Fall 2007

Classification of the Deletion Correcting Capabilities of Reed–Solomon Codes of Dimension Over Prime Fields

L. McAven
University of Wollongong, lukemc@uow.edu.au

R. Safavi-Naini
University of Wollongong, rei@uow.edu.au

Classification of the Deletion Correcting Capabilities of Reed–Solomon Codes of Dimension Over Prime Fields

Abstract

Deletion correction codes have been used for transmission synchronization and, more recently, tracing pirated media. A generalized Reed-Solomon (GRS) code, denoted by GRSk(l,q,alpha,v), is a code of length l over GF(q) with qk codewords. These codes have an efficient decoding algorithm and have been widely used for error correction and detection. It was recently demonstrated that GRS codes are also capable of correcting deletions. We consider a subclass of GRS codes with dimension k=2 and q prime, and study them with respect to deletion correcting capability. We give transformations that either preserve the code or maintain its deletion correction capability. We use this to define equivalent codes; and then use exhaustive and selective computer searches to find inequivalent codes with the highest deletion correcting capabilities. We show that, for the class under consideration, up to l-3 deletions may be corrected. We also show that for lles36 there exist codes with q2 codewords such that receiving only 3 out of t transmitted symbols of a codeword is enough to recover the codeword, thus meeting the bound specified above. We also specify some "nice" codes which are associated with the smallest field possible for codes of a given length and deletion correcting capability. Some of the identified codes are unique, with respect to the defined equivalence.

Keywords

Codes, deletion correction, Reed–Solomon.

Disciplines

Physical Sciences and Mathematics

Publication Details

This article was originally published as McAven, L & Safavi-Naini, R, Classification of the Deletion Correcting Capabilities of Reed–Solomon Codes of Dimension Over Prime Fields, IEEE Transactions on Information Theory, 53(6), 2007, 2280-2294. Copyright Institute of Electronics and Electrical Engineers 2007. Original article available here

As a consequence of Lemma 2 and Lemma 3 we have the following.

Corollary 1: G has a vertex cover of size t if and only if G'' has a stopping set of size $t(m+1)+m$, $1 \leq t \leq n-1$. Hence $(G,t) \in$ VERTEX COVER(=) if and only if $(G'', t(m+1)+m) \in$ STOPPING DISTANCE.

Corollary 2: G has a vertex cover of size at most t if and only if G'' has a stopping set of size at most $t(m+1)+m$, $t \in \{1, 2, .., n-1\}$. Hence $(G,t) \in$ VERTEX COVER if and only if $(G'', t(m+1)+m) \in$ STOPPING DISTANCE.

We are now ready to prove.

Theorem 1: STOPPING DISTANCE and STOPPING SET are NP-complete

Proof: Since G'' can be constructed from G in polynomial time ($O(mn)$ time suffices), it follows that VERTEX COVER(=) $\preceq_p$ STOPPING SET and VERTEX COVER $\preceq_p$ STOPPING DISTANCE from Corollary 1 and Corollary 2 respectively. It is easy to verify whether a given set of left vertices of a bipartite graph forms a stopping set in time linear in the size of the graph. Hence both STOPPING DISTANCE and STOPPING SET belong to the class NP. □

As a consequence, we have the following corollary.

Corollary 3: There is no polynomial time algorithm for computing the stopping distance of a Tanner graph unless P = NP.

Acknowledgment

The authors would like to thank Dr. L. Sunil Chandran for useful discussions, and the anonymous referees for their helpful comments. K. Murali Krishnan acknowledges sponsorship for the Ph.D. degree from the National Institute of Technology, Calicut under the QIP scheme.

References

[1] C. Di, D. Proietti, I. E. Telatar, T. J. Richardson, and R. L. Urbanke, "Finite length analysis of low-density parity-check codes on the binary erasure channel," *IEEE Trans. Inf. Theory.*, vol. 48, pp. 1570–1579, Jun. 2002.

[2] C. Di, A. Montanari, and R. Urbanke, "Weight distribution of LDPC code ensembles: Combinatorics meets statistical physics," in *Proc. IEEE Int. Symp. Inf. Theory*, Chicago, IL, Jul. 2004, p. 102.

[3] A. Orlitsky, K. Viswanathan, and J. Zhang, "Stopping set distribution of LDPC code ensembles," *IEEE Trans. Inf. Theory*, vol. 51, pp. 929–953, Mar. 2005.

[4] T. Tian, C. Jones, J. D. Villasenor, and R. D. Wesel, "Construction of irregular LDPC codes with low error floors," in *Proc. IEEE Int. Conf. Commun.*, Seattle, WA, May 2003, pp. 3125–3129.

[5] A. Ramamoorthy and R. Wesel, "Construction of short block length irregular LDPC codes," in *Proc. IEEE Int. Conf. Commun.*, Paris, France, Jun. 2004, pp. 410–414.

[6] A. Orlitsky, R. Urbanke, K. Viswanathan, and J. Zhang, "Stopping sets and girth of Tanner graphs," in *Proc. IEEE Int. Symp. Inf. Theory*, Lausanne, Jun. 2002, p. 2.

[7] M. Schwartz and A. Vardy, "On the stopping distance and the stopping redundancy of codes," *IEEE Trans. Inf. Theory*, vol. 52, pp. 922–932, Mar. 2006.

[8] R. M. Tanner, "A recursive approach to low-complexity codes," *IEEE Trans. Inf. Theory*, vol. 27, pp. 533–547, Sep. 1981.

[9] T. H. Cormen, C. E. Leicerson, and R. L. Rivest, *Introduction to Algorithms*. Cambridge, MA: MIT Press, 1990.

[10] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. New York: W. H. Freeman, 1979.

[11] S. Cook, "The complexity of theorem proving procedures," in *Proc. Third ACM Ann. Symp. Theory Comput.*, Shaker Heights, OH, May 1971, pp. 151–158.

[12] H. Pishro-Nik and F. Fekri, "On decoding of low-density parity-check codes over the binary erasure channel," *IEEE Trans. Inf. Theory*, vol. 50, pp. 439–454, Mar. 2004.

[13] J. Han and P. Siegel, Improved Upper Bounds on Stopping Redundancy [Online]. Available: http://www.arXiv.org, cs.IT/0511056, to be published

[14] A. Vardy, "The intractability of computing the minimum distance of a code," *IEEE Trans. Inf. Theory*, vol. 46, pp. 1757–1766, Nov. 1997.

[15] E. R. Berlekamp, R. J. McEliece, and H. C. A. van Tilborg, "On the inherent intractability of certain coding problems," *IEEE Trans. Inf. Theory*, vol. IT-24, pp. 384–386, May 1978.

Classification of the Deletion Correcting Capabilities of Reed–Solomon Codes of Dimension 2 Over Prime Fields

Luke McAven and Reihaneh Safavi-Naini*, Member, IEEE*

***Abstract*—Deletion correction codes have been used for transmission synchronization and, more recently, tracing pirated media. A Generalized Reed–Solomon (GRS) code, denoted by $GRS_k(\ell, q, \boldsymbol{\alpha}, \mathbf{v})$, is a code of length ℓ over $GF(q)$ with q^k codewords. These codes have an efficient decoding algorithm and have been widely used for error correction and detection. It was recently demonstrated that GRS codes are also capable of correcting deletions. We consider a subclass of GRS codes with dimension $k=2$ and q prime, and study them with respect to deletion correcting capability. We give transformations that either preserve the code or maintain its deletion correction capability. We use this to define equivalent codes; and then use exhaustive and selective computer searches to find inequivalent codes with the highest deletion correcting capabilities. We show that, for the class under consideration, up to $\ell - 3$ deletions may be corrected. We also show that for $\ell \leq 36$ there exist codes with q^2 codewords such that receiving only 3 out of ℓ transmitted symbols of a codeword is enough to recover the codeword, thus meeting the bound specified above. We also specify some "nice" codes which are associated with the smallest field possible for codes of a given length and deletion correcting capability. Some of the identified codes are unique, with respect to the defined equivalence.**

***Index Terms*—Codes, deletion correction, Reed–Solomon.**

I. Introduction

Error-correcting codes are widely used to correct substitution and erasure errors. A different, less studied, class of codes are the deletion correcting (DC) codes, introduced by Levenshtein [6] to correct synchronisation errors. The applications of DC codes include packet loss in Internet transmission [13] and, more recently, tracing pirate media [11]. Various studies of DC codes have been made [1], [2], [5]–[8], [12]–[14], [17]. These studies generally consider a small number of deletions, or a specific class of combinatorial based codes, or bounds of various sorts. Perfect deletion correcting codes are codes for which every possible word of some length over the associated alphabet is a subword of exactly one codeword. It is known that perfect codes exist. For example, there are many length 6 codes (over different alphabets) capable of correcting four deletions [12]. In that case any word of length 2 is a subword of exactly one codeword.

