Improved Soft Decoding of Reed-Solomon Codes on Gilbert-Elliott Channels

Michel Kulhandjian
School of Elec. Eng. & Comp. Sc.
University of Ottawa
Ottawa, Ontario, K1N 6N5, Canada
Email: mkk6@buffalo.edu

Hovannes Kulhandjian
Dep. of Elec. & Comp. Eng.
California State University
Fresno, CA 93740, U.S.A.
Email: hkulhandjian@csufresno.edu

Claude D'Amours
School of Elec. Eng. & Comp. Sc.
University of Ottawa
Ottawa, Ontario, K1N 6N5, Canada
Email: cdamours@uottawa.ca

***Abstract*—It was shown by Guruswami and Rudra that Reed-Solomon codes can be list decoded to recover from *phased* burst errors (i.e. errors occurring within fixed regular intervals) up to the information-theoretic limit and, in particular, beyond the Guruswami-Sudan bound. In this paper, we present evidence that the algorithm developed by Guruswami and Rudra can also give improvement for more "irregular" burst errors. We develop a low-complexity multiplicity assignment scheme for soft decoding of Reed-Solomon (RS) codes. Specifically, we present simulation results where such soft decoding of RS codes outperforms the existing soft decision decoding algorithms of Koetter and Vardy as well as the algorithm of Das and Vardy on Gilbert-Elliott channels (under QAM and BPSK modulations) for channels that are more bursty. We also present a theoretical result that shows that for certain Gilbert-Elliott channels, with high probability of errors, the output list size for list decoding RS codes is one.**

I. Introduction

Reed-Solomon (RS) codes are perhaps the most studied codes with a wide range of applications in digital communications and storage. In his ground-breaking work, Sudan showed that Reed-Solomon codes can be decoded from worst-case errors beyond the half-the-distance bound under *list decoding*, wherein the decoder is allowed to output a list of candidate codewords with the guarantee that it contains the transmitted codeword [1], [2]. This result was improved by Guruswami and Sudan who showed that Reed-Solomon codes of rates R can be efficiently list decoded from $1-\sqrt{R}$ fraction of errors [3]. For RS codes, this is to date the best known result for worst-case errors. The influential work of Koetter and Vardy [4] gave an algorithm to appropriately define the "multiplicities" for the Guruswami-Sudan algorithm, which led to a soft decision decoding algorithm for RS codes that outperformed existing RS decoding algorithms on certain stochastic channels.

Building on the results of Parvaresh and Vardy [5], Guruswami and Rudra [6] showed that the closely related Folded Reed-Solomon (FRS) codes of rate R can be list decoded up to the information-theoretic optimal $1-R$ fraction of worst-case errors. In particular, the FRS codes with folding parameter m, which are obtained by grouping together m consecutive symbols from RS codewords, can be list decoded from $1-\sqrt[s+1]{(mR/(m-s+1))^s}$ fraction of errors, where $s \leq m$ is a free parameter. Choosing m and s appropriately one can approach arbitrarily close to the bound of $1-R$. Note that the result of Guruswami and Rudra implies the following result for RS codes. If the errors occur in phased bursts of size m, then the algorithm in [6] can correct $1-\sqrt[s+1]{(mR/(m-s+1))^s}$ fraction of worst-case of such bursts.

Burst errors do occur in many information transmission and storage systems [7]. However, the extra restriction of *phased* burst errors in the result mentioned above is not always realistic. A natural question, which motivated our work, is whether the techniques from [6] can be used to (list) decode RS codes to recover from more general burst errors.

In this work, we consider perhaps the most natural channel to investigate the question above: the Gilbert-Elliott (GE) channel [8], [9]. In this model, the channel is described by a two-state Markov chain, where in the "good" state the channel introduces no (or very little) error while in the "bad" state the channel introduces (more) error. If the transition probabilities are small, then this models burst errors. Given the ubiquity of RS codes, in this work we exclusively focus on the performance of RS codes over the GE channel.

To the best of our knowledge, there is no prior work on performance of list decoders for RS codes on the GE channel. Performance analysis of non-interleaved Reed-Solomon codes over the finite-state channels (and in particular the GE channel) was presented in [10]. The analysis is combinatorial and it does not specifically analyze the algorithm performance. There have been other works on the GE channel, e.g., the development of a trellis structure based decoding algorithm that is based on *a posteriori* probability (APP) for linear block codes over a GE channel [11]. There is also a fair amount of literature on decoding RS codes from burst errors, though a lot of them deal with phased burst errors [12]–[15].

