Decoding of Interleaved Reed Solomon Codes over Noisy Data

Daniel Bleichenbacher[1], Aggelos Kiayias[2], and Moti Yung[3]

[1] Bell Laboratories,
Murray Hill, NJ, USA
bleichen@bell-labs.com
[2] Department of Computer Science and Engineering,
University of Connecticut, Storrs, CT, USA
aggelos@cse.uconn.edu
[3] Department of Computer Science,
Columbia University, New York, NY, USA
moti@cs.columbia.edu

Abstract. We consider error-correction over the Non-Binary Symmetric Channel (NBSC) which is a natural probabilistic extension of the Binary Symmetric Channel (BSC). We propose a new decoding algorithm for interleaved Reed-Solomon Codes that attempts to correct all "interleaved" codewords simultaneously. In particular, interleaved encoding gives rise to multi-dimensional curves and more specifically to a variation of the Polynomial Reconstruction Problem, which we call Simultaneous Polynomial Reconstruction. We present and analyze a novel probabilistic algorithm that solves this problem. Our construction yields a decoding algorithm for interleaved RS-codes that allows efficient transmission arbitrarily close to the channel capacity in the NBSC model.

1 Introduction

Random noise assumptions have been considered extensively in the coding theory literature with substantial results. One prominent example is Forney Codes [For66] that were designed over the binary symmetric channel (BSC). The BSC suggests that when transmitting binary digits, errors are independent and every bit transmitted has a fixed probability of error. The BSC provides a form of a random noise assumption, which allows probabilistic decoding for message rates that approach the capacity of the channel.

Worst-case non-ambiguous decoding (i.e., when only a bound on the number of faults is assumed and a unique solution is required) has a natural limitation of correcting a number of errors that is up to half the distance of the code. Going beyond this natural bound, either requires re-stating the decoding problem (e.g. consider list-decoding: output all possible decodings for a corrupted codeword), or assuming some "noise assumption" that will restrict probabilistically the combinatorial possibilities for a multitude of possible solutions. Typically, such assumptions are associated with physical properties of given channels (e.g.,

J.C.M. Baeten et al. (Eds.): ICALP 2003, LNCS 2719, pp. 97–108, 2003.

bursty noise, etc.). Recent breakthrough results by Guruswami and Sudan in list-decoding ([Sud97,GS98]) showed that decoding beyond the natural error-correction bound is possible in the worst-case, by outputting all possible decodings. Naturally, there are still limitations in the case of worst-case decoding that prohibit the decoding of very high error-rates.

In this work, motivated by the above, we investigate a traditional channel model that is native to the non-binary setting. The channel is called "Non-Binary Symmetric Channel" (NBSC), presented in figure 1.

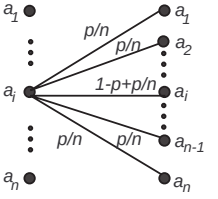

Fig. 1. A non-binary symmetric channel over an alphabet of n symbols. The probability of successful transmission is $1-p+p/n$. We will refer to p as the error-rate of the NBSC.

As a channel model for bit-level transmission the Non-Binary Symmetric Channel model usually applies to settings where aggregates of bits are sent and errors are assumed to be bursty. Thus, in contrast with the Binary Symmetric Channel, errors in consecutive bits are assumed from a Coding Theoretic perspective to be correlated. There are additional situations that have been considered in a number of Computer Science settings where the NBSC describes the transmission model. For example, consider the case of Information Dispersal Algorithms (IDA) introduced by Rabin in [Rab89] for omission errors, and extended by Krawczyk [Kra92] to deal with general errors. In this setting, a word is encoded into a codeword and various portions of the codeword are sent over different radio network channels, some of which may introduce errors. In the case where the channels are operating in different frequencies, errors may be introduced by jammed channels which emit white noise. Namely, they randomize the transmitted symbol. As a result the communication model in this case approximates the NBSC. Another setting which approximates the NBSC is the transmission of encrypted data where each sub codeword is sent encrypted with what is called "error propagation encryption mode." These popular modes (e.g. the CBC mode), over noisy channels, will produce a transmission that also approximates the NBSC model ([MOV96], page 230). Moreover the NBSC model has been used in the cryptographic setting as a way to hide information in schemes that employ intractability assumptions related to the hardness of decoding, see e.g. [KY01].

In this work we concentrate on Reed-Solomon Codes. The decoding problem of Reed-Solomon Codes (aka the Polynomial Reconstruction problem — PR) has

been studied extensively, see e.g. [Ber96,Sud97,GS98]. Here, we present a variation of the PR, which we call "Simultaneous Polynomial Reconstruction" and we present a novel probabilistic algorithm that solves it for settings of the parameters that are beyond the currently known solvability bounds for PR (without any effect on the solvability of the latter problem). Our algorithm is probabilistic and is employed in settings where errors are assumed to be random.