Manuscript received June 2, 2004; revised February 19, 2007.

L. McAven is with the Centre for Computer and Information Security Research, School of Computer Science and Software Engineering University of Wollongong, Australia (e-mail: lukemc@uow.edu.au).

R. Safavi-Naini is with the iCore Information Security Lab, Department of Computer Science, University of Calgary, Canada (e-mail: rei@epsc.ucalgary.ca).

Communicated by Ø. Ytrehus, Associate Editor for Coding Techniques.

Digital Object Identifier 10.1109/TIT.2007.896889

Sloane [13] surveys single deletion correcting codes. He primarily focuses on binary codes and discusses difficulties of constructing and analysing deletion correcting codes. Sloane reports on exhaustive searches to find the largest single deletion correcting binary codes of a given length, which showed that the Varshamov–Tenengolts codes [15] are of optimal size for length up to 9. Lenveshtein [6] had shown that these codes are capable of correcting one deletion.

It was recently observed [11] that Generalized Reed–Solomon (GRS) codes, extensively studied for their error correction capability, are also capable of deletion correction. A subsequent study [16] detailed a method for obtaining length 5 codes capable of correcting one deletion. That work also gave the results of numerical experiments to investigate the deletion correction capabilities of GRS codes.

An important advantage of using GRS codes, initially noted in [11], is the existence of an efficient deletion correcting algorithm. The decoding algorithm for GRS codes can be formulated as a polynomial reconstruction problem, to which the efficient list decoding algorithm of Guruswami and Sudan [3] applies.

A $\mathrm{GRS}_k(\ell, q, \boldsymbol{\alpha}, \mathbf{v})$ code is specified by the parameters $\boldsymbol{\alpha}$, $\mathbf{v}$, k, ℓ and q. We call $\boldsymbol{\alpha}$ and $\mathbf{v}$ the *selector* and *multiplier*, respectively. GRS codes for which $\mathbf{v} = \mathbf{1}$ is a vector of all ones, are the widely studied Reed–Solomon (RS) codes [10]. We focus on RS codes that are defined over a prime field GF(q) that have dimension $k = 2$ and so $q^k = q^2$ codewords. The choice of unit multiplier codes is because our previous computer searches [16], both exhaustive and selective, suggest that these codes appear likely to correct more deletions then nonunit multiplier codes. We also believe RS codes merit special attention because of their importance and wide application. Restricting ourselves to codes with dimension $k = 2$ allows us to exploit the linear relationship between codewords to analyze them.

We first prove an upper bound on the deletion correcting capability of the codes in this class. A follow-on question is; "When can the bound be achieved with equality?", which translates into; "What is the smallest field for which there exists a code that achieves the bound?" We use structured computer searches to obtain insight into the above questions and provide interesting results. We use properties of code classes to aid the search, hence being able to obtain insight into the above questions and produce results for larger (and so more interesting) fields. We also find codes that are the 'best', in ways that we will define.

In our computer searches we restrict our attention to finite fields of prime order to simplify our search and classification. Classification of codes over non-prime fields introduces other factors, such as the choice of primitive polynomial, that complicate the task of obtaining experimental results (some experimental results for fields of prime power size are given in [16]). The theoretical results in this correspondence, or slightly modified versions of them, are applicable to nonprime fields. In particular, we show the bound on deletion capacity is for dimension $k = 2$ RS codes, over prime and nonprime fields.

For RS codes, we show that distinct $\boldsymbol{\alpha}$ can result in the same code (the same set of codewords). Indeed, an affine transformation applied to the vector $\boldsymbol{\alpha}$ results in $\boldsymbol{\alpha}'$, which generates the same code in this sense. This allows us to associate each unit multiplier code with a code having a selector of the form $(0\ 1\ \alpha_3\ \ldots\ \alpha_\ell)$, and hence reduce exhaustive searches of $k = 2$ unit multiplier codes to a small proportion $(1/(q^2 - q))$ of all codes in the class.

We define equivalent codes as codes that are obtained through applying a deletion correcting distance preserving transformation, preserving at the level of codeword to codeword with a map across the entire codebook.

We define equivalent codes as codes related through the application of a distance preserving (deletion correcting distance) transformation. We refer to such a transformation as a isomorphism in Section II-B, and note that the codes have the same deletion correcting capability. We seek transformations on selectors that result in equivalent codes. Unlike error correcting codes, for which column permutations leave the codes invariant from a distance distribution view point, a general column permutation changes the deletion correcting capability of codes. We show, however, that there is a nontrivial permutation for which the deletion correcting capability will remain the same.

We enumerate inequivalent RS codes (over prime field) parameterized by q, ℓ and r, where r is the number of deletions that a code can correct. We prove that $r = \ell - 3$ is an upper bound on the deletion correcting capability, that is $r \leq \ell - 3$. We have identified examples of codes with $r = \ell - 3$ for $\ell \leq 36$. For $\ell \leq 8$ we have identified, and proven by exhaustive search, the smallest q for which such codes exist. When the number of such inequivalent codes is known and is small, we have listed the complete set in Appendix. In some cases there are very few codes with these parameters. For example, there is only one code for $q = 23$, $\ell = 6$ and $r = 3$, and there is no code with $\ell = 6$, $r = 3$ for any $q < 23$. We give explicit examples of such codes with complete codebooks.

The rest of the correspondence is structured as follows. In Section II we introduce the basics of deletion correcting codes, GRS codes and RS codes. We define the notion of equivalence of codes in this context.

In Section III, we define and discuss the affine and reversal transformations. In Section IV, we give bounds on the deletion correcting capability and enumerate the distinct codes with the same deletion correcting capabilities. Finally, Section V contains a summary of, and discussion on, our results. An Appendix contains lists of small sets of inequivalent codes for particular prime RS codes parameterized by (q, ℓ, r).

II. Preliminaries

A. Deletion Correcting

Let a and b be strings over a q—ary alphabet A. We denote the length of a, that is the number of elements (letters) in it, by $|a|$. We say a string a is a *subword* of b if a can be obtained from b by only removing elements of b. For example, 2234 is a subword of 142254364, while 452 is not (since reordering is required).

A *q—ary code* is a collection of q—ary words. A linear code of dimension k is a subspace of dimension k of GF(q).

A code can *correct* r deletions if any string of length $\ell - r$ is a subword of at most one codeword. We say such a code has a deletion correcting capacity of r. For a particular code Γ we use the notation $r(\Gamma)$ to denote the deletion correcting capability.

To find the deletion correcting capability of a code we need to find the length of the longest common subwords of any pair of codewords, across all pairs of codewords in the code. Let u and v be two codewords of a code Γ and let $\rho(u, v)$ denote *a* longest common subword of u and v, of length $|\rho(u, v)|$. There may be many common subwords with this same length. We define $\mathcal{R}(\Gamma) = \max_{u,v\in\Gamma,u\neq v} |\rho(u, v)|$ and let $s = \mathcal{R}(\Gamma) + 1$. Then s is the unique subword length, that is; the length of the shortest subword that uniquely identifies a codeword. It follows that Γ is an r—deletion correcting code, where $r = (\ell - s)$.