In this paper, we adapt the soft decoding algorithm for FRS codes from [6] to decode RS codes. Indeed, any decoding algorithm for FRS codes can be naturally adapted to decode RS codes: given a received word, one can bundle up m consecutive symbols to define an intermediate received word that can then be fed into the FRS decoder. Since there is a natural bijection between an RS code and the corresponding FRS code, the output of the FRS decoder can be interpreted as RS codewords. Note that in the translation above, we can *choose* the parameters m and s. Potentially, one can pick these parameters based on the channel parameters.

The bulk of our results in this article are obtained through simulations for moderate and high rate RS codes of block length 255 (over $GF(256)$). We consider the GE channel under both 256-QAM (quadrature amplitude modulation) (in which case an independent errors caused by additive white Gaussian noise (AWGN) in both the inphase and quadrature components of each modulated QAM symbol) and binary phase-shift keying (BPSK) modulation (in which case the AWGN noise acts on each bit). Our results indicate that the soft decoding algorithm in [6] has a lot of potential for correcting general stochastic burst errors. We would like to stress that in our simulations, we did *not* implement the entire soft decoding algorithm from [6], as the current implementations are not that efficient in practice with decoding complexity $(Nm)^{\mathcal{O}(s)}$. In particular, we developed low-complexity multiplicity assignment scheme and in our simulations we determined whether a decoding error occurred or not by verifying if the transmitted codeword satisfied the successful decoding constraint[1]. We believe our results provide more motivation to design practical implementations of algorithms in [6].

An important matter that we have not discussed so far is the number of codewords that are output by the list decoding algorithm. The theoretical bounds in [6] are polynomially small, though they are still quite large for practical systems. However, the bounds in [6] are for worst-case errors. As a new theoretical contribution of this article, we observe that a recent result on list sizes for random errors [16] and (well-known) results on the mixing time on Markov chains imply that for certain GE channels, with high probability over the random errors, any list decoding algorithm for RS codes (with not too large dimension) needs only to output *one* codeword.

The rest of the paper is organized as follows. In Section II, we discuss the preliminaries. In Section III, we present a theorem regarding the average list size for RS codes. In Section IV, we describe our soft decoding algorithm for RS codes on $\mathrm{GE}_{\mathrm{QAM}}$. In Section V, we present our simulation methodology and results before presenting our conclusions in Section VI.

II. Preliminaries

A. Reed-Solomon and Related Codes

Reed-Solomon codes over $\mathbb{F}_q$ of dimension k and block length $n \leq q$, are defined as follows (for some $S \subseteq \mathbb{F}_q$ with $|S| = n$). We map a message in $\mathbb{F}_q^k$ to a degree $k-1$ polynomial f and the corresponding codeword is the vector $(f(\alpha))_{\alpha \in S}$. A common instantiation is $S = \{1, \gamma, \gamma^2, \ldots, \gamma^{n-1}\}$, where γ is the generator of the multiplicative subgroup of $\mathbb{F}_q$. We will denote this code by $\mathrm{RS}[k, n]$. The m folded version of $\mathrm{RS}[k, n]$, which we will denote by $\mathrm{FRS}[k, n, m]$, is a code of block length $N' = n/m$ over $\mathbb{F}_q^m$, where n is divisible by m. The jth symbol for $0 \leq j < n/m$ consists of the m-tuple $(f(\gamma^{jm}), f(\gamma^{jm+1}), ..., f(\gamma^{jm+m-1}))$ where as before f is the degree $k-1$ polynomial representing the message. The list decoding algorithm for $\mathrm{FRS}[k, n, m]$ proceeds by reducing the task to list decoding a related code $\mathrm{PV}[k, n, m, s]$ for some parameter $s \leq m$, which is a special instantiation of the codes considered by Parvaresh and Vardy [5] and it is defined as follows. For each symbol from $\mathbb{F}_q^m$ in any codeword in $\mathrm{FRS}[k, n, m]$ replace it by $(m-s+1)$ symbols from $\mathbb{F}_q^s$ by "sliding" a window of length s over the $\mathbb{F}_q^m$ symbol. Note that the code has a block length $N = (m-s+1)N' = (m-s+1)n/m$. We will crucially use the following result concerning soft decoding these codes.