Next we concentrate on the "code interleaving" encoding schema, see e.g. section 7.5, [VV89], which is a technique used to increase the robustness of a code in the setting of burst errors. We consider the problem of decoding interleaved Reed-Solomon Codes and we discover the relationship of this problem to the problem of Simultaneous Polynomial Reconstruction. In particular we show that the two problems are equivalent when interleaved Reed-Solomon Codes are applied over a channel that satisfies the NBSC model.

Subsequently using our algorithm for Simultaneous Polynomial Reconstruction we present a novel decoding algorithm for interleaved Reed-Solomon Codes in the NBSC model that is capable of correcting any error-rate up to $\frac{r}{r+1}(1-\kappa)$ where r is the "amount of interleaving" and κ is the message rate.

We observe that traditional decoding of interleaved RS-Codes does not improve the error-rate that can be corrected. In fact, error-rates only up to $\frac{1-\kappa}{2}$ can be corrected (uniquely) in the worst-case, and in the NBSC model list-decoding algorithms ([GS98]) for unique decoding can be also employed thus correcting error-rates up to $1-\sqrt{\kappa}$.

Nevertheless using our algorithm for Simultaneous Polynomial Reconstruction we correct error-rates up to $\frac{r}{r+1}(1-\kappa)$ (with high probability). An immediate corollary is that we can correct any error-rate bounded away from $(1-\kappa)$ provided that the alphabet-size is selected to be large enough. In other words, interleaved RS-Codes reach the channel's capacity as the amount of interleaving $r \to \infty$ (something that requires that the alphabet-size n over which the NBSC model is employed to also satisfy $n \to \infty$).

Organization. In section 2 we present our variation of the Polynomial Reconstruction problem and we describe and analyze a probabilistic algorithm that solves this problem. Subsequently in section 3 we describe the relation of this problem to the decoding of Interleaved Reed-Solomon codes and we show how our algorithm is employed in this domain. We use the notation $[n]$ to denote the set $\{1,\ldots,n\}$.

2 The Algorithm

In this section we present a probabilistic algorithm that solves efficiently the following problem, which we call the Simultaneous Polynomial Reconstruction:

Definition 1. *(Simultaneous Polynomial Reconstruction — SPR) For $n,k,t,r \in \mathbb{N}$, an instance of SPR is a set of tuples $\{\langle z_i, y_{i,1},\ldots,y_{i,r}\rangle\}_{i=1}^{n}$ over a finite field $\mathbb{F}$ with $i \neq j \to z_i \neq z_j$ that satisfies the following:*

1. *There exists an $I \subseteq [n]$ with $|I| = t$, and polynomials $p_1,\ldots,p_r \in \mathbb{F}[x]$ of degree less than k, such that $p_\ell(z_i) = y_{i,\ell}$ for all $i \in I$ and $\ell \in [r]$.*

2. For all $i \notin I, \ell \in [r]$ it holds that $y_{i,\ell}$ are uniformly distributed over $\mathbb{F}$.
Goal: Recover $p_1, \ldots, p_r$.

We remark that the goal of Simultaneous Polynomial Reconstruction, assuming a large underlying finite-field $\mathbb{F}$, is well-defined (in other words the probability that another tuple of r polynomials $p'_1, \ldots, p'_r$ exists that would fit the data in the same way $p_1, \ldots, p_r$ do, is very small). Taking this into account, the SPR problem with parameters n, k, t, r reduces easily to the Polynomial Reconstruction Problem with parameters n, k, t, (by simply reducing the n tuples to pairs by discarding $r-1$ coordinates — it follows easily that the recovery of p_1 would reveal the remaining polynomials). Thus, we would be interested in algorithmic solutions for the SPR problem when the parameters n, k, t are selected to be beyond the state-of-the-art solvability of the PR problem.

2.1 Description of the Algorithm

The algorithmic construction that we present amends the prototypical decoding paradigm (fitting the data through an error-locator polynomial, see e.g. [BW86, Ber96]) to the setting of Simultaneous Polynomial Reconstruction. More specifically our algorithm can be seen as a generalization of the Berlekamp-Welch algorithm for Reed-Solomon Decoding, [BW86]. The parameter settings where our algorithm works is

$$t \geq \frac{n + rk}{r + 1}$$

observe that for $r = 1$ the above bound on t coincides with the bound of the [BW86]-algorithm, whereas when $r > 1$ less agreement is required (t is allowed to be smaller).

Let $\{\langle z_i, y_{i,1}, \ldots, y_{i,r}\rangle\}_{i=1}^n$ be an instance of the SPR problem with parameters n, k, t, r. Further observe that the condition on t above implies that $r \geq \frac{n-t}{t-k}$.