B. Reed–Solomon (GRS) Codes

Let F_q, q prime, be a field of q elements. Let k be an integer and

$$F_q[x]_k = \{f(x) : f(x) \text{is a polynomial over } F_q : \deg(f) < k\}.$$

Let $\alpha_1, \alpha_2, \ldots, \alpha_\ell \in F_q$, $\ell \leq q$ be distinct. Let $v_1, v_2, \ldots, v_\ell \in F_q$ be non-zero. Write $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_\ell)$ and $\mathbf{v} = (v_1, v_2, \ldots, v_\ell)$. A k-dimensional prime Generalized Reed–Solomon (GRS) code of length ℓ is the set of all vectors

$$(v_1 f(\alpha_1), v_2 f(\alpha_2), \ldots, v_\ell f(\alpha_\ell)), \quad \forall\, f \in F_q[x]_k.$$

This code is denoted by $\text{GRS}_k(\ell, q, \boldsymbol{\alpha}, \mathbf{v})$. We refer to $\boldsymbol{\alpha}$ as the *selector* and $\mathbf{v}$ as the *multiplier*. We note that $\text{GRS}_k(\ell, q, \boldsymbol{\alpha}, \mathbf{v})$ has q^k codewords. We say a particular vector, or codeword, in the code/set is *associated* with the polynomial $f(x)$ used to generate it.

We identify two particular sets of GRS codes (over prime field).

Definition 1 (GRS Classes): For fixed ℓ, q, k, let $G(\ell, q, k)$ be the collection of codes obtained by taking all $\text{GRS}_k(\ell, q, \boldsymbol{\alpha}, \mathbf{v})$ codes with all possible $\boldsymbol{\alpha}$ and $\mathbf{v}$.

For a particular code in $G(\ell, q, k)$, say $\text{GRS}_k(\ell, q, \boldsymbol{\alpha}, \mathbf{v})$, we define $s(\ell, q, k, \boldsymbol{\alpha}, \mathbf{v}) = \mathcal{R} + 1$, where $\mathcal{R}$ was defined in the previous section. We then define

$$\sigma(\ell, q, k) = \min_{\boldsymbol{\alpha}} \min_{\mathbf{v}} s(\ell, q, k, \boldsymbol{\alpha}, \mathbf{v}) \tag{1}$$

to be the minimum $s(\ell, q, k, \boldsymbol{\alpha}, \mathbf{v})$ for $G(\ell, q, k)$. This gives the highest deletion correcting capability for given parameters ℓ, q, k.

We are interested in a subclass where the multiplier values are all one. Such codes are Reed–Solomon codes over prime fields.

Definition 2 (RS Classes): For fixed ℓ, q, k, the subset of codes in $\text{GRS}_k(\ell, q, \boldsymbol{\alpha}, \mathbf{v})$ for which $v_i = 1, 1 \leq i \leq \ell - 1$ are called RS codes and are denoted by $\tilde{G}(\ell, q, k)$.

We use $\tilde{G}(\ell, q, 2)$ to denote the subset of $\tilde{G}(\ell, q, k)$ class with $k = 2$.

We note that, as specified, not all elements of $G(\ell, q, k)$ are distinct. That is two distinct $\boldsymbol{\alpha}$ may result in the same set of codewords. Also two codes Γ and Γ' may have *equivalent* deletion correcting properties. Two codes are *isomorphic* if there is a one-to—one mapping T between Γ and Γ' such that for any pair of codewords $u, v \in \Gamma$ with $|\rho(u, v)| = t$, we have $|\rho(T(u), T(v))| = t$, and T is referred to as an *isomorphism*. Two codes are called *equivalent* if there is an isomorphism between them. We are interested in the enumeration of inequivalent codes and give, in Section IV, lists and counts of codes identified as inequivalent under the classes of isomorphism defined in Section III.

C. RS Codes and Deletion Correcting

The statements in this section regarding the subwords hold for non-prime RS codes also.

Let us use an example of a $\tilde{G}(\ell, q, 2)$ code, for $q = 7$ and $\ell = 4$, to illustrate RS codes and deletion correcting capability. We choose the selector $\boldsymbol{\alpha} = (1\ 3\ 0\ 4)$. The codebook for this code is shown in the expression at the bottom of the page. Each codeword is of length $\ell = 4$ and is associated with a polynomial of the form $f(x) = a_1 x + a_0$. The ith letter in a codeword is $f(\alpha_i)$. This code is capable of correcting up to one deletion. This means there are no length three words which are subwords of two distinct codewords, but that there are words of length two which are subwords of two distinct codewords.

For any code with unit multipliers, the first q codewords, those associated with the polynomials of degree 0, are constant codewords. Thus, for any given length, there is a single subword only for each of those codewords. For example, any subword of the codeword associated with the polynomial $0x + 2$ will always be a string of 2's, of whatever length the subword is.

Each codeword is associated with a polynomial of degree 1 and so has distinct components, that is each letter in the codeword differs from each other letter. Thus two subwords, of any length, taken from different columns of a codeword are always distinct.[1] Furthermore, the length of the longest common subword between, a codeword associated with a polynomial of degree 0 and any codeword associated with a polynomial of degree 1, is at most 1. Using this property effectively means that; to find the deletion correcting capability of a code we can restrict our attention to finding the length of subwords common to two codewords, both of which are associated with polynomials of degree 1.[2]

III. Equivalent Codes in $\tilde{G}(\ell, q, 2)$

In this section we present two transformations of selector vectors that result in equivalent codes. The first transformation results in a selector vector which generates the same code; that is, a code with the same set of codewords. The second transformation results in a different code (different set of vectors) that has the same deletion correcting capabilities through the isomorphism.

For a scalar s and a vector $\mathbf{v}$ with the ith component v_i, $i = 1, \ldots, n$, we adopt the convention that $\mathbf{v} + s$, denotes a vector $\mathbf{r}$ with components $r_i = v_i + s$, $i = 1, \ldots, n$.

A. Affine Transformations of the Selector

Theorem 1: If two codes Γ and Γ' in $\tilde{G}(\ell, q, 2)$ have respective selectors $\boldsymbol{\alpha} = (\alpha_1 \ldots \alpha_\ell)$ and $\boldsymbol{\alpha}' = (\alpha'_1 \ldots \alpha'_\ell)$ and there is an affine transformation T such that $\alpha'_i = T(\alpha_i)$, $i = 1, \ldots \ell$, then the two

[1]This allows us to identify both the deleted elements and the locations from which they have been deleted.

[2]Note that in the above case the longest common subword may be longer if the multipliers differ from 1, supporting the observation that unit multiplier codes (i.e., RS codes) would have higher deletion correcting capabilities. This property probably becomes less significant for larger qs, since the proportion of codewords associated with degree 0 polynomials becomes insignificant.

$0x+0$	0000	$0x+1$	1111	$0x+2$	2222	$0x+3$	3333
$0x+4$	4444	$0x+5$	5555	$0x+6$	6666	$1x+0$	1304
$1x+1$	2415	$1x+2$	3526	$1x+3$	4630	$1x+4$	5041
$1x+5$	6152	$1x+6$	0263	$2x+0$	2601	$2x+1$	3012
$2x+2$	4123	$2x+3$	5234	$2x+4$	6345	$2x+5$	0456
$2x+6$	1560	$3x+0$	3205	$3x+1$	4316	$3x+2$	5420
$3x+3$	6531	$3x+4$	0642	$3x+5$	1053	$3x+6$	2164
$4x+0$	4502	$4x+1$	5613	$4x+2$	6024	$4x+3$	0135
$4x+4$	1246	$4x+5$	2350	$4x+6$	3461	$5x+0$	5106
$5x+1$	6210	$5x+2$	0321	$5x+3$	1432	$5x+4$	2543
$5x+5$	3654	$5x+6$	4065	$6x+0$	6403	$6x+1$	0514
$6x+2$	1625	$6x+3$	2036	$6x+4$	3140	$6x+5$	4251
$6x+6$	5362						

.

codes have the same set of codewords and therefore the same deletion correcting capability.