Theorem 1 ([6], [5]): Let $1 \leq s \leq m$ be integers and let n be an integer that is divisible by m. Given non-negative integer weights $\{w_{i,\boldsymbol{\alpha}}\}$ for every $1 \leq i \leq N$ and $\boldsymbol{\alpha} \in \mathbb{F}_q^s$, there exists a list decoding algorithm that outputs all the codewords $(c_1, \ldots, c_N) \in \mathrm{PV}[k, n, m, s]$ that satisfy

$$\sum_{i=1}^{N} w_{i,c_i} > \sqrt[s+1]{(k-1)^s W}, \tag{1}$$

where
$$W = (s+1)! \sum_{i=1}^{N} \sum_{\boldsymbol{\beta} \in \mathbb{F}_q^s} \binom{w_{i,\boldsymbol{\beta}} + s}{s+1}. \tag{2}$$

The algorithm runs in polynomial time in q^s and W. □

B. Gilbert-Elliott Channel

We will model burst noise in this paper via the Gilbert-Elliott (GE) noise model. In this model, there are two states: G ("good" state) and B ("bad" state). From state G, the channel can transition into state B with probability b and vice-versa with probability g. With probability $1-b$ ($1-g$ resp.), the channel remains in state G (B resp.). This is a simple two state Markov chain with the following steady state probabilities

$$\pi_G = \frac{g}{g+b} \text{ and } \pi_B = \frac{b}{g+b}.$$

Note that the closer the quantity $1-g-b$ (which in the literature it is referred as the "burst factor" or "channel memory") is to 1, the more "bursty" the channel is. It is easy to check that the channel transition matrix has second eigenvalue equal to $1-g-b$, which along with the Chernoff bound for Markov chains from [17], implies the following result.

Lemma 1 ([17]): Let $g, b, \epsilon > 0$ be real numbers such that $g+b+\epsilon \leq 1$. Let n be a large enough integer and let B_n be the number of "bad" states encountered in a random walk on the GE Markov chain with g and b as transition probabilities. Then,

$$\mathbf{Pr}\left[\left|B_n - \frac{bn}{g+b}\right| > \epsilon n\right] \leq 2e^{-(g+b)\epsilon^2 n}. \qquad \square$$

To complete the description of the GE channel, we need to specify how noise acts in the "good" and "bad" states. We will consider three variations: (i) In the first version, which we refer to as $\mathrm{GE}_{\mathrm{rand}}$, there is no noise in the "good" state, while in the "bad" state, the symbol from $\mathbb{F}_q$ is transformed into one of the other $q-1$ possibilities with equal probability. (ii) In the second version, which we refer to as $\mathrm{GE}_{\mathrm{QAM}}$, in the "good" state the channel adds AWGN with power spectral

[1]Due to this assumption, we determine the decoding error probability in our results rather than the codeword error probability as the latter is generally used when the transmitted codeword is decoded uniquely.

density $\frac{N_G}{2}$. In the "bad" state, the channel adds AWGN with power spectral density $\frac{N_B}{2}$, where $N_B > N_G$. We stress that in this model, the noise always acts at the level of symbols from $\mathbb{F}_q$. (iii) In the third version, which we refer to as $\mathrm{GE}_{\mathrm{BPSK}}$ is applicable when q is a power of 2, the channel is identical to $\mathrm{GE}_{\mathrm{QAM}}$ except in this case the noise acts at the bit level.

In this paper, for $\mathrm{GE}_{\mathrm{QAM}}$ we will be working with $\mathbb{F}_{256}$-QAM modulation and for $\mathrm{GE}_{\mathrm{BPSK}}$ we will be working with $\mathbb{F}_{256}$ BPSK modulation.

III. Average List Size for RS codes on $\mathrm{GE}_{\mathrm{rand}}$

We present the following theorem regarding the average list size for RS codes on the $\mathrm{GE}_{\mathrm{rand}}$ channel of Section II-B:

Theorem 2: Let $g, b, \epsilon, R > 0$ be real numbers such that $g+b+\epsilon \leq 1$ and $b/(b+g)+\epsilon \leq 1-R-\epsilon$. Then, the following results holds for RS codes with rate R and large enough block length n. With all but an exponentially small probability, for the $\mathrm{GE}_{\mathrm{rand}}$ channel with transition probabilities g and b, the transmitted codeword is the only codeword within relative Hamming distance $b/(g+b) - \epsilon$ from the received word. □

From Lemma 1, with all but an exponentially small probability, the fraction of symbols that fall in the "bad" state is given by $\frac{b}{g+b} - \epsilon \leq \rho \leq \frac{b}{g+b} + \epsilon$. Then using a recent result by Rudra and Uurtamo [16] we prove the theorem.