Define the following system of rn equations:

$$[m_1(z_i) = y_{i,1}E(z_i)]_{i=1}^n \ldots [m_r(z_i) = y_{i,r}E(z_i)]_{i=1}^n \qquad (*)$$

where the unknowns are the coefficients of the polynomials $m_1, \ldots, m_r, E$. Each m_ℓ is a polynomial of degree less than $n - t + k$ and E is a polynomial of degree at most $n - t$ with constant term equal to 1. It follows that the system has $r(n - t + k) + n - t$ unknowns and thus it is not underspecified (i.e., the number of equations is at least as large as the number of unknowns); this follows from the condition on r.

Our algorithm for SPR simply solves system $(*)$ to recover the polynomials $m_1, \ldots, m_r, E$ and outputs $m_1/E, \ldots, m_r/E$ as the solution to the given SPR instance. This is accomplished by selecting an appropriate square sub-system of $(*)$ defined explicitly in section 2.3.

This completes the description of our algorithm. We argue about its correctness in the following two sections. We remark that the novelty of our approach relies on the probabilistic method that is employed to ensure the uniqueness of the error-locator polynomial E.

2.2 Feasibility

In this section we argue that for a given SPR instance $\{\langle z_i, y_{i,1}, \ldots, y_{i,r}\rangle\}_{i=1}^{n}$, one of the possible outputs of the algorithm of section 2.1 is the solution of the SPR instance. Observe that due to item 1 of definition 1, there exists $I \subseteq [n]$ with $|I| = t$ such that $p_\ell(z_i) = y_{i,\ell}$ for $i \in I$ and all $\ell \in [r]$ for some polynomials $p_1, \ldots, p_r \in \mathbb{F}[x]$ (which constitute the solution of the SPR instance).

Let $\tilde{E}(x) = (-1)^{n-|I|} \prod_{i \notin I}(x/z_i - 1)$. Observe that $\tilde{E}$ has constant term 1 and degree $n-t$. Further, if $\tilde{m}_\ell(x) := p_\ell(x)\tilde{E}(x)$ it holds that $\tilde{m}_\ell(z_i) = p_\ell(z_i)\tilde{E}(z_i) = y_{i,\ell}\tilde{E}(z_i)$, for all $i = 1, \ldots, n$. The degree of $\tilde{m}_\ell$ is less than $n - t + k$. Observe that the polynomials $\tilde{E}, \tilde{m}_1, \ldots, \tilde{m}_r$ constitute a possible solution of the system $(*)$. Moreover (by construction) $\tilde{m}_\ell(x)/\tilde{E}(x) = p_\ell(x)$ for $\ell = 1, \ldots, r$ and as a result one of the possible outputs of the algorithm of section 2.1 is indeed the solution of the given SPR instance.

2.3 Uniqueness

The crux of the analysis of our algorithm is the technique we introduce to show the uniqueness of the solution constructed in the previous section.

In a nutshell we will present a technique for constructing a minor for the matrix of system $(*)$ that is non-singular with high probability. It is exactly at this point that item 2 of definition 1 will be employed in a non-trivial manner. We present the technique as part of the proof of the theorem below. The reader is also referred to figure 2 for a graphical representation of the method.

Theorem 1. *The matrix of the linear system $(*)$ has a minor of order $r(n - t + k) + n - t$ denoted by $\hat{A}$ that is non-singular with probability at least $1 - \frac{n-t}{|\mathbb{F}|}$.*

Proof. Consider the following matrices, for $\ell = 1, \ldots, r$:

$$M = \begin{pmatrix} 1 & z_1 & z_1^2 & \ldots & z_1^{n-t+k-1} \\ 1 & z_2 & z_2^2 & \ldots & z_2^{n-t+k-1} \\ \vdots & \vdots & \vdots & \ldots & \vdots \\ 1 & z_n & z_n^2 & \ldots & z_n^{n-t+k-1} \end{pmatrix} \quad M_\ell = \begin{pmatrix} y_{1,\ell} z_1 & y_{1,\ell} z_1^2 & \ldots & y_{1,\ell} z_1^{n-t} \\ y_{2,\ell} z_2 & y_{2,\ell} z_2^2 & \ldots & y_{2,\ell} z_2^{n-t} \\ \vdots & \vdots & \ldots & \vdots \\ y_{n,\ell} z_n & y_{n,\ell} z_n^2 & \ldots & y_{n,\ell} z_n^{n-t} \end{pmatrix}$$

Given these definitions, it follows that the matrix of the system $(*)$ is the following (where $\mathbf{0}$ stands for a $n \times (n - t + k)$-matrix with 0's everywhere):

$$A = \begin{pmatrix} M & \mathbf{0} & \ldots & \mathbf{0} & -M_1 \\ \mathbf{0} & M & \ldots & \mathbf{0} & -M_2 \\ \vdots & \vdots & \ldots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \ldots & M & -M_r \end{pmatrix}$$

We index each row of A by the pair $\langle i, \ell\rangle$ with $i \in \{1, \ldots, n\}$ and $\ell \in \{1, \ldots, r\}$. The ℓ-th block row of A contains the rows $\langle 1, \ell\rangle, \ldots, \langle n, \ell\rangle$.