Proof: Let Γ and Γ' have deletion correcting capabilities r and r', respectively. Let $T = aX + b$ be the affine transformation relating the selectors of Γ and Γ', that is $\boldsymbol{\alpha} = a\boldsymbol{\alpha}' + b$, $a \neq 0$, $a, b \in F_q$. For any $a_0, a_1 \in F_q$, that is any codeword $a_1\mathbf{x} + a_0$ in Γ, the codeword is $a_1(\boldsymbol{\alpha}) + a_0$. Applying the affine transformation we see the same codeword in Γ' is of the form $(a_0 + ba_1) + a_1 a(\boldsymbol{\alpha}')$. This is a codeword in Γ' since the selector is $\boldsymbol{\alpha}'$, the polynomial degree is at most 1 and both $(a_0 + ba_1)$ and $a_1 a$ are in F_q. Thus any codeword in Γ is also a codeword in Γ' (although there are different polynomials associated with the codeword in the different codes). Since the number of codewords in Γ and Γ' is equal they contain the same codewords, and thus have equal deletion correcting capabilities. $\square$

Let us consider an example of this correspondence through an affine transformation. In the previous section we considered a code with selector $\boldsymbol{\alpha} = (1\ 3\ 0\ 4)$. Let us consider the code with selector related by the affine transformation $\boldsymbol{\alpha}' = 2\boldsymbol{\alpha} + 3 = (5\ 2\ 3\ 4)$. For each polynomial in Γ, we give the codeword and the polynomial in Γ' which has the same codeword.

Γ		Γ'	Γ		Γ'
$0x+0$	0000	$0x+0$	$0x+1$	1111	$0x+1$
$0x+2$	2222	$0x+2$	$0x+3$	3333	$0x+3$
$0x+4$	4444	$0x+4$	$0x+5$	5555	$0x+5$
$0x+6$	6666	$0x+6$	$1x+0$	1304	$4x+2$
$1x+1$	2415	$4x+3$	$1x+2$	3526	$4x+4$
$1x+3$	4630	$4x+5$	$1x+4$	5041	$4x+6$
$1x+5$	6152	$4x+0$	$1x+6$	0263	$4x+1$
$2x+0$	2601	$1x+4$	$2x+1$	3012	$1x+5$
$2x+2$	4123	$1x+6$	$2x+3$	5234	$1x+0$
$2x+4$	6345	$1x+1$	$2x+5$	0456	$1x+2$
$2x+6$	1560	$1x+3$	$3x+0$	3205	$5x+6$
$3x+1$	4316	$5x+0$	$3x+2$	5420	$5x+1$
$3x+3$	6531	$5x+2$	$3x+4$	0642	$5x+3$
$3x+5$	1053	$5x+4$	$3x+6$	2164	$5x+5$
$4x+0$	4502	$2x+1$	$4x+1$	5613	$2x+2$
$4x+2$	6024	$2x+3$	$4x+3$	0135	$2x+4$
$4x+4$	1246	$2x+5$	$4x+5$	2350	$2x+6$
$4x+6$	3461	$2x+0$	$5x+0$	5106	$6x+3$
$5x+1$	6210	$6x+4$	$5x+2$	0321	$6x+5$
$5x+3$	1432	$6x+6$	$5x+4$	2543	$6x+0$
$5x+5$	3654	$6x+1$	$5x+6$	4065	$6x+2$
$6x+0$	6403	$3x+5$	$6x+1$	0514	$3x+6$
$6x+2$	1625	$3x+0$	$6x+3$	2036	$3x+1$
$6x+4$	3140	$3x+2$	$6x+5$	4251	$3x+3$
$6x+6$	5362	$3x+4$			

We want to count inequivalent codes. We define a standard representation for codes and use that to distinguish inequivalent codes.

Corollary 1: A code Γ represented by $\alpha_1 > 0$ and/or $\alpha_2 > 1$ can also be represented by a unique selector vector with $\alpha'_1 = 0$, $\alpha'_2 = 1$. We call this the *standard representation* or *standard form* of the code.

Proof: Let the code Γ have a selector $\boldsymbol{\alpha}$. Consider the selector $\boldsymbol{\alpha}' = \frac{\boldsymbol{\alpha}-\alpha_1}{\alpha_2-\alpha_1}$, where $\alpha'_i = \frac{\alpha_i-\alpha_1}{\alpha_2-\alpha_1}$ for all $i = 1, \dots, \ell$ and α'_i and α_i denote the ith component of $\boldsymbol{\alpha}'$ and $\boldsymbol{\alpha}$, respectively. Since $\alpha_2 \neq \alpha_1$, by definition, an inverse $A = (\alpha_2 - \alpha_1)^{-1}$ exists we have $\alpha'_i = A\alpha_i - \alpha_1 A$ and since $\alpha_1 > 0$ the relationship between $\boldsymbol{\alpha}$ and $\boldsymbol{\alpha}'$ is an affine transformation. Using Theorem 1, we conclude that $\boldsymbol{\alpha}'$ generates the same code as $\boldsymbol{\alpha}$. Evaluating the first two elements of $\boldsymbol{\alpha}'$ we find $\alpha'_1 = 0$ and $\alpha'_2 = 1$.

The two parameters of the affine transformation are fixed by the need to fix the 0 and 1 in the first two components of the selector in standard form and so the transformation and the standard form representation of the vector are unique. $\square$

These results are especially useful for codes in $\tilde{G}(\ell, q, 2)$ classes. There we only need to consider codes in the standard representation, that is with $\alpha_1 = 0$ and $\alpha_2 = 1$. This reduces the search space by a factor of $1/(q^2 - q)$.

B. Selector Reversal

Here we show that the deletion correcting capabilities of $\tilde{G}(\ell, q, k)$ codes is invariant under reversal of the selector, for arbitrary k.[3]

For error correcting codes the error correcting capability is invariant under *any permutation* of the columns of the code. This is not the case for deletion correcting capability, as the following example illustrates. Consider two codewords $c_1 = (1\ 2\ 3\ 4\ 5)$ and $c_2 = (5\ 2\ 4\ 6\ 7)$. The cyclic permutation of columns $(1\ 5\ 4\ 3\ 2)$ gives the codewords $c'_1 = (2\ 3\ 4\ 5\ 1)$ and $c'_2 = (2\ 4\ 6\ 7\ 5)$. While c_1 and c_2 have a longest common subword $(2\ 4)$ of length 2, c'_1 and c'_2 have a longest common subword $(2\ 4\ 5)$ of length 3.

We define $\overline{x}$ for a word x where $\overline{x}_i = x_{|x|+1-i}$. That is, $\overline{x}$ is x written backwards.

Lemma 1: If a is a subword of b, than $\overline{a}$ is a subword of $\overline{b}$.

Proof: Let a_i, a_{i+1} be a subword of a. Then by definition $\overline{b}$ contains a_{i+1}, a_i. The result follows for any length subword a_i, $a_{i+1} \dots a_{i+u}$ by noting that the subword can be written as $(((a_i, a_{i+1}), a_{i+2}) \dots a_{i+u})$, and using recursion. $\square$

Lemma 2: For any two words c_1 and c_2, $\rho(\overline{c_1}, \overline{c_2}) = \overline{\rho(c_1, c_2)}$.

Proof: Since $\rho(c_1, c_2)$ is a subword of c_1 and c_2, Lemma 1 tells us that $\overline{\rho(c_1, c_2)}$ is a subword of $\overline{c_1}$ and $\overline{c_2}$. Assume $\rho(\overline{c_1}, \overline{c_2}) = p$ where $|p| > |\overline{\rho(c_1, c_2)}|$, i.e., that there exists a subword common to $\overline{c_1}$ and $\overline{c_2}$ which is longer than $\overline{\rho(c_1, c_2)}$. By Lemma 1, $\overline{p}$ is a subword of both c_1 and c_2, implying $|\rho(c_1, c_2)| \geq |\overline{p}| = |p|$. But since p was defined to satisfy $|p| > |\overline{\rho(c_1, c_2)}| = |\rho(c_1, c_2)|$, such a p cannot exist and therefore the longest subword common to $\overline{c_1}$ and $\overline{c_2}$ is $\overline{\rho(c_1, c_2)}$, as required. $\square$

Theorem 2: If a length ℓ code Γ with selector $\boldsymbol{\alpha}$ has a deletion correcting capability of r, then the code specified by the selector $\boldsymbol{\alpha}' : \alpha'_i = \alpha_{\ell+1-i}$, that is $\boldsymbol{\alpha}' = \overline{\boldsymbol{\alpha}}$, also has a deletion correcting capability of r.

