as K increases and the decoder will be blocked less frequently. As a result, the decoder improves its correction capability and P_{CD} increases. Unfortunately, P_{WD} increases as well.

We also observed experimentally that the presence of small absorbing sets leads to a decrease of P_{FA}. This hypothesis has been validated by analyzing the evolution of parity check equations during the different steps of the iterative decoding. We have applied the same method as Richardson for the study of trapping sets [24]. A vector of N random binary chips is generated, and then decoded with Gallager B algorithm [34]. The decoder uses only one elementary parity check matrix $\mathbf{E}_a$ generated with trinomial $g_a(x) = 1 + x^{i_a} + x^{r_a}$, according to (3). The values of parameters i_a and r_a depend on the m-sequence's characteristic polynomial. They are given in Table **??** for m-sequences with characteristic polynomial 2415 and 4445 in octal notation.

We observed the decoder was systematically blocked when it reaches the maximum number of iterations, set to 50. Then we identified the fully absorbing sets corresponding to Fig. 2 that were blocked at the 50^{th} iteration. Eventually, we measured the probability that one of theses absorbing set was already blocked at the Q^{th} iteration. It is denoted $P_{\text{block}}(Q)$. Table I and II show the simulation results for m-sequence 2415 and 4445, and different trinomials $g_a(x)$. The decoder cannot converge towards a codeword and thus does not produce a false alarm. This example demonstrates the impact of absorbing sets on the decoder's performance. This will be even more visible with simulation results presented in Section V.

C. Influence of hybrid cycles

The construction method proposed in this paper assumes every parity check equation connecting 2 variables of $D_K(k)$ is not connected to a third variables of $D_K(k)$. If this happens, this parity check equation does not belong to $E(D_K)$ but to $O(D_K)$. As a consequence, the variables connected to this node have more connections with $O(D_K)$ than with $E(D_K)$ and $D_K(k)$ is not an absorbing set any more. We will now show that this situation may occur if there exists 'hybrid' cycles in the graph.

Let us assume $D_K(k)$ is an absorbing set. According to the construction method, $D_{K+1}(k)$ is obtained by a duplication and a translation of $D_K(k)$. The translated sets are connected with parity check nodes belonging to the added elementary parity check matrix. If for instance $K = 2Q + 1$, variables of $D_{2Q}(k)$ and $D_{2Q}(k + \delta_c)$ of the new absorbing set $D_{2Q+1}(k) = D_{2Q}(k) \cup D_{2Q}(k + \delta_c)$ are connected with parity check equations such as $F_c(k + \alpha + \delta_c)$, where α is any index of a variable belonging to $D_{2Q}(k)$ (Fig. 4). Let us assume the node $F_c(k + \delta_c)$, which links $y(k)$ and $y(k + \delta_c)$, is also connected to a third variable $y(k + \alpha)$ belonging to $D_{2Q+1}(k)$. We also assume that $y(k + \alpha) \in D_{2Q}(k)$. If $y(k + \alpha) \in D_{2Q}(k + \delta_c)$, the explanation that will be given now is also valid by replacing k by $k + \delta_c$.

If variable $y(k + \alpha) \in D_{2Q}(k)$, there exists already a path between $y(k)$ and $y(k + \alpha)$ in the graph induced by the variables of $D_{2Q}(k)$ and the parity check nodes connecting the variables of $D_{2Q}(k)$. This property is easily proven by induction. This path does not contain parity check nodes like $F_c(k + ...)$ which depends on parameters i_c and r_c. As a consequence, if $F_c(k + \delta_c)$ connects $y(k)$ and $y(k + \alpha)$, this creates a cycle in the graph. This cycle contains parity check nodes belonging to different elementary parity check matrices. It is referred as "hybrid". In addition, $F_c(k + \delta_c)$ is no more in $E(D_{2Q+1}(k))$ and $y(k)$ is connected to more check nodes belonging to $O(D_{2Q+1}(k))$. As a result, $D_{2Q+1}(k)$ is not an absorbing set.

We will now give an example corresponding to the decoding of the m-sequence defined by the following characteristic polynomial: $p(x) = 1 + x^2 + x^3 + x^8 + x^{10}$. Let us consider $K = 3$ parity check trinomials of this m-sequence: $g_a(x) = 1+x^{34}+x^{65}$, $g_b(x) = 1+x^{37}+x^{76}$ and $g_c(x) = 1+x^{72}+x^{77}$. A $(18, 72)$ absorbing set $D_3(k)$ is constructed with the method proposed in the previous section: $D_3(k) = D_2(k) \cup D_2(k+r_c)$. Figure 6 shows the connections of variable $y(k+r_b-i_b+r_c)$ (in the center of the figure). When there is no hybrid cycle,