In other words, for instantiations of the $\mathrm{GE}_{\mathrm{rand}}$, where there are $\rho \leq \delta - \epsilon$ fraction of "bad" states (the locations of the "bad" states can be adversarial), for any code (over large enough alphabet) with relative distance δ, with all but an exponentially small probability, only the transmitted codeword is within a relative Hamming distance ρ from the received word. Theorem 2 now follows from the fact that for RS codes of rate R, $\delta \geq 1 - R$ (and that the result from [16] holds for RS codes with large enough block lengths).

IV. The Soft Decoding Algorithm for RS Codes

In this section, we describe our soft decoding algorithm for RS codes on $\mathrm{GE}_{\mathrm{QAM}}$. In what follows, A and r are two integer parameters. We assume that q-QAM modulation is given by the function $\phi : \mathbb{F}_q \to \mathbb{R}^2$. Thus, the received word is a vector $\mathbf{y} = (y_1, \ldots, y_n) \in (\mathbb{R}^2)^n$.

The main idea in the algorithm is to convert the decoding process for $\mathrm{RS}[k, n]$ into one for the corresponding $\mathrm{PV}[k, n, m, s]$ code. In particular, the received word $\mathbf{y}$ is converted into the corresponding received word $\mathbf{y}' \in \left((\mathbb{R}^2)^s\right)^N$ where $N = (m-s+1)n/m$ (i.e., divide $\mathbf{y}$ into blocks of size m and then run a "sliding" window of size s over each block over $(\mathbb{R}^2)^m$). Then, $\mathbf{y}'$ is demodulated to obtain weights for every possible symbol in $\mathbb{F}_q^s$ for every position. These weights are then fed into the algorithm in Theorem 1.

The demodulation is implemented in the following manner. In the first step, for every position $1 \leq i \leq n$, we compute the Euclidean distance from the received symbol y_i to $\phi(\beta)$ for every $\beta \in \mathbb{F}_q$. Let $\mathbf{d}_i \in \mathbb{R}^r$ denote the vector of the r smallest of such distances[2]. Then, for each position $1 \leq j \leq N$ for $\mathbf{y}'$, we compute the r smallest Euclidean distances (over $\mathbb{R}^{2s}$) between y_j and $\phi(\alpha)$ for $\boldsymbol{\alpha} \in \mathbb{F}_q^s$, where $\phi((\alpha_1, \ldots, \alpha_s)) = (\phi(\alpha_1), \ldots, \phi(\alpha_s)) \in (\mathbb{R}^2)^s$. Let the vector of such distances be denoted by $\mathbf{d}'_j$ for every $1 \leq j \leq N$ (note that $\mathbf{d}'_j$ can be computed from $\mathbf{d}_i$, where $i = \ell + (j-1) \mod (m-s+1) + m \left\lfloor \frac{j-1}{m-s+1} \right\rfloor$ for $1 \leq \ell \leq s$).

For every $1 \leq j \leq N$ and $\boldsymbol{\alpha} \in \mathbb{F}_q^s$ such that the distance between y_j and $\phi(\boldsymbol{\alpha})$ is not among the smallest r values, assign $w_{j,\boldsymbol{\alpha}} = 0$. Next, we specify how the remaining weights are chosen. For every $1 \leq j \leq N$, define $d^j_{\min}$ to be the smallest component in the vector $\mathbf{d}'_j$ and let $\mathbf{u}_j = \mathbf{d}'_j - (d^j_{\min}, \ldots, d^j_{\min})$. From $\mathbf{u}_j$, we define a probability vector $\mathbf{p}_j$ that represents the probability distribution whose support is over $\boldsymbol{\alpha} \in \mathbb{F}_q^s$ that have an entry in $\mathbf{u}_j$ and the probability value assigned is proportional to $\exp(-u_j(\boldsymbol{\alpha}))$. Finally, we compute $w_{j,\boldsymbol{\alpha}}$ (for $\boldsymbol{\alpha}$ that appears in $\mathbf{u}_j$) as $\left\lfloor \frac{A \cdot p_j(\boldsymbol{\alpha})}{p_{\min}} \right\rfloor$, where $p_{\min}$ is the smallest component value over all $\mathbf{p}_j$ $(1 \leq j \leq N)$.