Proof: Since $r(\Gamma) = \ell - \mathcal{R}(\Gamma) - 1$ and $r(\Gamma') = \ell - \mathcal{R}(\Gamma') - 1$, we may equivalently demonstrate that $\mathcal{R}(\Gamma') = \mathcal{R}(\Gamma)$

$$\begin{aligned}\mathcal{R}(\Gamma) &= \max_{c_1, c_2 \in \Gamma, c_1 \neq c_2} |\rho(c_1, c_2)| \\ &= \max_{c_1, c_2 \in \Gamma, c_1 \neq c_2} |\overline{\rho(c_1, c_2)}| \\ &= \max_{c_1, c_2 \in \Gamma, c_1 \neq c_2} |\rho(\overline{c_1}, \overline{c_2})| \text{ by Lemma 2.} \\ &= \max_{\overline{c_1}, \overline{c_2} \in \Gamma', \overline{c_1} \neq \overline{c_2}} |\rho(\overline{c_1}, \overline{c_2})| \\ &= \mathcal{R}(\Gamma').\end{aligned}$$ $\square$

The above theorem shows that the code with selector $\boldsymbol{\alpha}'$, generated by the reversal transformation, is isomorphic to $\boldsymbol{\alpha}$ and so they are equivalent. Consider, for example, the selectors of codes in $\tilde{G}(\ell, q, 2)$ with $q = 13$, $\ell = 5$, that have deletion correcting capability of 2; that is $r = 2$. There are only two such codes, $\boldsymbol{\alpha} = (0\ 1\ 7\ 6\ 2)$ and $\boldsymbol{\alpha}' = (0\ 1\ 11\ 3\ 6)$ in the standard representation. Now consider the reversal selector obtained from $\boldsymbol{\alpha}$; that is $\overline{\boldsymbol{\alpha}} = (2\ 6\ 7\ 1\ 0)$. We see that

[3]A similar isomorphism exists for general GRS codes if one reverses both the multiplier and the selector.

$\overline{\boldsymbol{\alpha}} = 4\boldsymbol{\alpha}' + 2$, that is $\overline{\boldsymbol{\alpha}}$ and $\boldsymbol{\alpha}'$ are related by an affine transformation. Thus the reversal transformation on one code gives the other code and so there is only one inequivalent code in the standard representation. Thus using either $\boldsymbol{\alpha}$ or $\boldsymbol{\alpha}'$ we can generate, using the affine and reversal transformations, any other selector corresponding to a $\tilde{G}(\ell, q, 2)$ code with $q = 13$, $\ell = 5$ and $r = 2$.

Corollary 2: Using the reversal transformation, a code in the standard representation is isomorphic to at most one other code in the standard representation.

Proof: Consider a code with selector $\boldsymbol{\alpha}$ in standard form $\alpha_1 = 0$ and $\alpha_2 = 1$. The reversal of $\boldsymbol{\alpha}$ gives the selector $\overline{\boldsymbol{\alpha}} = (\alpha_\ell\ \alpha_{\ell-1}\ \ldots\ \alpha_3\ 1\ 0)$. By Corollary 1, there is only one affine transformation which can be applied to $\overline{\boldsymbol{\alpha}}$ to obtain a selector $\boldsymbol{\alpha}'$ satisfying $\alpha'_1 = 0$ and $\alpha'_2 = 1$. The transformation (and thus selector) is specified by $\boldsymbol{\alpha}' = \frac{\overline{\boldsymbol{\alpha}} - (\overline{\alpha})_1}{(\overline{\alpha})_2 - (\overline{\alpha})_1} = \frac{\overline{\boldsymbol{\alpha}} - \alpha_\ell}{\alpha_{\ell-1} - \alpha_\ell}$, where $(\overline{\alpha})_i$ means the ith element of $\overline{\boldsymbol{\alpha}}$. It is possible for $\boldsymbol{\alpha}$ to equal $\boldsymbol{\alpha}'$. It is however possible for selector reversal to result in the same code, that is, a code with the same standard representation. □

C. Codes That are Invariant Under Selector Reversal

Consider a selector $\boldsymbol{\alpha} = (0\ 1\ \alpha_3\ \ldots\ \alpha_{\ell-1}\ \alpha_\ell)$. The reversal is $\overline{\boldsymbol{\alpha}} = (\alpha_\ell\ \alpha_{\ell-1}\ \ldots\ \alpha_3\ 1\ 0)$. Following Theorem 1 we apply the affine transformation to obtain the selector of the new code in standard form, $\boldsymbol{\alpha}' = \frac{\overline{\boldsymbol{\alpha}} - \alpha_\ell}{\alpha_{\ell-1} - \alpha_\ell}$.

For the two codes to be the equivalent through the reversal transformation and affine transformation, we must have $\boldsymbol{\alpha}' = \boldsymbol{\alpha}$, and so

$$\alpha_i = \frac{\alpha_{\ell+1-i} - \alpha_\ell}{\alpha_{\ell-1} - \alpha_\ell} \tag{2}$$

for all $i = 1, \ldots, \ell$.

The $i = 1$ and $i = 2$ conditions imply $\alpha_1 = \frac{\alpha_\ell - \alpha_\ell}{\alpha_{\ell-1} - \alpha_\ell} = 0$ and $\alpha_2 = \frac{\alpha_{\ell-1} - \alpha_\ell}{\alpha_{\ell-1} - \alpha_\ell} = 1$. These simply represent the transformation to the standard representation. When $\ell = 2$ we only have those conditions and there is only a single selector in standard form $(0\,1)$, and it satisfies this condition.

The conditions for $i = \ell$ and $i = \ell - 1$ reduce to the same condition for $\boldsymbol{\alpha}'$ and $\boldsymbol{\alpha}$ to be equal, that is

$$\alpha_\ell - \alpha_{\ell-1} = 1\,. \tag{3}$$

Substituting this back into (2) gives the reduced condition

$$\alpha_i = \alpha_\ell - \alpha_{\ell+1-i}. \tag{4}$$

The simplicity of this reduction means we can count the number of codes with the specified invariance. In the case $\ell = 3$ the only selector with such reversal invariance is $(0\,1\,2)$, since (3) becomes $\alpha_3 - \alpha_2 = 1$, or $\alpha_3 = 2$.

Theorem 3: The number of length $\ell \geq 4$ selectors $\boldsymbol{\alpha} = (0\ 1\ \alpha_3\ \ldots\ \alpha_{\ell-1}\ \alpha_\ell)$, over a prime field F_q, $q \geq \ell$, which specify a code with the same standard form before and after reversal is given by the expression

$$\frac{(q-3)!!}{\left(q - \left(2\lfloor\frac{\ell}{2}\rfloor + 1\right)\right)!!}. \tag{5}$$

where $x!! = x(x-2)(x-4)\ldots(x \bmod 2 + 2)$.

Proof: We proceed by identifying the number of relations which restrict the values of the selector elements. From (3), we obtain the relation $\alpha_\ell - \alpha_{\ell-1} = 1$. The value of α_ℓ cannot be equal to 0 or 1 as these values already appear in the selector vector. Furthermore, setting $\alpha_\ell = 2$ gives $\alpha_{\ell-1} = 1$, which has already appeared in the selector. This relation thus allows $(q-3)$ values to be chosen for α_ℓ, with no freedom in the subsequent choice of $\alpha_{\ell-1}$. For $\ell = 4$ we obtain only this enumeration and so we have $(q-3)$ such selectors.