each variable is connected to 5 nodes belonging to $E(D_3(k))$ and 4 to $O(D_3(k))$. It is thus an absorbing set. In this situation, variable $y(k + r_b - i_b + i_c)$ (colored in red in the figure) does not belong to $D_3(k)$. On the other hand, the 3 selected trinomials give birth to a hybrid cycle, depicted in Figure 7. This is due to the following equality: $i_c + r_b - i_b = i_a + r_c$ $(72+76-37 = 34+77)$. This adds a new connection between the node $F_c(k + r_b - i_b)$ and variable $y(k + i_a + r_c)$. Since $y(k + i_a + r_c)$ already belongs to $D_2(k)$ and thus to $D_3(k)$, the node $F_c(k + r_b - i_b)$ is now connected to 3 variables in $D_3(k)$ and belongs to $O(D_3(k))$. As a result, variables $y(k + r_b - i_b + r_c)$, $y(k + r_b - i_b)$ and $y(k + i_a + r_c)$ have more neighbors in $O(D_3(k))$ than in $E(D_3(k))$ and $D_3(k)$ is not an absorbing set anymore. This means, the decoder will be able to correct errors in this set and this increases its ability to converge to a valid codeword. In this case, the probability of false alarm will increase.

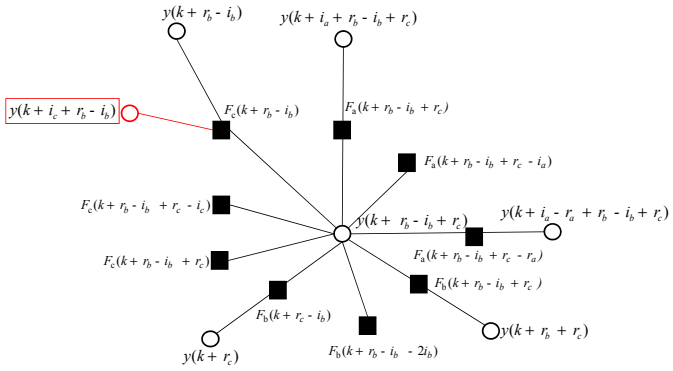

Figure 6. Destruction of a fully absorbing set with a cycle of length 6

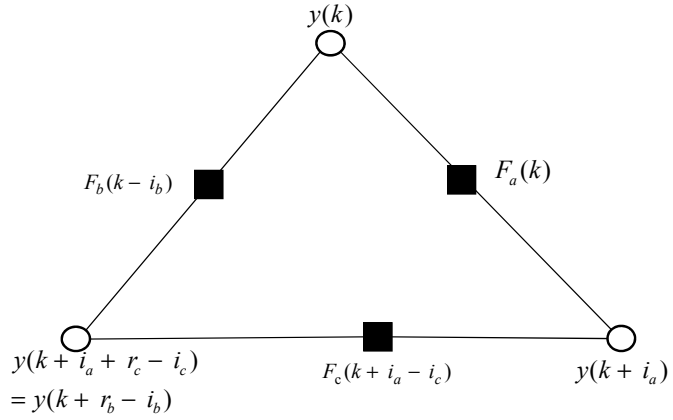

Figure 7. Hybrid cycle of length 6

IV. Selection of Parity Check Trinomials

The impact of hybrid cycles gave us the idea to derive an algorithm for selecting the parity check trinomials used for decoding. If one seeks to minimize the probability of false alarm, it is necessary to select trinomials that do not destroy absorbing sets. One way to achieve this goal is to select trinomials that minimize the number of hybrid cycles. A prerequisite is that there is no cycle of length 4 in the graph. Then, we choose to minimize the number of hybrid cycles of length 6 and 8 which are mostly encountered. While the impact of cycles of length 6 has been deeply analyzed in the previous section, cycles of length 8 will have the same effects on detection performance. To do so, it is required to compute the number of cycles of length 6 and 8: I_6 and I_8. This will provide a measure of the number of hybrid cycles. Indeed, there exists native cycles of length 6 and 8 which can be enumerated. These native cycles exist whatever the selected trinomials. This is for example the cycles of length 6 and 8 depicted in Fig. 2 and 8. Subtracting the number of native cycles of length 6 and 8 from the actual number of cycles gives a measure of the number of hybrid cycles, which can then be used to derive an algorithm for selecting trinomials.

A. Detection of cycles of length 4

A cycle of length 4 exists if it is possible to draw a square in the parity check matrix. This happens if the dot product between two distinct rows is strictly larger than 1. Let us consider the possibility to have such a cycle between 2 matrices E_a and E_b. Assuming $r_b \geq r_a$ and using the circulant structure of each matrix, a cycle of length 4 exists if and only if one of the following conditions is satisfied:

$$r_b - i_b = i_a; \quad r_b - i_b = r_a; \quad r_b - i_b = r_a - i_a; \quad r_a - i_a = i_b \tag{9}$$

This property is easily proven by enumerating the theoretical possibilities. If there is only one elementary parity check matrix ($a = b$), the only possibility is $r_a = 2i_a$. The trinomials are selected so that none of the equalities defined in (9) is satisfied.