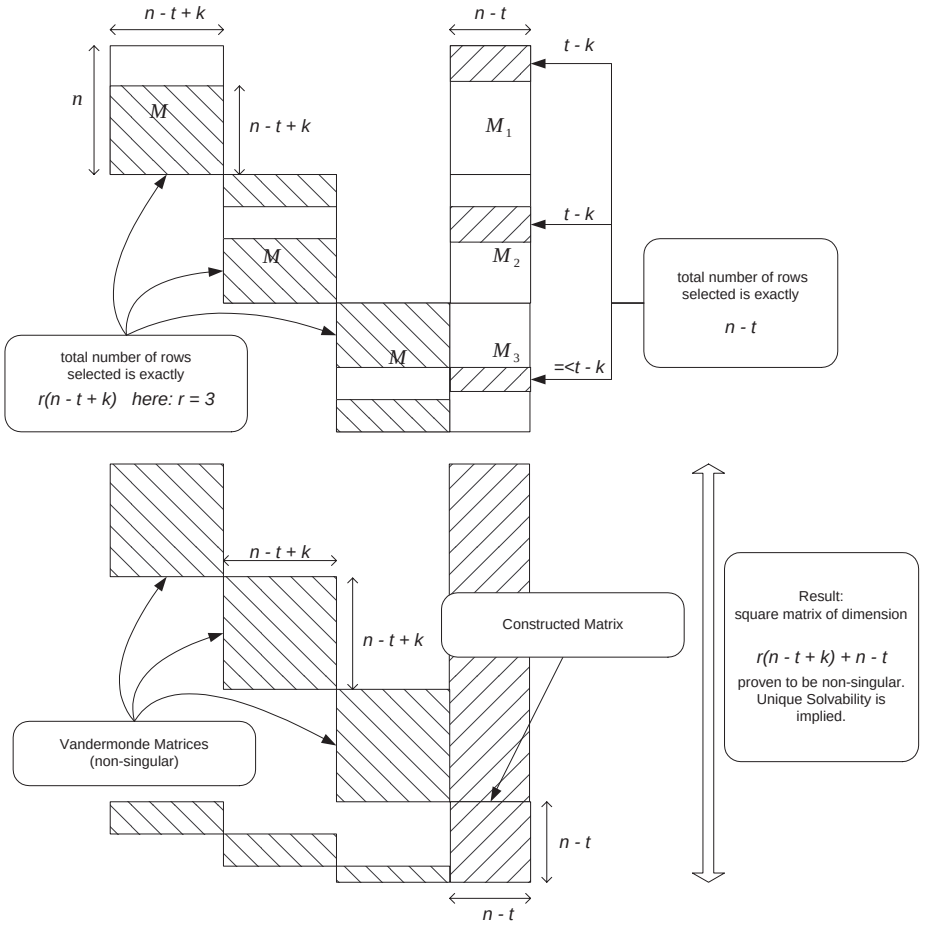

Fig. 2. Constructing the matrix $\hat{A}^*$ from the matrix of the system $(*)$. Refer to the proof of theorem 1 for the definitions of the matrices shown above.

Now we select a square sub-matrix $\hat{A}$ of A by removing $r(t-k)-(n-t)$ rows as follows: starting from the r-th block row we remove a number of rows $x \in \{0, \ldots, t-k\}$ indexed by $\langle n, r\rangle, \ldots, \langle n-t+k+1, r\rangle$ (in this order) until $\hat{A}$ becomes square or x reaches $t-k$. Then we repeat the same procedure for the block-row $(r-1)$ and so on, until $\hat{A}$ becomes square. Next, we will show that $\hat{A}$ is non-singular with high probability.

Without loss of generality we assume that $I = \{n-t+1, \ldots, n\}$. The proof is identical for any other choice of I.