If $\ell \geq 6$ is even we obtain from (4) a list of $\ell/2 - 2$ equations $\alpha_3 + \alpha_{\ell-2} = \alpha_\ell, \alpha_4 + \alpha_{\ell-3} = \alpha_\ell, \ldots \alpha_{\ell/2} + \alpha_{\ell/2+1} = \alpha_\ell$. Each equation implies we can choose one of the components other than α_ℓ independently, and obtain the other relative to α_ℓ and that choice. In making the choice we must avoid all the values already used in the selector. For the first of those equations we need to avoid 0, 1, $\alpha_{\ell-1}$ and α_ℓ. In addition we must avoid making α_3 and $\alpha_{\ell-2}$ equal to each other, that is avoid $\alpha_3 = \alpha_{\ell-2} = 2^{-1}\alpha_\ell$. We thus have $(q-5)$ possibilities for α_3, which then fixes $\alpha_{\ell-2}$ also. Each subsequent equation is used to choose one selector component and derive a second one. The chosen selector should avoid the previous values chosen and derived for the selector values, as well as $2^{-1}\alpha_\ell$. For the jth equation then we have $q - (2j+3)$ possible values. Each of these equations results in additional possibilities independent of the $(q-3)$ factor from the freedom described in the first paragraph of the proof. The total number of equations is therefore

$$(q-3)\Pi_1^{\ell/2-2} q - (2j+3) = \frac{(q-3)!!}{(q-(\ell+1))!!}.$$

If ℓ is odd the selector component for $i = (\ell+1)/2$ is evaluated through (4) as $2\alpha_{(\ell+1)/2} = \alpha_\ell$. For any of the $(q-3)$ valid α_ℓ there is always a $\alpha_{(\ell+1)/2}$ not equal to $0, 1, \alpha_{\ell-1}$ or α_ℓ. This is also the very value avoided in the "odd" counting to ensure α_i and $\alpha_{\ell+1-i}$ are not equal, so we avoid all previously specifed selector values also. Thus the number of reversal invariant selectors for odd ℓ is equal to the number of reversal invariant selectors for the even number $\ell - 1$. This is represented by the use of the floor function in (5). □

IV. Deletion Correcting Capability Bounds and the Enumeration of Inequivalent Codes for $\tilde{G}(\ell, q, 2)$

In this section we give a bound on the deletion correcting capability of codes and enumerate, and in some cases list, the inequivalent $\tilde{G}(\ell, q, 2)$ codes.

Theorem 4: For an RS code with $k = 2$ and $\ell \geq 3$, the largest deletion correcting capability possible is $\ell - 3$.

Proof: Recall from Section II-C that the RS codewords associated with polynomials of degree 0 have constant components and so have one subword of constant component of any length t. Furthermore, codewords associated with polynomials of degree exactly 1 have all subwords distinct. For a codeword associated with a polynomial of degree 1, there are $\frac{\ell!}{(\ell-t)!t!}$ subwords of length $\ell - t$. To be able to correct t deletions these must be distinct from the subwords of every other codeword associated with a polynomial of degree 1. Thus, we need $(q^2 - q)\frac{\ell!}{(\ell-t)!t!}$ distinct words of length $\ell - t$. For a given field F_q there are $q^{\ell-t}$ words of length $\ell - t$.

Thus there can only be enough subwords to correct deletions if

$$(q^2 - q)\frac{\ell!}{(\ell-t)!t!} \leq q^{\ell-t}.$$

Let $t = \ell - 2$. The equation reduces to $(q-1)\ell(\ell-1)/2 \leq q$. But this cannot be satisfied for $\ell \geq 3$ and so no RS code of dimension $k = 2$ is capable of correcting $(\ell - 2)$ deletions.

Let $t = (\ell - 3)$. For this case the condition above reduces to $(q-1)\ell(\ell-1)(\ell-2)/6 \leq q^2$, which can be satisfied by large enough q. □

A. Experimental Results

We have performed extensive computer searches to find inequivalent codes that have the best performance, in one of two senses which we will describe. We are also interested in determining the number of codes with such properties.

We firstly consider codes that satisfy the bound in Theorem 4. In Table I we give the current state of our computer search to find codes with the highest deletion correcting capability for $\ell \leq 25$. These results are significant improvements over previously reported results [16]. We

TABLE I

A Tabulation of Experimental Upper Bounds on the Value of the $\sigma(\ell, q, 2)$ for Prime q. The Highest Deletion Correcting Capability Is Given by $\ell - \sigma(\ell, q, 2)$. We Emphasize These Exhaustive and Selective Results Are for Codes With Unit Multipliers (i.e., for RS Codes). The (ℓ, q) Classes Marked With a $*$ Have Been Exhaustively Surveyed. The Rows and Columns Label the Prime Value q and Code Length ℓ, Respectively

$q \backslash \ell$	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	q
5	4*	4*																					5
7	3*	4*	4*	5*																			7
11	3*	4*	4*	4*	5*	5*	6*	7*															11
13	3*	3*	4*	4*	5*	5*	6*	6*	7*	7													13
17	3*	3*	4*	4*	4*	5*	5	6	6	7	7	8	8	9									17
19	3*	3*	4*	4*	4*	5*	5	6	6	7	7	8	8	8	9	9							19
23	3*	3*	3*	4*	4*	5	5	5	6	6	7	7	7	8	8	9	9	10	10	11			23
29	3*	3*	3*	4*	4	4	5	5	5	6	6	7	7	8	8	8	9	9	10	10			29
31	3*	3*	3*	4*	4	4	5	5	5	6	6	7	7	7	8	8	9	9	9	10	10		31
37	3*	3*	3*	4*	4	4	5	5	5	6	6	6	7	7	7	8	8	8	9	9	10		37
41	3*	3*	3*	4*	4	4	4	5	5	5	6	6	6	7	7	7	8	8	9	9	9		41
43	3*	3*	3*	4*	4	4	4	5	5	5	6	6	6	7	7	7	8	8	8	9	9		43
47	3*	3*	3*	3*	4	4	4	5	5	5	6	6	6	7	7	7	7	8	8	9	9	9	47
53	3*	3*	3*	3	4	4	4	5	5	5	5	6	6	6	7	7	7	7	8	9	9	9	53
59	3*	3*	3*	3	4	4	4	4	5	5	5	6	6	6	7	7	7	7	8	8	9	9	59
61	3*	3*	3*	3	4	4	4	4	5	5	5	6	6	6	7	7	7	7	8	8	8	9	61
67	3*	3*	3*	3	4	4	4	4	5	5	5	6	6	6	6	7	7	7	8	8	8	9	67
71	3*	3*	3*	3	3	4	4	4	5	5	5	6	6	6	6	7	7	7	8	8	8	8	71
73	3*	3*	3*	3	3	4	4	4	5	5	5	5	6	6	6	7	7	7	7	8	8	8	73
79	3*	3*	3*	3	3	4	4	4	5	5	5	5	6	6	6	6	7	7	7	8	8	8	79
83	3*	3*	3*	3	3	4	4	4	5	5	5	5	6	6	6	6	7	7	7	7	8	8	83
89	3*	3*	3*	3	3	4	4	4	5	5	5	5	5	6	6	6	7	7	7	7	8	8	89
97	3*	3*	3*	3	3	4	4	4	4	5	5	5	5	6	6	6	6	7	7	7	8	8	97
101	3*	3*	3	3	3	4	4	4	4	5	5	5	5	6	6	6	6	7	7	7	8	8	101
103	3*	3*	3	3	3	4	4	4	4	5	5	5	5	6	6	6	6	7	7	7	7	8	103
107	3*	3*	3	3	3	4	4	4	4	5	5	5	5	6	6	6	6	6	7	7	7	8	107
109	3*	3*	3	3	3	4	4	4	4	5	5	5	5	6	6	6	6	6	7	7	7	8	109
113	3*	3*	3	3	3	4	4	4	4	5	5	5	5	6	6	6	6	6	6	7	7	8	113
127	3*	3*	3	3	3	4	4	4	4	5	5	5	5	5	6	6	6	6	6	7	7	7	127
131	3*	3*	3	3	3	4	4	4	4	5	5	5	5	5	6	6	6	6	6	7	7	7	131
137	3*	3*	3	3	3	4	4	4	4	4	5	5	5	5	6	6	6	6	6	7	7	7	137
139	3*	3*	3	3	3	3	4	4	4	4	5	5	5	5	5	6	6	6	6	7	7	7	139
149	3*	3*	3	3	3	3	4	4	4	4	5	5	5	5	5	6	6	6	6	7	7	7	149
151	3*	3*	3	3	3	3	4	4	4	4	5	5	5	5	5	6	6	6	6	6	7	7	151
157	3*	3*	3	3	3	3	4	4	4	4	5	5	5	5	5	6	6	6	6	6	7	7	157
163	3*	3*	3	3	3	3	4	4	4	4	5	5	5	5	5	6	6	6	6	6	7	7	163
167	3*	3*	3	3	3	3	4	4	4	4	5	5	5	5	5	6	6	6	6	6	7	7	167
173	3*	3*	3	3	3	3	4	4	4	4	5	5	5	5	5	6	6	6	6	6	7	7	173
179	3*	3*	3	3	3	3	4	4	4	4	5	5	5	5	5	6	6	6	6	6	7	7	179
181	3*	3*	3	3	3	3	4	4	4	4	4	5	5	5	5	5	6	6	6	6	7	7	181
191	3*	3*	3	3	3	3	4	4	4	4	4	5	5	5	5	5	6	6	6	6	6	7	191
193	3*	3*	3	3	3	3	4	4	4	4	4	5	5	5	5	5	6	6	6	6	6	7	193
197	3*	3*	3	3	3	3	4	4	4	4	4	5	5	5	5	5	6	6	6	6	6	7	197
199	3*	3*	3	3	3	3	4	4	4	4	4	5	5	5	5	5	6	6	6	6	6	7	199
211	3*	3*	3	3	3	3	4	4	4	4	4	5	5	5	5	5	6	6	6	6	6	7	211
223	3*	3*	3	3	3	3	4	4	4	4	4	5	5	5	5	5	6	6	6	6	6	7	223
227	3*	3*	3	3	3	3	4	4	4	4	4	5	5	5	5	5	6	6	6	6	6	7	227
229	3*	3*	3	3	3	3	4	4	4	4	4	5	5	5	5	5	6	6	6	6	6	7	229
233	3*	3*	3	3	3	3	3	4	4	4	4	5	5	5	5	5	5	6	6	6	6	6	233
389	3*	3	3	3	3	3	3	3	4	4	4	4	4	5	5	5	5	5	5	6	6	6	389
683	3	3	3	3	3	3	3	3	3	4	4	4	4	4	4	5	5	5	5	5	5	5	683
1093	3	3	3	3	3	3	3	3	3	3	4	4	4	4	4	4	4	5	5	5	5	5	1093
1747	3	3	3	3	3	3	3	3	3	3	3	4	4	4	4	4	4	4	4	5	5	5	1747
2477	3	3	3	3	3	3	3	3	3	3	3	3	4	4	4	4	4	4	4	4	4	5	2477
3499	3	3	3	3	3	3	3	3	3	3	3	3	3	4	4	4	4	4	4	4	4	4	3499
4877	3	3	3	3	3	3	3	3	3	3	3	3	3	3	4	4	4	4	4	4	4	4	4877
6619	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	4	4	4	4	4	4	4	6619
8849	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	4	4	4	4	4	4	8849
11987	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	4	4	4	4	4	11987
15227	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	4	4	4	4	15227
18979	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	4	4	4	18979
23993	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	4	4	23993
29959	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	4	29959
36997	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	36997
	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	