Our soft decoding algorithm for $\mathrm{GE}_{\mathrm{BPSK}}$ is similar to the one for $\mathrm{GE}_{\mathrm{QAM}}$ above. In BPSK modulation, $\{0, 1\}$ will be mapped to $\{-a, a\}$ for some parameter $a > 0$.

Let the BPSK modulation be given by the function $h : \mathbb{F}_q \to \{-a, a\}^{\log_2(q)}$, for some parameter $a > 0$. Note that the received word will be a vector $\mathbf{y} = (y_1, \ldots, y_n) \in (\mathbb{R}^{\log_2(q)})^n$. As with our algorithm for $\mathrm{GE}_{\mathrm{QAM}}$, for every position $1 \leq i \leq n$, we compute $\mathbf{d}_i \in \mathbb{R}^r$ to be the vector of the r smallest such Euclidean distances between y_i and $h(\beta)$ for $\beta \in \mathbb{F}_q$ and then follow exactly the same procedure as for q-QAM to obtain the weights for the soft decoding algorithm in Theorem 1.

V. Simulation Results

A. Setup

We ran simulations on $\mathrm{RS}[k, 255]$ for $k \in \{145, 191\}$ over $\mathbb{F}_{256}$. We compared the performance of our soft decoding algorithm with RS (hard) list decoding and the soft decoding algorithm of Koetter and Vardy (KV)[3]. The KV soft decoding algorithm, we recall, involves a parameter that controls the number of iterations. We optimized this parameter empirically to give the best performance possible for the KV algorithm. We also simulated the performance of the soft decoding algorithm by Das and Vardy (DV) [18]. However, the performance of the DV soft decoder was inferior[4] to that of the KV decoder on $\mathrm{GE}_{\mathrm{QAM}}$ and $\mathrm{GE}_{\mathrm{BPSK}}$. We picked 1000 random messages from the RS code and ran the simulation for multiple random noise samples keeping track of decoding failures to compute the decoding error probability.

B. Choosing the parameters

In our simulations, $r = 4$ and $A = 10^4$ performed best and we present the results for those choices, for $\mathrm{GE}_{\mathrm{QAM}}$ and $\mathrm{GE}_{\mathrm{BPSK}}$, in which we fixed the SNR of the "bad" state to be 5dB and -5dB, respectively.

[2] To simplify the presentation, we will not explicitly keep track of these r values in $\mathbb{F}_q$, though in our simulations we do.

[3] Specifically, we computed $\mathbf{p}_j$ for $1 \leq j \leq n$ as before but with $s = m = 1$. Then, the KV algorithm was used to compute the weights.

[4] This was true even with the different demodulation used in [18], which was personally communicated to us by Das.

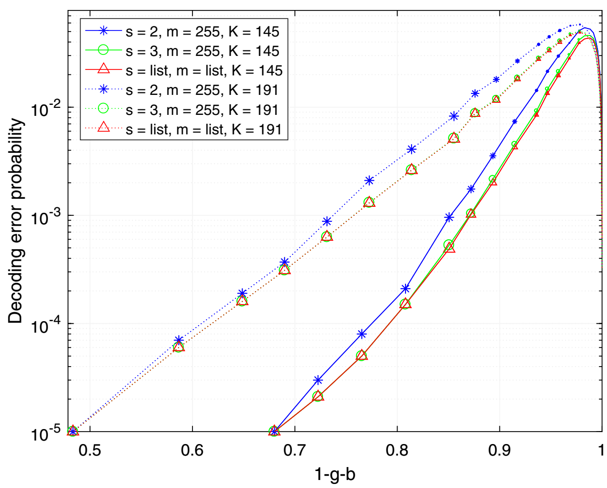

Fig. 1: Choosing s and m for $\mathrm{GE}_{\mathrm{QAM}}$ simulation for $k = 145$, $g/b = 15$, 28dB SNR in "good" state; for $k = 191$, $g/b = 30$, 30dB SNR in "good" state.