Now let us denote by N_ℓ a $(n-t+k)$-Vandermonde matrix over the elements $\{z_1, \ldots, z_n\} - \{z_{1+(\ell-1)(t-k)}, \ldots, z_{\ell(t-k)}\}$. Also we define M'_ℓ to be the sub-matrix of M_ℓ with the rows $\langle x+(\ell-1)(t-k), \ell\rangle$ removed for $x = 1, \ldots, t-k$. Finally let

V_ℓ be a $(n-t)\times(n-t+k)$-matrix that is 0 everywhere except for the rows u that satisfy the property that there is an $x \in [t-k]$ such that $u = x+(\ell-1)(t-k) \leq n-t$; such a row of V_ℓ will be equal to the tuple $\langle 1, z_u, \ldots, z_u^{n-t+k-1}\rangle$. The matrix $\hat{A}^*$ defined below is a rearrangement of the rows of $\hat{A}$.

$$\hat{A}^* = \begin{pmatrix} N_1 & \mathbf{0} & \ldots & \mathbf{0} & -M_1' \\ \mathbf{0} & N_2 & \ldots & \mathbf{0} & -M_2' \\ \vdots & \vdots & \ldots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \ldots & N_r & -M_r' \\ V_1 & V_2 & \ldots & V_r & -\hat{M} \end{pmatrix}$$

where the right low corner matrix $\hat{M}$ is defined below:

$$\hat{M} = \begin{pmatrix} y_{1,1}z_1 & y_{1,1}z_1^2 & \cdots & y_{1,1}z_1^{n-t} \\ y_{2,1}z_2 & y_{2,1}z_2^2 & \cdots & y_{2,1}z_2^{n-t} \\ \vdots & \vdots & \ldots & \vdots \\ y_{t-k,1}z_{t-k} & y_{t-k,1}z_{t-k}^2 & \cdots & y_{t-k,1}z_{t-k}^{n-t} \\ y_{t-k+1,2}z_{t-k+1} & y_{t-k+1,2}z_{t-k+1}^2 & \cdots & y_{t-k+1,2}z_{t-k+1}^{n-t} \\ y_{t-k+2,2}z_{t-k+2} & y_{t-k+2,2}z_{t-k+2}^2 & \cdots & y_{t-k+2,2}z_{t-k+2}^{n-t} \\ \vdots & \vdots & \ldots & \vdots \\ y_{2(t-k),2}z_{2(t-k)} & y_{2(t-k),2}z_{2(t-k)}^2 & \cdots & y_{2(t-k),2}z_{2(t-k)}^{n-t} \\ \vdots & \vdots & \ldots & \vdots \\ y_{n-t,\ell}z_{n-t} & y_{n-t,\ell}z_{n-t}^2 & \cdots & y_{n-t,\ell}z_{n-t}^{n-t} \end{pmatrix}$$

We will argue that $\hat{A}^*$ is non-singular. First observe that the determinant of $\hat{A}^*$ can be seen as a multivariate polynomial over the variables $y_{i,\ell}$ where $i \in [n]$ and $\ell \in [r]$ (taking into account the fact that $y_{i,\ell}$ for $i \in I$ are only k-wise independent — note that without loss of generality we assume that the solution of a SPR instance is uniformly random: indeed given a SPR instance we can easily randomize its solution by adding a random polynomial of degree less than k to each of the r coordinates; naturally, if a solution is found we will have to subtract the randomization polynomial from each coordinate).

Suppose now we want to eliminate V_1. In particular to eliminate the first non-zero row of V_1 we should find $\lambda_{t-k+1}, \ldots, \lambda_n$ such that $\sum_{j=t-k+1}^{n} \lambda_j z_j^m = -z_1^m$ for each $m \in [n-t+k-1] \cup \{0\}$.

Now let us choose some assignment for the values $y_{1,1}, \ldots, y_{n,1}$; we set $y_{1,1} = \ldots = y_{t-k,1} = 2$ and $y_{t-k+1,1} = \ldots = y_{n,1} = 1$. It follows that the first row of $\hat{M}$ is rewritten as $\langle 2z_1, \ldots, 2z_1^{n-t}\rangle$. It follows that after the elimination of the first row of V_1 the first row of $\hat{M}$ becomes equal to $\langle z_1, \ldots, z_1^{n-t}\rangle$.

Regarding the step above, observe the following: (i) the assignment we did for the $y_{i,\ell}$ values is consistent with their dependency condition: $y_{n-t+1,\ell}, \ldots, y_{n,\ell}$ must be k-wise independent; (ii) by applying the same elimination method to the remaining non-zero rows of $V_1, V_2, \ldots, V_r$ and for each $\ell \in [r]$ making the assignment $y_{i,\ell} = 2$ for each $i \in \{x+(\ell-1)(t-k) \leq n-t \mid x = 1, \ldots, t-k\}$ and

$y_{i,\ell} = 1$ otherwise, it follows that we will eliminate all $V_1, \ldots, V_r$. After this is accomplished observe that in place of the matrix $\hat{M}$ there will be a Vandermonde-like matrix of order $n - t$ that is non-singular.