also list, in Table II, examples of codes with the highest deletion correcting capability. We note that the results in Table I are obtained by a mix of exhaustive and non-exhaustive searches. The former cases are marked by '*'. The entries in the table are of $\sigma(\ell, q, 2)$, which is the length of the shortest subword that uniquely identifies codewords of codes in $\tilde{G}(\ell, q, 2)$ and so $\sigma(\ell, q, 2) = 3$ means that the deletion correcting capability is $\ell - 3$, and thus the highest possible according to Theorem 4. We observe that for any ℓ in the table, the bound in The-

TABLE II
SOME EXAMPLES OF THE BEST CODES FOUND, IN THE SENSE OF SMALLEST q FOR GIVEN ℓ AND r. THE CODES IN THE FIRST SECTION OF THE TABLE ARE NOT OPTIMAL, IN THE SENSE $r < \ell - 3$, BUT ARE SIGNIFICANT IMPROVEMENTS ON THE RESULTS OF [16]. THE CODES IN THE SECOND SECTION OF THE TABLE ARE OPTIMAL, IN THE SENSE $r = \ell - 3$

q	ℓ	r	α
11	9	4	(0 1 2 10 4 6 5 8 9)
13	13	6	(0 1 7 8 3 11 4 2 12 9 6 5 10)
17	12	6	(0 1 8 10 6 7 16 15 5 12 3 13)
23	11	6	(0 1 20 4 14 16 3 15 8 2 19)
23	13	7	(0 1 8 12 17 21 10 13 11 19 4 6 7)
29	9	5	(0 1 13 11 26 27 4 18 10)
41	10	6	(0 1 37 5 12 39 30 29 11 24)
59	11	7	(0 1 54 28 26 15 40 17 12 35 43)
71	8	5	(0 1 64 42 70 48 40 41)
139	9	6	(0 1 5 95 129 78 79 88 113)
233	10	7	(0 1 2 9 135 227 68 202 174 14)
389	11	8	(0 1 2 5 7 120 360 18 99 281 378)
683	12	9	(0 1 2 5 7 18 46 434 437 534 177 409)
1093	13	10	(0 1 2 5 7 18 24 61 1008 807 707 1019 931)
1747	14	11	(0 1 2 5 7 18 24 44 59 608 1518 692 1478 731)
2477	15	12	(0 1 2 5 7 18 24 44 59 67 903 1839 2209 2465 1617)
3499	16	13	(0 1 2 5 7 18 24 44 59 67 152 1272 3192 3312 143 2227)
4877	17	14	(0 1 2 5 7 18 24 44 59 67 101 218 1931 4835 2092 4494 495)
6619	18	15	(0 1 2 5 7 18 24 44 59 67 101 218 358 2625 4937 422 4154 6532)
8849	19	16	(0 1 2 5 7 18 24 44 59 67 101 218 225 358 4815 5157 8393 3040 5265)
11987	20	17	(0 1 2 5 7 18 24 44 59 67 101 225 250 357 422 1975 9561 11780 6675 6744)
15227	21	18	(0 1 2 5 7 18 24 44 59 67 101 218 225 333 399 591 2402 15049 13124 6960 12509)
18979	22	19	(0 1 2 5 7 18 24 44 59 67 101 218 225 279 410 476 729 7798 14501 1839 15348 4403)
23993	23	20	(0 1 2 5 7 18 24 44 59 67 101 218 225 279 410 476 637 910 10845 7925 19933 16366 11382)
29959	24	21	(0 1 2 5 7 18 24 44 59 67 101 218 225 279 358 440 467 570 1634 21776 25173 9744 28169 29747)
36997	25	22	(0 1 2 5 7 18 24 44 59 67 101 218 225 279 422 467 567 570 688 1470 10747 9562 15939 20406 33155)
45497	26	23	(0 1 2 5 7 18 24 44 59 67 101 218 225 279 410 422 471 643 781 1017 1389 13605 22405 13535 44530 27227)
56999	27	24	(0 1 2 5 7 18 24 44 59 67 101 218 225 279 358 422 467 471 736 791 936 2580 17735 33926 13462 16727 32794)
67499	28	25	(0 1 2 5 7 18 24 44 59 67 101 218 225 279 358 422 467 471 570 580 1572 1763 2284 51986 38417 58446 31056 60095)
86993	29	26	(0 1 2 5 7 18 24 44 59 67 101 218 225 279 358 422 471 570 580 643 1171 1196 1363 1942 7146 30287 76088 79401 ... ... 73323)
99991	30	27	(0 1 2 5 7 18 24 44 59 67 101 218 225 279 358 422 467 471 570 637 688 921 1356 1590 3544 7015 38270 65856 ... ... 81477 12911)
120691	31	28	(0 1 2 5 7 18 24 44 59 67 101 218 225 279 358 422 467 471 570 580 1084 1186 1337 1750 2703 4460 6877 119243 ... ... 58051 90704 92833)
144983	32	29	(0 1 2 5 7 18 24 44 59 67 101 218 225 279 358 422 467 471 580 786 808 908 1119 1319 1753 2012 4206 6794 55281 ... ... 78151 34019 79376)
169991	33	30	(0 1 2 5 7 18 24 44 59 67 101 218 225 279 358 422 467 471 570 637 805 921 1048 1366 1972 2452 3450 4421 9944... ... 108285 162094 74880 109191)
189997	34	31	(0 1 2 5 7 18 24 44 59 67 101 218 225 279 358 422 467 471 570 580 736 1084 1196 1235 1275 3014 3873 4535 5815 ... ... 21962 169804 137158 38325 118616)
239999	35	32	(0 1 2 5 7 18 24 44 59 67 101 218 225 279 358 422 467 471 570 580 736 825 1084 1223 1257 1337 1926 1966 2288 ... ... 6228 9776 12390 79449 121080 209265)
274973	36	33	(0 1 2 5 7 18 24 44 59 67 101 218 225 279 358 422 467 471 570 580 736 825 1084 1223 1281 1763 2131 2730 2854 ... ... 4634 6954 10480 37213 73629 165748 132173)

orem 4 can be achieved with equality if q is sufficiently large (in each column there is a row with an entry equal to 3).