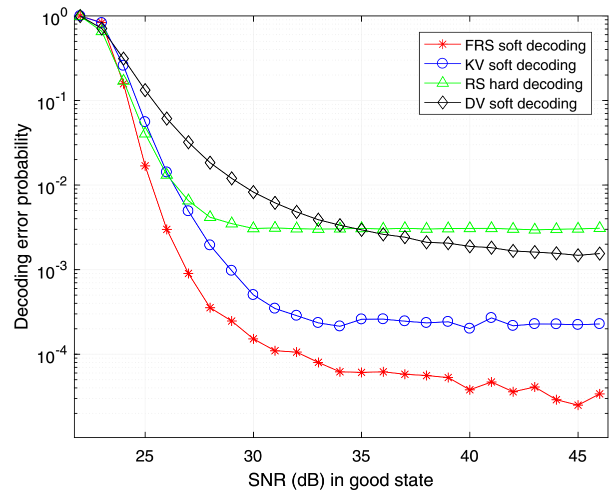

Fig. 2: $\mathrm{RS}[145, 255]$ on $\mathrm{GE}_{\mathrm{QAM}}$ with $g = 0.15$ and $g/b = 15$.

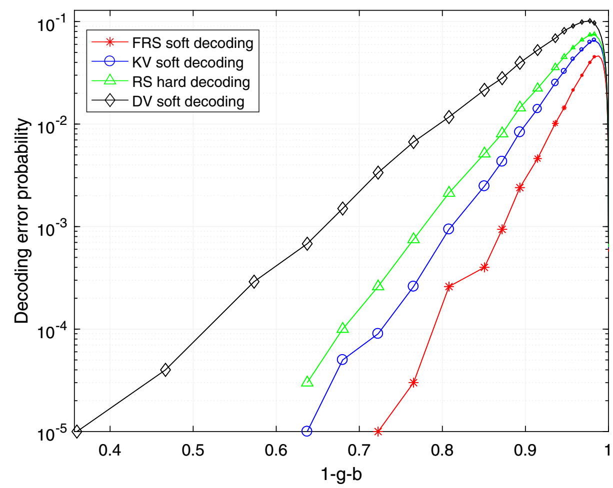

Fig. 3: $\mathrm{RS}[145, 255]$ on $\mathrm{GE}_{\mathrm{QAM}}$ with $g/b = 15$ and 28dB SNR in "good" state.

Intuitively, since the burst errors in the GE channel are not phased, it would make sense to pick large m – in particular $m = 255$, which seems like a reasonable choice. Furthermore, the proof of Theorem 1 crucially uses the fact that consecutive regions of uncorrupted symbols give rise to a lot of "sliding windows" of size s that are uncorrupted. A larger value of s tends to give better bounds. However, larger s works only if the non-error locations mostly occur in consecutive windows of size (much) larger than s. Again, on the GE channel (especially, for larger values of b) this would be unlikely. Thus, $s = 2$ or 3 seems to be a reasonable choice. We picked $s = 3$ and $m = 255$ in our simulations based on preliminary experiments as illustrated in Fig. 1 for $\mathrm{GE}_{\mathrm{QAM}}$. For $\mathrm{GE}_{\mathrm{BPSK}}$, $s = 2$ and $m = 255$ or $s = 3$ and $m = 255$ are good choices - the plots are omitted due to lack of space.

C. Simulation Results for $\mathrm{GE}_{\mathrm{QAM}}$

In our simulation results, our suggested FRS soft decoding algorithm outperforms both the KV and DV soft decoding procedures as well as RS hard list decoding.

Fig. 2 plots the decoding error probability of the four algorithms versus the SNR in the "good" state for $\mathrm{RS}[145, 255]$. At error rate of 10^{-3}, we have a gain of about 2dB over the KV soft decoding algorithm, while only our suggested soft decoding algorithm obtains a decoding error probability lower than 10^{-4} (at 32dB SNR or higher in the "good" state).

Fig. 3 plots the decoding error probability of the four algorithms versus $\mu \triangleq 1 - g - b$. Again, the superiority of the suggested soft decoding algorithm is apparent. At $\mu \approx 0.9$ our decoding error rate is about 7 times lower than other decoding algorithms.

The performance of the algorithms for $\mathrm{RS}[191, 255]$ is very similar to $\mathrm{RS}[145, 255]$ - the plots are omitted due to lack of space.