B. Number of cycles of length 6 and 8

In order to compute I_6, we apply the method proposed by Halford and Chugg in [26]. Let us define the following matrix notations:

$$\begin{array}{ll} Y = X - \lambda = [y(i,j)] & y(i,j) = x(i,j) - \lambda \\ Y = max(X, 0) = [y(i,j)] & y(i,j) = max(x(i,j), 0) \\ U = X \circ Y = [u(i,j)] & u(i,j) = x(i,j)y(i,j) \\ Z(X) = X - X \circ I & \end{array} \tag{10}$$

I is the identity matrix having the same size as matrix X. The operator that sets the diagonal elements of matrix X to zero is denoted $Z(X)$.
Let us also define $A = EE^T$, $B = E^T E$, $\tilde{A} = A \circ I$, $\tilde{B} = B \circ I$, $\tilde{B}_m = max(\tilde{B} - 1, 0)$ and $\tilde{A}_m = max(\tilde{A} - 1, 0)$, where E is the parity check matrix defined by (2) and (3). Applying the method described in [26], we obtain:

$$\begin{array}{ll} I_6 = & \frac{1}{6} tr(L) \\ L = & Z(A)A^2 - E\tilde{B}_m BE^T - Z(A\tilde{A}_m)A \\ & -EZ(B\tilde{B}_m)E^T + \tilde{A}_m E\tilde{B}_m E^T \end{array} \tag{11}$$

We assume each elementary parity check matrix E_a is circulant. This assumption is valid when the receiver observes the sequence over its entire length $M = N$. This greatly simplifies the computation of I_6. It will be shown in Section V that the derived selection procedure will also be efficient when the receiver observes only a portion of the sequence ($M < N$).

Theorem 3: Let E be a row circulant parity check matrix built with K parity check polynomials of weight t in accordance with (2) and (3). The number of cycles of length 6 of E is:

$$I_6 = \frac{1}{6}(tr\left(A^3\right) - KNt((K^2+3K+1)t^2 - 3(K+1)t + 2)) \quad (12)$$

Proof: The evaluation of $tr(L)$ is detailed in Appendix A. ■

Since *t*, K and N are fixed parameters, the minimization of I_6 is equivalent to minimizing $tr(A^3)$. This can be done by selecting the set of trinomials leading to the smallest value of I_6, according to (12). The complexity of computing $tr\left(A^3\right)$ could be reduced. This can be done by exploiting the structure of matrix A, and the circulant property of its sub-matrices.

The same method is applied to compute the number of cycles of length 8 (I_8).

Theorem 4: Let us reuse the definition of the row circulant matrix E from Theorem 3. The number of cycles of length 8 of E is defined by:

$$I_8 = \frac{1}{8}(tr(A^4) - 4(Kt+t-2)tr(A^3) + (3K^3+10K^2+10K+3)NKt^4 - (14K^2+32K^+14)NKt^3 + 22(K+1)NKt^2 - 11NKt) \quad (13)$$

Proof: The proof is detailed in Appendix B. ■

The results provided by (12) and (13) have been successfully cross-checked with the software library developed by Halford and Chugg [26]. It used to be downloadable, but this is unfortunately no longer the case.

C. Algorithm

The selection algorithm relies on the minimization of hybrid cycles of length 6 and 8. This maximizes the number of absorbing sets and thus minimizes the probabilities of false alarm and wrong decoding. Subtracting the number of native cycles of length 6 and 8 from the actual number of native cycles gives a measure of the number of hybrid cycles. I_6 and I_8 are given by (12) and (13). We will now evaluate the number of native cycles.

Within each matrix E_a, there are 3 cycles of length 6 containing variable $y(k)$. This is illustrated by Fig. 2 for one cycle. The 2 others are obtained by exchanging k by $k - i_a$ and $k + r_a - i_a$. Since there are 3 variables per cycle of length 6, I_6 is lower bounded by:

$$I_{6,min} = NK \quad (14)$$

Fig. 8 shows the 6 cycles of length 8 containing variables $y(k)$ and $y(k+i_a)$ and check nodes of two elementary parity check matrices E_a and E_b. Since $y(k)$ is connected to 6 other variables through the 3 parity check equations of E_a, there are 6 figures equivalent to Fig. 8. As a result, there are 36 cycles of length 8 containing $y(k)$. If $K > 1$ trinomials are used, there are C_K^2 pairs of polynomials, and each pair gives birth to $36N/4 = 9N$ cycles of length 8. As a result, I_8 is lower bounded by:

$$I_{8,min} = 9NC_K^2 = 9NK(K-1)/2 \quad (15)$$

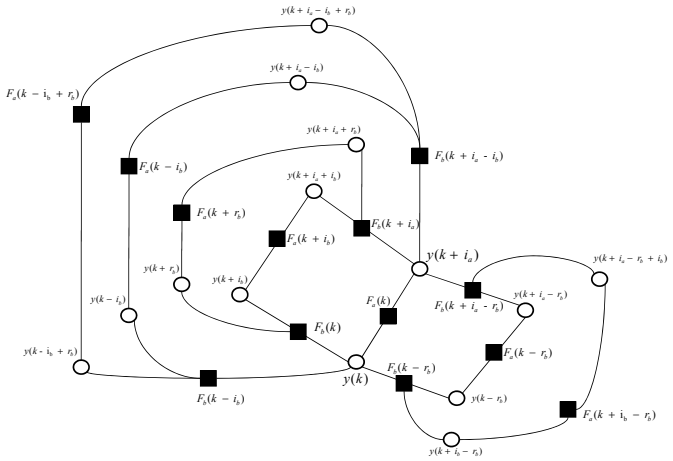

Figure 8. Cycles of length 8 containing $y(k)$ and $y(k+i_a)$

The algorithm looks for the combination of parity check trinomials that minimizes the number of hybrid cycles of length 6 and 8. These numbers are given by the differences $I_6 - I_{6,min}$ and $I_8 - I_{8,min}$. Simulation results have shown that trinomials configurations satisfying $I_6 = I_{6,min}$ are systematically found. On the other hand, this is not the case for $I_8 - I_{8,min}$. As a consequence, the algorithm looks for a configuration $(g_0(x), \cdots, g_{K-1}(x))$ satisfying $I_6 = I_{6,min}$ and minimizing $I_8 - I_{8,min}$. The algorithm is detailed below. The loop is stopped after N_s iterations to avoid an endless search.

Algorithm Selection of K parity check trinomials - Minimization of I_6 and I_8

$q = 0$ and min $= +\infty$
while $q < N_s$ **do**
  Select K distinct trinomials $g_0(x), \cdots, g_{K-1}(x)$ without cycles of length 4.
  Compute I_6 according to (12)
  if $I_6 == NK$ **then** compute I_8 according to (13)
    if (I_8 < min) **then** save configuration $(g_0(x), \cdots, g_{K-1}(x))$ and min $\leftarrow I_8$
    end if
  end if
end while

V. Performance

In this section, the performance of the algorithm for selecting parity check trinomials is presented. The receiver observes M samples $R(0), \cdots, R(M-1)$, according to the model of (1) and try to decode an m-sequence **y** the characteristic polynomial of which is $p(x) = 1 + x^2 + x^5 + x^8 + x^{11}$. Trinomials have then been derived from $p(x)$ using an exhaustive search. The parity check matrix E used by the decoder concatenates $K = 5$ elementary matrices, according to (2). For practical reasons, the decoder implements a Min-Sum (MS) algorithm [30] since it does not require the estimation of the noise variance unlike the Sum-Product algorithm [18].

The decoder is modeled as a function $dec_y(.)$ defined by (4). It stops when either all the parity check equations are satisfied or the maximum number of iteration $N_{iter} = 60$ is reached. Its performances (P_{CD}, P_{WD}, and P_{FA}) are defined by (5). Parity check trinomials are noted $g_a(x) = 1 + x^{i_a} + x^{r_a}$. The values of i_a and r_a are listed in Table V of Appendix C.

Table III shows P_{FA} for different configurations of selected trinomials detailed in Table IV. The measurements are independent from the input gaussian noise variance: it is due to a known property of the MS algorithm. It is insensitive to a uniform scaling of input variables $R(i)$'s [30]. Consequently, the decoder will have the same performance if the input variables are gaussian with variance 1 or σ^2. We observe that P_{FA} ranges from 0.77 to 10^{-6} depending on the selected configuration. This is a huge variation. The lower bound on the number of cycles of length 6 is $I_{6,min} = 2047 \times 5 = 10235$. The probability of false alarm is reduced from 0.77 to 10^{-2} when I_6 drops from 38893 cycles (configuration 'a') to the lower bound (configuration 'e'). Reducing the number of cycles of length 8 improve dramatically the performance. P_{FA} drops to 10^{-6} when $I_6 = I_{6,min}$ and I_8 is reduced to 196512 cycles (configuration 'g'). These results validate our interpretation of the effect of absorbing sets on the false alarm rate, described in Section III-B. When the number of cycles is reduced, small absorbing sets are preserved which favors the blocking of the decoder and hence prevents false alarms. If we denote $P_t = P$("at least one absorbing set is blocked"), then we have $P_{FA} = 1 - P_t$. When the number of absorbing sets is large, we have $P_t \to 1$ but $P_t \neq 1$.

Table IV also shows P_{FA} when the sequence is not observed over its entire length. The signal processed by the decoder is trunked from $M = 2047$ to $M = 1023$ chips. With this configuration, the elementary matrix E_a reduces to the submatrix determined by the first $M - r_a$ rows and the first M columns (where r_a is the degree of the trinomial generating E_a), corresponding to the $M - r_a$ parity check equations on the observed sequence of length M. Since the effective E_a matrices are no longer circulant, the evaluation of the number of cycles with (12) and (13) is no longer valid and the selection algorithm relies on a wrong assumption. Nevertheless, one observes that P_{FA} is not sensitive to this truncation. In fact, the algorithm will reduce the number of hybrid cycles even if the size of the effective parity check matrix is reduced. This explains why it remains efficient.