It follows that $\det(\hat{A}^*)$ (seen as a multivariate-polynomial) is not the zero-polynomial and thus, by Schwartz's Lemma [Sch80], it cannot be 0 in more than a $\frac{n-t}{|\mathbb{F}|}$-fraction of its domain (where $n - t$ is the total degree of the polynomial $\det(\hat{A}^*)$). As a result $\det(\hat{A}^*)$ will be 0 with probability at most $\frac{n-t}{|\mathbb{F}|}$. ■

It follows easily from the above theorem that the system $(*)$ accepts at most one solution. Naturally the non-singularity of $\hat{A}$ is not sufficient to ensure the existence of a solution. Nevertheless we know that $(*)$ accepts at least one solution (as constructed explicitly in section 2.2). It follows that system $(*)$ has a unique solution (that coincides with the solution constructed in section 2.2) and this solution can be found by solving the system that has $\hat{A}$ as its matrix.

To improve the efficiency of our algorithm observe that is is not necessary to solve the linear-system with matrix $\hat{A}$ directly; instead, we can derive easily a system of $n - t$ equations that completely determines the polynomial E; it is obvious that the recovery of E will reveal all solutions of the given SPR instance. This is so, since finding all roots of E will reveal the error-locations of the given SPR-instance and then the recovery of $p_1, \ldots, p_r$ can be done by interpolation. A system of $n - t$ equations that determines E completely can be found by eliminating all variables that correspond to the polynomials m_ℓ from at most $t - k$ rows of the ℓ-th block row of matrix $\hat{A}$, for $\ell = 1, \ldots, r$. Such elimination will be possible for exactly $n - t$ rows.

3 Decoding Interleaved RS-Codes in the NBSC Model

In this section we present a coding theoretic application of our algorithm of section 2 to the case of interleaved Reed-Solomon Decoding. First we recall the notion of interleaved codes.

3.1 Interleaved Codes

Interleaved codes are not an explicit family of codes, but rather an encoding mode that can be instantiated over any concrete family of codes. In The mode can be applied to any family of codes; in this section we give a code independent description.

Let Σ' be an alphabet with $|\Sigma'| = \sqrt[r]{|\Sigma|}$. Let $\phi : \Sigma \to (\Sigma')^r$ be some 1-1 mapping. We will denote $\phi(x)$ by the string $x^\phi[1]x^\phi[2]\ldots x^\phi[r]$, where $x^\phi[\ell] \in \Sigma'$, for $\ell = 1, \ldots, r$, for any $x \in \Sigma$.

Now let $enc : (\Sigma')^k \to (\Sigma')^n$ be an encoding function. An interleaved code w.r.t. ϕ for enc is a function $enc_\phi : (\Sigma)^k \to (\Sigma)^n$ that is defined as follows: Let $m_0 m_1 \ldots m_{k-1} \in (\Sigma)^k$. First the following strings of $(\Sigma')^n$ are computed:

$$c_{1,1} \dots c_{n,1} = enc(m_0^{\phi}[1] \dots m_{k-1}^{\phi}[1])$$

$$\vdots$$

$$c_{1,r} \dots c_{n,r} = enc(m_0^{\phi}[r] \dots m_{k-1}^{\phi}[r])$$

The interleaved encoding is defined as follows:

$$enc_{\phi}(m_0 m_1 \dots m_{k-1}) = \phi^{-1}(c_{1,1} \dots c_{1,r}) \dots \phi^{-1}(c_{n,1} \dots c_{n,r})$$

A graphical representation of code interleaving is presented in figure 3.

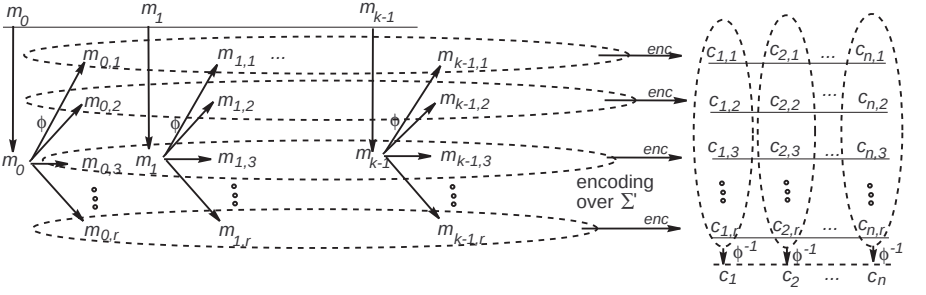

Fig. 3. Encoding schema for an interleaved code. Single subscript symbols (m_i, c_i) belong to the "outer" alphabet Σ; double subscript symbols $(m_{i,j}, c_{i,j})$ belong to the "inner" alphabet Σ'.

Such interleaved encodings will be said to be of degree r over the alphabet Σ' (we will also call it "amount of interleaving").