D. Simulation Results for $\mathrm{GE}_{\mathrm{BPSK}}$

In our simulation studies for $\mathrm{GE}_{\mathrm{BPSK}}$, the results are not uniformly good for our soft decoding algorithm. The suggested FRS soft decoding algorithm outperforms the KV algorithm and RS hard list decoding (as well as the DV algorithm) for certain ranges of parameters only, in particular for smaller values of g and b that describe "bursty" channels (and for $k = 191$ with larger value of a). This result is also somewhat intuitive as under $\mathrm{GE}_{\mathrm{BPSK}}$ the noise acts on bits and for larger values of b the probability of a *symbol* in $\mathbb{F}_{256}$ being in error increases. (Note that this is less likely in $\mathrm{GE}_{\mathrm{QAM}}$, where the noise acts on symbols in $\mathbb{F}_{256}$.)

Fig. 4 plots the decoding error probability of the four algorithms versus the SNR in the "good" state for $\mathrm{RS}[145, 255]$. After 4dB SNR in the "good" state, our algorithm outperforms

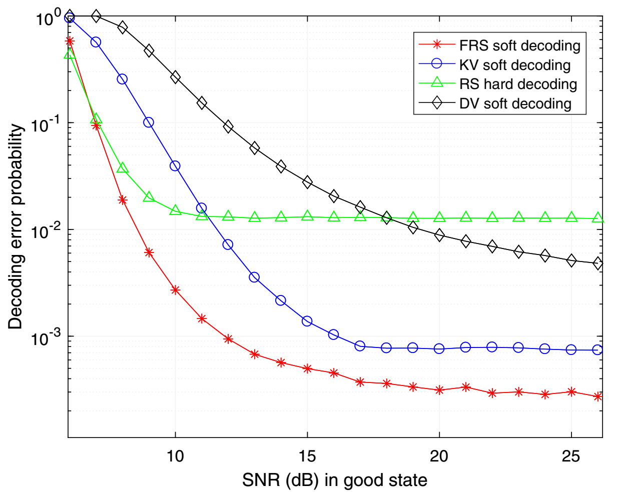

Fig. 4: $\mathrm{RS}[145, 255]$ on $\mathrm{GE}_{\mathrm{BPSK}}$ with $g = 0.02$, $g/b = 12$ and $a = 1$.

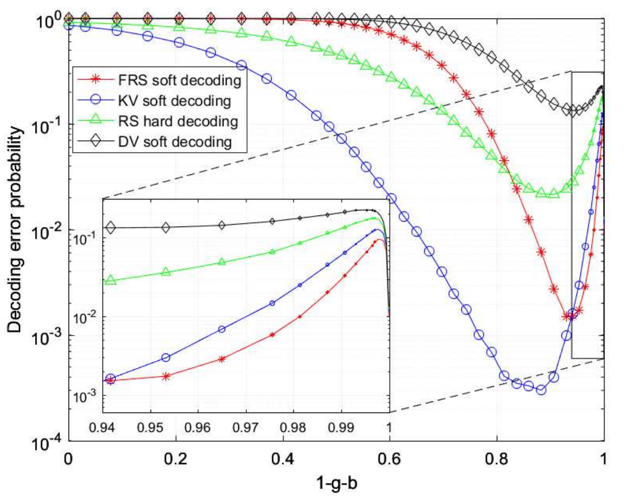

Fig. 5: $\mathrm{RS}[145, 255]$ on $\mathrm{GE}_{\mathrm{BPSK}}$ with $g/b = 6$, 14dB SNR in "good" state and $a = 1$.

the other three algorithms.

In Fig. 5, we plot the decoding error probability of the four algorithms versus μ. For μ above 0.95, which is of primary interest as they closely model burst errors, the proposed algorithm performs superior compared to the other three algorithms. For smaller μ values, the KV soft decoding algorithm performs the best.

The performance of the algorithms for $\mathrm{RS}[191, 255]$ is very similar to $\mathrm{RS}[145, 255]$ - the plots are omitted due to lack of space.

VI. Conclusion

In this paper, we extended the Reed-Solomon codes list decoding algorithm developed by Guruswami and Rudra originally for *phased* burst errors to more "irregular" burst errors. Particularly, we develop a low-complexity multiplicity assignment scheme for soft decoding of Reed-Solomon (RS) codes and in our simulation results outperforms the existing soft decision decoding algorithms of Koetter and Vardy as well as Das and Vardy on Gilbert-Elliott (GE) more bursty channels. We also presented a theoretical result in which we showed that for certain GE channels, with high probability of errors, the output list size for list decoding RS codes is *one*.