Fig. 9 shows the error probability $P_e = 1 - P_{CD} = P_{ND} + P_{WD}$ as a function of the input SNR for configurations 'a', 'e' and 'g'. P_e is the Frame Error Rate (FER) measured when decoding conventional error correcting schemes. One observes that P_e is sensitive to the number of hybrid cycles. When I_6 and I_8 are large (configurations 'a' and 'e'), there are many hybrid cycles that destroy absorbing sets and thus improve the error correction capability of the decoder. On the other hand, when I_6 reaches its lower bound and I_8 is minimized (configuration 'g'), P_e increases noticeably. The SNR required to reach the target point $P_e = 10^{-2}$ is increased by almost 2.5 dB between configurations 'a' and 'g'. The curve denoted 'a+verif' shows P_e for configuration 'a' followed by a verification phase, i.e. the initial state estimated by the decoder is used to generate the corresponding m-sequence which is then correlated with the input signal. A detection threshold is set so that $P_{FA} \approx 10^{-6}$. We observe a gain of $1.5dB$ compared to configuration 'g' for a similar probability of false alarm. On the other hand, this method requires the implementation of a correlation over a long integration length that increases the complexity of the receiver.

Fig. 10 shows the probability of wrong detection P_{WD} as a function of the input SNR for configurations 'a', 'c' and 'e'. We observe that P_{WD} follows the same trends as P_{FA} in the very small SNR region. The elimination of hybrid cycles of length 6 reduces P_{WD} from 0.7 to 10^{-2}, which is the same reduction as the one observed for false alarms. If, in addition, the number of cycles of length 8 is minimized, there is almost no hybrid cycles and the decoder is blocked when the SNR is very small. This eliminates wrong detection. For instance, we did not observe any wrong detections with configuration 'g'.

Table III
PROBABILITY OF FALSE ALARM FOR THE CONFIGURATIONS OF $K = 5$ PARITY CHECK TRINOMIALS.

Reference	P_{FA} ($M = N = 2047$)	P_{FA} ($M = 1023$ and $N = 2047$)
a	0.77	0.68
b	0.297	-
c	0.17	-
d	0.094	0.086
e	0.0159	-
f	$5.2\,10^{-4}$	$4.3\,10^{-4}$
g	$1.0\,10^{-6}$	-

Table IV
NUMBER OF CYCLES OF LENGTH 6 AND 8 FOR THE CONFIGURATIONS OF $K = 5$ PARITY CHECK TRINOMIALS.

Reference	configuration	I_6	I_8
a	$(g_1, g_2, g_3, g_4, g_6)$	38893	589536
b	$(g_1, g_2, g_3, g_4, g_9)$	24564	442152
c	$(g_1, g_2, g_3, g_4, g_{11})$	20470	350037
d	$(g_1, g_2, g_3, g_5, g_7)$	16376	419635
e	$(g_1, g_2, g_3, g_5, g_8)$	10235	450340
f	$(g_1, g_2, g_3, g_8, g_{10})$	10235	337755
g	$(g_1, g_2, g_3, g_{12}, g_{13})$	10235	196512

VI. CONCLUSION

The iterative message-passing detection is an interesting solution for searching long m-sequences. The goal is to implement a receiver capable to simultaneously detect the presence of an m-sequence $\mathbf{y}$ and estimate its initial state. To achieve this task, the receiver implements a decoder that tries to detect m-sequences as codewords of a linear code. The principle for the decoding of an m-sequence is to build a sparse parity check matrix E and then to apply an iterative message passing decoding algorithm on the induced bipartite graph. The parity check matrix E concatenates K elementary parity check matrices, each being generated by a single reference parity check equation of weight t.

We have first proposed a method for identifying small absorbing sets in the Tanner graph induced by E when $t = 3$. The presence of absorbing sets tends to favor the blocking of the decoder when it is only fed by noise. This avoids false

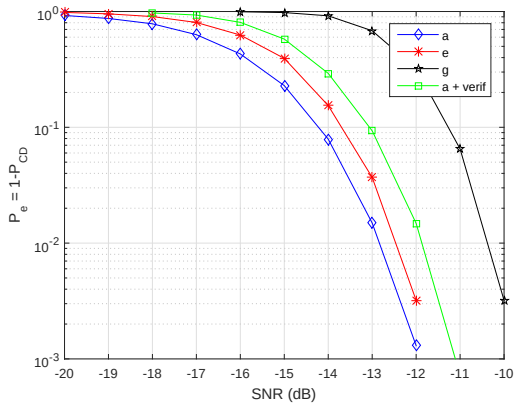

Figure 9. Probability of error detection ($P_e = 1 - P_{CD}$)