The common way to use an interleaved code, is simply decode each of the code words $(c_{1,i} \dots c_{n,i})$ separately. Such a decoding does not increase the error correction rate. The advantage is the fact that burst errors are distributed over several code words, and therefore employing interleaving over bursty channels increases the chances of error-correction.

We emphasize here that under reasonable channel assumptions it might be possible to take advantage of interleaving and attempt to correct all code words simultaneously. Indeed, in contrast to the standard approach of decoding each one of the codewords individually, we will present a decoding technique that attempts to correct all codewords simultaneously assuming that the NBSC model describes the transmission channel. This methodology will increase the possible error-rates that the interleaved code can withstand.

3.2 Interleaved Reed-Solomon Codes

Let $\Sigma = GF(2^B)$ be the alphabet for the encoding function (without loss of generality we will focus only on binary extension fields — all our results hold

also for general finite fields). The parameters are $n, k \in \mathbb{N}$ where $\kappa := k/n$ is the message rate. We assume additionally a parameter $r \in \mathbb{N}$ with the property $br = B$ (we remark here that a similar scheme is also possible when B is prime, however, for notational simplicity we do not deal with this case in this abstract). Let $z_1, \ldots, z_n \in GF(2^b)$ be fixed distinct constants.

We now describe the case of interleaved Reed-Solomon Codes. First, observe that there exists a straightforward bijection mapping $\phi : GF(2^B) \to (GF(2^b)^r$. Given $m_0 \ldots m_{k-1} \in GF(2^B)$ we define the following polynomials over $GF(2^b)$, for $\ell = 1, \ldots, r$:

$$p_\ell(x) := m_0^\phi[\ell] + m_1^\phi[\ell]x + \ldots + m^\phi[\ell]_{k-1}x^{k-1}$$

The encoding of $m_0 \ldots m_{k-1}$ is set to be the string

$$\phi^{-1}(p_1(z_1) \ldots p_r(z_1)) \ldots \phi^{-1}(p_1(z_n) \ldots p_r(z_n))$$

The common way to decode RS-interleaved-codes is to concentrate to each of the r coordinates individually and employ the decoding algorithm of the underlying RS-Code over Σ'. This can be done as follows: given a (partially corrupted) codeword $c_1 \ldots c_n \in (\Sigma)^n$ we treat the string $c_1^\phi[1] \ldots c_n^\phi[1] \in (\Sigma')^n$ as a partially corrupted RS-codeword over Σ' and we employ the RS-Decoding of Berlekamp-Welch to recover p_1. Observe that the recovery of p_1 will imply the recovery of $p_2, \ldots, p_r$ immediately, provided that the error-rate is at most $\frac{1-\kappa}{2}$ (the error-rate is taken over the channel that transmits $GF(2^B)$ symbols; it is easy to verify that in the NBSC model all codewords $c_1^\phi[\ell] \ldots c_n^\phi[\ell]$, $\ell = 1, \ldots, r$ will have identical error-pattern with very high probability).

Moreover, due to assured unique solution with high probability in our case, one can further employ the Guruswami-Sudan list-decoding algorithm that will produce a unique solution with high probability for error-rates up to $1 - \sqrt{\kappa}$. The main focus of the next section is to go beyond this bound.

3.3 The Decoding Algorithm

In this section we reduce the problem of decoding interleaved Reed-Solomon Codes in the NBSC model to the problem of Simultaneous Polynomial Reconstruction. Given this result, our algorithm for the latter problem yields a decoding algorithm for interleaved RS-codes.

Consider interleaved RS-Codes with parameters $r, n, k, t \in \mathbb{N}$, where r is the amount of interleaving. Also let $\phi : GF(2^B) \to GF(2^b)^r$ be the bijection mapping employed for the interleaving.

Let $c_1 \ldots c_n \in (GF(2^B))^n$ be the received codeword. Let $y_{i,1} \ldots y_{i,r} = \phi(c_i)$, with $y_{i,\ell} \in GF(2^b)$ for all $i = 1, \ldots, n, \ell = 1, \ldots, r$.

Suppose now that $i \in \{1, \ldots, n\}$ is an error-location for the codeword $c_1 \ldots c_n$. It follows that c_i is uniformly distributed over $GF(2^B)$ (because of the employment of the NBSC model). Since ϕ is a bijection it follows easily that each of $y_{i,1}, \ldots, y_{i,r}$ are uniformly distributed over $GF(2^b)$.