Acknowledgement

We would like to thank Dr. Atri Rudra, a top theorists in the field, for all the intriguing research discussions and valuable feedback, which has significantly improved the presentation of this research article.

References

[1] M. Sudan, "Decoding of Reed Solomon codes beyond the error-correction bound," *J. Complexity*, vol. 13, no. 1, pp. 180–193, 1997.
[2] V. Guruswami, "Linear-algebraic list decoding of folded reed-solomon codes," in *Proc. of IEEE 26th Annual Conference on Computational Complexity (CCC) 2011*, San Jose, CA, USA, June 2011, pp. 77–85.
[3] V. Guruswami and M. Sudan, "Improved decoding of Reed-Solomon and algebraic-geometry codes," *IEEE Trans. Inform. Theory*, vol. 45, pp. 1757–1767, Sept. 1999.
[4] R. Koetter and A. Vardy, "Algebraic soft-decision decoding of Reed-Solomon codes," *IEEE Trans. Inform. Theory*, vol. 49, pp. 2809 – 2825, Nov 2003.
[5] F. Parvaresh and A. Vardy, "Correcting errors beyond the Guruswami-Sudan radius in polynomial time," in *Proc. of 46th Annual IEEE Sympos. on Foundations of Comp. Sc.*, Pittsburgh, USA, Oct. 2005, pp. 285–294.
[6] V. Guruswami and A. Rudra, "Explicit codes achieving list decoding capacity: Error-correction with optimal redundancy," *IEEE Trans. Inform. Theory*, vol. 54, pp. 135–150, Jan. 2008.
[7] T. Kolan and R. M. Roth, "Burst list decoding of interleaved reed-solomon codes," *IEEE Trans. Inform. Theory*, vol. 60, no. 1, pp. 182–190, 2014.
[8] E. N. Gilbert, "Capacity of a burst-noise channel," *Bell System Technical Journal*, vol. 39, pp. 1253–1265, 1960.
[9] E. O. Elliot, "Estimates of error rates for codes on burst-noise channels," *Bell System Technical Journal*, vol. 42, pp. 1977–1997, 1963.
[10] C. Pimentel and I. Blake, "Non-interleaved Reed-Solomon coding performance on finite state channels," in *Proc. IEEE Internat. Conf. Comm.*, vol. 3, Montreal, Canada, June 1997, pp. 1493–1497.
[11] W. G. H. Zepernick and M. Caldera, "APP decoding of non-binary block codes on Gilbert-Elliott channels using generalized weight polynomials," in *Proc. IEEE Internat. Symp. Inform. Theory and its Applications*, Auckland, New Zealand, Dec. 2008, pp. 1–6.
[12] J. Chen and P. Owsley, "Burst-error-correcting algorithm for Reed-Solomon codes," *IEEE Trans. Inform. Theory*, vol. 38, pp. 1807–1812, Nov. 1992.
[13] V. Y. Krachkovsky, "Reed-Solomon codes for correcting phased error bursts," *IEEE Trans. Inform. Theory*, vol. 49, pp. 2975–2984, Nov. 2003.
[14] M. H. Mohamed, S. Puchinger, and M. Bossert, "Guruswami-sudan list decoding for complex reed-solomon codes," in *Proc. of 11th IEEE International ITG Conference on Systems, Communications and Coding, (SCC) 2017*, Hamburg, Germany, Feb. 2017, pp. 1–6.
[15] S. Egorov, "A list decoding algorithm for practical reed-solomon codes," in *Proc. of IEEE East-West Design Test Symposium (EWDTS 2013)*, Rostov-on-Don, Russia, Sept. 2013, pp. 1–4.
[16] A. Rudra and S. Uurtamo, *Two Theorems on List Decoding*. Springer, Berlin, Heidelberg, 2010, in: Serna M., Shaltiel R., Jansen K., Rolim J. (eds) Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques. RANDOM 2010, APPROX 2010. Lecture Notes in Computer Science.
[17] C. A. León and F. Perron, "Optimal Hoeffding bounds for discrete reversible markov chains," *Annals of Applied Prob.*, vol. 14, no. 2, pp. 958–970, 2004.
[18] H. Das and A. Vardy, "Multiplicity assignments for algebraic soft-decoding of reed-solomon codes using the method of types," in *Proc. IEEE Internat. Symp. Inform. Theory (ISIT)*, Seoul, Korea, 2009, pp. 1248–1252.