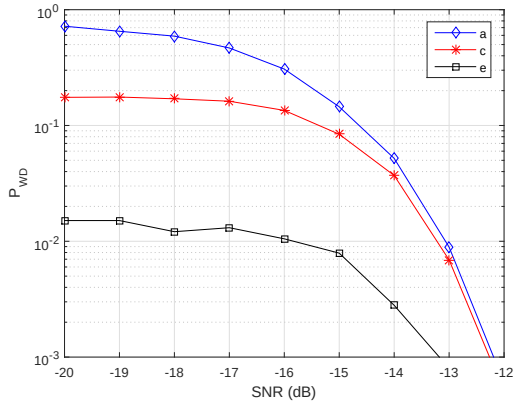

Figure 10. Probability of wrong detection (P_{WD})

alarms and also minimizes wrong detection in the very small SNR region. Then we have shown how absorbing sets can be destroyed by hybrid cycles. These specific cycles contain parity check nodes belonging to different elementary parity check matrices.

Then, we have proposed an algorithm for selecting these reference parity check equations of weight $t = 3$. It minimizes the number of hybrid cycles in the Tanner graph of the decoder in order to reduce significantly the probabilities of false alarm and wrong detection. To do so, we computed the number of cycles of length 6 and 8 for a redundant parity check matrix concatenating K circulant elementary parity check matrices of row weight t. Then lower bounds on the number of cycles of length 6 and 8 have been derived for $t = 3$. The difference between the actual number of cycles and the lower bounds gives a measure of the number of hybrid cycles of length 6 and 8. The proposed algorithm selects a combination of reference equations that minimize these measures. It was found to be very efficient for almost eliminating false alarms and wrong detections. This was obtained at a cost of a degradation of the probability of correct detection.

In this paper, we proved our method identifies small absorbing sets in the Tanner graph. Unfortunately, we did not prove they are *fully* absorbing sets and hence cannot claim they are fixed points of the Gallager B decoding algorithm applied to the parity check matrix E. We strongly believed the *fully* attribute is related to the absence of hybrid cycles but this remains to be validated. Eventually, the principle of our algorithm could be extended to other cyclic codes having a weight larger than $t = 3$.

APPENDIX

A. Computation of $tr(L_{mod})$

Using the trace property ($tr(AB) = tr(BA)$) and expending terms of the form $Z(X)$, (11) can be rearranged as follows :

$$\begin{aligned} I_6 = \quad & \frac{1}{6} tr(L_{mod}) \\ L_{mod} = \quad & A^3 - \tilde{A}A^2 - 2\tilde{B}_m B^2 - \tilde{A}_m A^2 \\ & + (A\tilde{A}_m \circ I)A + (B\tilde{B}_m \circ I)B \\ & + \tilde{A}_m E \tilde{B}_m E^T \end{aligned} \tag{16}$$

$\tilde{A}$ is the matrix formed with the diagonal elements of A. As a consequence, we have $\tilde{A} = tI$ and $\tilde{A}_m = (t-1)I$, where I is identity matrix having the same size as matrix A. Similarly, $\tilde{B} = KtI$ and $\tilde{B}_m = (Kt-1)I_M$. In addition, invoking the commutative property of the trace, we have $tr(A) = tr(EE^T) = tr(E^TE) = tr(B) = KNt$ and $tr(B^2) = tr(A^2)$. Inserting all these results in (16), one obtains :

$$\begin{aligned} tr(L_{mod}) = \quad & tr\left(A^3\right) - (2(K+1)t - 3)tr\left(A^2\right) \\ & + KNt\left((K^2+K+1)t^2 - 2(K+1)t + 1\right) \end{aligned} \tag{17}$$

$tr\left(A^2\right)$ can also be evaluated using the circulant property of sub-matrices E_a. Let us define matrix $F_{ab} = E_a E_b^T$, then $tr\left(A^2\right)$ can be written as follows :

$$tr\left(A^2\right) = \sum_{i=0}^{K-1}\sum_{j=0}^{K-1}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}\left(F_{ij}(u,v)^2\right) \tag{18}$$

Since E_a and E_b are circulant, this property also holds for F_{ab}. As a consequence :

$$S_{i,j} = \sum_{u=0}^{N-1}\sum_{v=0}^{N-1} F_{ij}(u,v)^2 = N\sum_{v=0}^{N-1} F_{ij}(0,v)^2 \tag{19}$$

Moreover, we have $F_{ba} = F_{ab}^T$, thus $S_{j,i} = S_{i,j}$

$$tr\left(A^2\right) = \sum_{i=0}^{K-1} S_{i,i} + \sum_{i=0}^{K-1}\left(\sum_{j=0}^{i-1} S_{i,j} + \sum_{j=i+1}^{K-1} S_{i,j}\right) \tag{20}$$

If $i \neq j$, $F_{ij}(0,v)$ is either equal to 0 or 1. Since there is no cycle of length 4, it is not possible to have more than one coincidence position between polynomials $g_a(x)$ and $g_b(x)$. As a result, there are t^2 elements equal to 1 when $i \neq j$. If $i = j$, $F_{ii}(0,0) = t$ and $F_{ii}(0,v)$ is also either equal to 0 or 1 for $v > 0$. There are $t^2 - t$ elements equal to 1 and one element equal to t^2. As a consequence, we have :

$$\begin{aligned} S_{i,j} &= Nt^2 \quad \text{if } i \neq j \\ S_{i,i} &= N(2t^2 - t) \end{aligned} \tag{21}$$

Inserting (21) in (20), one obtains :

$$tr\left(A^2\right) = NKt((K+1)t - 1) \tag{22}$$

This result is inserted in (17) to give (12).