On the other hand there exist polynomials $p_1, \ldots, p_r \in GF(2^b)[x]$ of degree less than k such that for all $i \in \{1, \ldots, n\}$ with i not an error-location, it holds that $y_{i,1} = p_1(z_i), \ldots, y_{i,r} = p_r(z_i)$. The following proposition is immediate:

Proposition 1. *Let $c_1 \ldots c_n \in GF(2^B)^n$ be an encoding of a message $m_0 \ldots m_{k-1} \in GF(2^B)^k$ using the interleaved Reed-Solomon encoding scheme with parameters n, k, r that has e errors (over the NBSC model). Then the tuples $\{\langle z_i, y_{i,1}, \ldots, y_{i,r}\rangle\}_{i=1}^n$ as defined above constitute an instance of the SPR problem with parameters $n, k, t := n - e, r$ over the field $GF(2^b)$, $b = B/r$.*

Based on our algorithm of section 2 we deduce:

Corollary 1. *There exists a decoding algorithm for interleaved Reed-Solomon codes over parameters n, k, r that corrects any error-rate ϵ up to*

$$\epsilon \leq \frac{r}{r+1}(1 - \kappa)$$

with probability $1 - \frac{n-t}{2^b}$.

Example: Suppose that the message-rate is $1/4$ and the error-rate is $11/16$. We employ the interleaved RS-schema for $r = 11$ with alphabets $\Sigma = GF(2^B) = GF(2^{440})$ and $\Sigma' = GF(2^b) = GF(2^{40})$. Observe that such error-rates are not correctable by considering the interleaved codewords individually (indeed, even list-decoding algorithms, e.g. the [GS98]-method would work only for error-rates up to $1/2$). Suppose now that the block-size is $n = 64$. Our probabilistic decoding algorithm for such interleaved RS-codes corresponds to solving the SPR problem on parameters $n = 64$, $k = 16$, $t = 20$, $r = 11$ over the finite-field $GF(2^{40})$ and thus we will succeed in decoding with probability least $1 - 2^{-34}$.

Remark: We note that employing our methodology, setting and analysis techniques in other cases (i.e. simultaneous decoding of all interleaved codewords for other families of interleaved codes in the NBSC model) is an interesting research direction.

An independent solution of the Simultaneous Polynomial Reconstruction Problem was presented recently by Coppersmith and Sudan in [CS03]. Their solution requires $t > \sqrt[r+1]{nk^r} + k + 1$ which improves on our bound $t \geq \frac{n+rk}{r+1}$ in cases where $t > 2k$.

Acknowledgement. The authors wish to thank Alexander Barg for helpful discussions.

References

[Ber96] Elwyn R. Berlekamp, *Bounded distance+1 soft-decision Reed-Solomon decoding*, IEEE Trans. Info. Theory, vol. IT-42, pp. 704–720, May 1996.

[BW86] Elwyn R. Berlekamp and L. Welch, *Error Correction of Algebraic Block Codes*. U.S. Patent, Number 4,633,470, 1986.

[CS03] Don Coppersmith and Madhu Sudan, *Reconstructing Curves in Three (and higher) Dimensional Space from Noisy Data*, to appear in the proceedings of the 35th ACM Symposium on Theory of Computing (STOC), June 9–11, 2003, San Diego, California.

[For66] G. David Forney, *Concatenated Codes*, MIT Press, Cambridge, MA, 1966

[GS98] Venkatesan Guruswami and Madhu Sudan, *Improved Decoding of Reed-Solomon and Algebraic-Geometric Codes.* In the Proceedings of the 39th Annual Symposium on Foundations of Computer Science, IEEE Computer Society, pp. 28–39, 1998.

[KY01] Aggelos Kiayias and Moti Yung, *Secure Games with Polynomial Expressions*, In the Proceedings of the 28th International Colloquium in Algorithms, Languages and Programming (ICALP), 2001, LNCS Vol. 2076, pp. 939–950.

[Kra92] Hugo Krawczyk, *Distributed Fingerprints and Secure Information Dispersal*, PODC 1992, pp. 207–218.

[MS77] F. J. MacWilliams and N. Sloane, *The Theory of Error Correcting Codes.* North Holland, Amsterdam, 1977.

[MOV96] Alfred J. Menezes, Paul C. van Oorschot and Scott A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1996.

[Rab89] Michael O. Rabin, *Efficient dispersal of information for security, load balancing, and fault tolerance*, J. ACM 38, pp. 335–348, 1989.

[Sch80] J. T. Schwartz, *Fast Probabilistic Algorithms for Verifications of Polynomial Identities*, Journal of the ACM, Vol. 27(4), pp. 701–717, 1980.

[Sud97] Madhu Sudan, *Decoding of Reed Solomon Codes beyond the Error-Correction Bound.* Journal of Complexity 13(1), pp. 180–193, 1997.

[VV89] S. A. Vanstone and P. C. VanOorshot, *An Introduction to Error Correcting Codes with Applications*, Kluwer Academic Publishers, 1989.