B. Computation of I_8

As for the computation of I_6, we apply the method proposed in [26] for the evaluation of I_8 the number of cycles of length 8. The girth of the decoding graph is $g = 6$. As a result, I_8 is given by :

$$\begin{array}{l} I_8 = \dfrac{1}{8} tr(L_8) \\ L_8 = P_5^U E^T A - L_{(0,6)}^U A - L_{(4,2)}^U A - L_{(1,6)}^U E^T \\ -L_{(5,2)}^U E^T \end{array} \quad (23)$$

where P_5, $L_{(0,6)}^U$, $L_{(4,2)}^U$, $L_{(1,6)}^U$ and $L_{(5,2)}^U$ are matrices defined in [26], and $A = EE^T$. The trace of these matrices can be computed, as it has been done in the Appendix A for I_6:

$$\begin{array}{l} I_8 = \dfrac{1}{8}\left(tr(A^4) + \alpha_3 tr(A^3) + \alpha_2 tr(A^2) + \alpha_1 tr(A)\right) \\ \alpha_3 = 4(2 - Kt - t) \\ \alpha_2 = (5K^2 + 8K + 5)t^2 - 17(K+1)t + 15 \\ \alpha_1 = -(2K^3 + 3K^2 + 3K + 2)t^3 + 2(4K^2 + 5K + 4)t^2 \\ -10(K+1)t + 4 \end{array} \quad (24)$$

$tr(A) = KNt$ and $tr(A^2)$ is given by (22). These results are integrated in (23) to give (13).

C. List of trinomials

The coefficients of the parity check trinomials $g_a(x) = 1 + x^{i_a} + x^{r_a}$ that have been used for the performance assessment are given in Table V. For each trinomial, we have checked the associated elementary parity check matrix E_a defined by (3) has full rank $(N - r)$.

Table V
PARITY CHECK TRINOMIALS FOR THE M-SEQUENCE GENERATED BY $p(x) = 1 + x^2 + x^5 + x^8 + x^{11}$.

	r_l	i_l
g_1	49	4
g_2	73	22
g_3	93	56
g_4	98	8
g_5	114	83
g_6	146	44
g_7	186	112
g_8	196	16
g_9	228	166
g_{10}	261	80
g_{11}	372	224
g_{12}	465	136
g_{13}	866	339

REFERENCES

[1] S. W. Golomb, *Shift Register Sequences*, San Francisco, CA, USA, 1967.

[2] M.K. Simon, J. K. Omura, R. A. Scholtz, and B. K. Levitt, *Spread Spectrum Communications Handbook*, New York, USA, 1994.

[3] E.D. Kaplan and C.J. Hegarty, *Understanding GPS: Principles and Applications*, Norwood, MA, USA, 2006.

[4] 3GPP TS25.213 v.4.4.0, "3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Spreading and modulation (FDD)," 2004.

[5] A. Polydoros and C. Weber, "A unified approach to serial search spread-spectrum code acquisition–part I: general theory," *IEEE Trans. on Communications*, vol. 32, no. 5, pp. 542–549, 1984.

[6] F.J. MacWilliams and N.J.A. Sloane, *The theory of error-correcting codes*, New-York, USA, 1981.

[7] W. Meier and O. Staffelbach, "Fast correlation attacks on certain stream ciphers," *Journal of Cryptology*, vol. 1, no. 3, pp. 159–176, 1989.

[8] A. Canteaut and M. Trabbia, "Improved fast correlation attacks using parity-check equations of weight 4 and 5," in *Advances in Cryptology - EUROCRYPT 2000*. Springer, 2000, pp. 573–588.

[9] T. Johansson and F. Jonsson, "Improved fast correlation attacks on stream ciphers via convolutional codes," in *Advances in Cryptology - EUROCRYPT 1999*. Springer, 1999, pp. 347–362.

[10] K.M. Chugg and M. Zhu, "A new approach to rapid PN code acquisition using iterative message passing techniques," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 5, pp. 884–897, 2005.

[11] F. Principe, K.M. Chugg, and M. Luise, "Performance evaluation of message-passing-based algorithms for fast acquisition of spreading codes with application to satellite positioning," in *NAVITEC, Noordwijk, The Netherlands*, December 2006.

[12] R. Kerr and J. Lodge, "Iterative signal processing for blind code phase acquisition of CDMA 1x signals for radio spectrum monitoring," *Journal of Electrical and Computer Engineering*, vol. 2010, pp. 3, 2010.

[13] M. des Noes, V. Savin, L. Ros, and J.M. Brossier, "Blind identification of the scrambling code of a reverse link CDMA 2000 transmission," in *International Conference on Communications (ICC)*, Budapest, Hungary, 2013, IEEE.

[14] M. des Noes, V. Savin, L. Ros, and J.M. Brossier, "Blind identification of the uplink scrambling code index of a WCDMA transmission and application to femtocell networks," in *International Conference on Communications (ICC)*, Budapest, Hungary, 2013, IEEE.

[15] M. des Noes, V. Savin, L. Ros, and J.M. Brossier, "Iterative decoding of gold sequences," in *International Conference on Communications, London, UK*. IEEE, 2015.

[16] M. des Noes, V. Savin, L. Ros, and J. M. Brossier, "Improving the decoding of m-sequences by exploiting their decimation property," in *Proceedings of the 21st European Signal Processing Conference (EUSIPCO)*. IEEE, 2013, pp. 1–5.

[17] N. Weinberger and N. Merhav, "Codeword or noise? exact random coding exponents for joint detection and decoding," *IEEE Trans. on Information Theory*, vol. 60, no. 9, pp. 5077–5094, 2014.

[18] F.R. Kschischang, B.J. Frey, and H.A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. on Information Theory*, vol. 47, no. 2, pp. 498–519, 2001.

[19] O.W. Yeung and K.M. Chugg, "An iterative algorithm and low complexity hardware architecture for fast acquisition of long PN codes in UWB systems," *The Journal of VLSI Signal Processing*, vol. 43, no. 1, pp. 25–42, 2006.

[20] F. Principe, K.M. Chugg, and M. Luise, "Rapid acquisition of Gold codes and related sequences using iterative message passing on redundant graphical models," in *Proceedings of the IEEE Military Communications Conference (MILCOM'06)*, 2006, pp. 1–7.

[21] S. Maitra, K. C. Gupta, and A. Venkateswarlu, "Results on multiples of primitive polynomials and their products over GF(2)," *Theoretical Computer Science*, vol. 341, no. 1, pp. 311–343, 2005.

[22] M.A. Richards, *Fundamentals of radar signal processing*, New-York, USA, 2005.

[23] L. Dolecek, P. Lee, Z. Zhang, V. Anantharam, B. Nikolic, and M. Wainwright, "Predicting error floors of structured LDPC codes: Deterministic bounds and estimates," *IEEE Journal on Selected Areas in Communications*, vol. 27, no. 6, pp. 908–917, 2009.

[24] T. Richardson, "Error floors of LDPC codes," in *Proceedings of the annual Allerton conference on communication control and computing*. The University; 1998, 2003, vol. 41, pp. 1426–1435.

[25] Z. Zhang, L. Dolecek, B. Nikolic, V. Anantharam, and M. Wainwright, "Investigation of error floors of structured low-density parity-check codes by hardware emulation," in *Global Telecommunications Conference (GLOBECOM'06)*. IEEE, 2006, pp. 1–6.

[26] T. R. Halford and K. M. Chugg, "An algorithm for counting short cycles in bipartite graphs," *IEEE Trans. on Information Theory*, vol. 52, no. 1, pp. 287–292, 2006.

[27] R.L. Peterson, R.E. Ziemer, and D.E. Borth, *Introduction to Spread-Spectrum Communications*, Prentice Hall, 1995.

[28] P. Chose, A. Joux, and M. Mitton, "Fast correlation attacks: An algorithmic point of view," in *Advances in Cryptology - EUROCRYPT 2002*. Springer, 2002, pp. 209–221.

[29] D. Wagner, "A generalized birthday problem," in *Advances in cryptology - CRYPTO 2002*, pp. 288–304. Springer, 2002.

[30] N. Wiberg, *Codes and decoding on general graphs*, Ph.D. dissertation, Linköping University, Sweden, 1996.

[31] L. Dolecek, Z. Zhang, V. Anantharam, M. J Wainwright, and B. Nikolic, "Analysis of absorbing sets and fully absorbing sets of array-based LDPC codes," *IEEE Trans. on Information Theory*, vol. 56, no. 1, pp. 181–201, 2010.

[32] Q. Huang, Q. Diao, S. Lin, and K. Abdel-Ghaffar, "Trapping sets of structured LDPC codes," in *International Symposium on Information Theory Proceedings (ISIT)*. IEEE, 2011, pp. 1086–1090.

[33] O. Milenkovic, E. Soljanin, and P. Whiting, "Asymptotic spectra of trapping sets in regular and irregular LDPC code ensembles," *IEEE Trans. on Information Theory*, vol. 53, no. 1, pp. 39–55, 2007.

[34] R. Gallager, "Low-density parity-check codes," *IRE Trans. on Information Theory*, vol. 8, no. 1, pp. 21–28, 1962.

[35] Y. Han and W. E. Ryan, "Low-floor decoders for LDPC codes," *IEEE Trans. on Communications*, vol. 57, no. 6, pp. 1663–1673, 2009.