Second, we consider codes that satisfy a property, for example the highest deletion correction capability, over the smallest size prime field. Of particular value are the smallest fields for which codes with $r = \ell - 3$ have been found. For nonexhaustive searches the smallest field provides an upper bound on the value. In Table II we give examples of such codes, specified by the selector, for $\ell \leq 36$.

Let $\mathcal{Q}(\ell, r)$ denote the smallest prime q for which we have a code in $\tilde{G}(\ell, q, 2)$ with a deletion correcting capability r. Then $\mathcal{Q}(\ell, \ell - 3)$ gives the smallest q, for a given ℓ, for which $\sigma(\ell, q, 2) = 3$, that is for which the code meets the deletion correcting bound. Using Table I we can see that $\mathcal{Q}(4, 1) = 7$ and so there is no single deletion correcting code of length 4 for $q = 3$ or $q = 5$.

Using Table I we have $\mathcal{Q}(4, 1) = 7$, $\mathcal{Q}(5, 2) = 13$, $\mathcal{Q}(6, 3) = 23$, $\mathcal{Q}(7, 4) = 47$ and $\mathcal{Q}(8, 5) = 71$, all as exhaustively tested minimums. These values suggest the smallest prime field, with the best deletion correcting capability possible, grows quickly as we increase the length. We see this is supported by the current experimental evidence in Fig. 1.

The value of the upper bound on $\mathcal{Q}(\ell, r)$ for $4 \leq \ell \leq 36$ is given in the array below and in Fig. 1. Note again that the "*" entries are proven, by exhaustive searches, to be minimums.

ℓ	$Q(\ell, \ell-3)$	ℓ	$Q(\ell, \ell-3)$	ℓ	$Q(\ell, \ell-3)$
4	7*	15	2477	26	45497
5	13*	16	3499	27	56999
6	23*	17	4877	28	67499
7	47*	18	6619	29	86933
8	71*	19	8849	30	99991
9	139	20	11987	31	120691
10	233	21	15227	32	144983
11	389	22	18979	33	169991
12	683	23	23993	34	189997
13	1093	24	29959	35	239999
14	1747	25	36997	36	274973

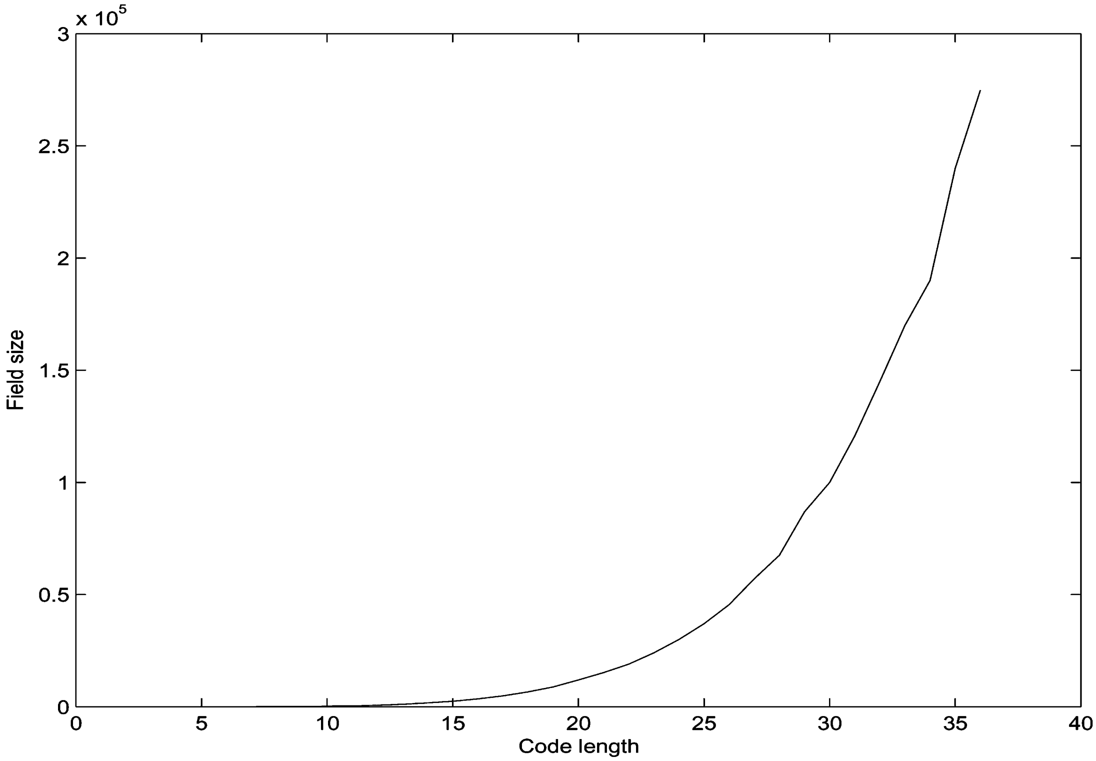

Fig. 1. This figure illustrates the currently determined values of $\mathcal{Q}(\ell, \ell - 3)$ for ℓ from 4 to 36. We see that the size of q increases rapidly, approximately as a constant multiple of $\ell^{5.25}$.

Table II has two parts. The first part of the table gives selective examples of *the best* codes, in the sense of smallest q for a give (ℓ, r) pair. For example, for $\ell = 9$ and $r = 4$, the smallest $q = 11$. Some examples for smaller ℓ and r appear in Section IV-B, in the inequivalence sets given in Appendix, and in Tables IV–VII.

In the second part of Table II, examples of codes with $\sigma(\ell, q, 2) = 3$, the best deletion correcting capability possible, are given for $8 \leq \ell \leq 36$.

B. Tabulations of Inequivalence Set Cardinalities

In Section III, we considered two isomorphisms of selector vectors, that result in codes equivalent to the original one. We would like to enumerate inequivalent $\tilde{G}(\ell, q, 2)$ codes for a given set of parameters. Let $\mathbb{P}[q, \ell]_r$ denote the set of inequivalent codes (in standard form) of length ℓ over a field of size q and with deletion correcting capability of r.

Note that $\mathbb{P}[\mathcal{Q}(\ell, r), \ell]_r$ denotes the set of codes with length ℓ, deletion correcting capability r, and the smallest known q with that length and deletion correcting capability. In Table III we tabulate the cardinalities of the various inequivalent code sets, for primes from 5 to 97 and for various lengths. For $q = 5, 7$ and 11 we have completed exhaustive enumerations for lengths up to and including $\ell = q$. As the value of the field q increases, the exhaustive enumerations become increasingly time consuming; thus for $q > 11$ we do not have enumerations for all lengths up to q.

In cases where there are only a small number of distinct codes for given small q and small ℓ, we explicitly give the codes, in terms of the selectors, in Section IV-D and in Tables IV–VII. The $\mathbb{P}[\mathcal{Q}(\ell, r), \ell]_r$ of small cardinality are listed in Appendix.

An interesting result of our search is the explicit construction of a code whose parameters achieve a bound, proposed elsewhere, with equality. In particular, for $q = 7$, $\ell = 7$, it has been proven in [4], [9] that codes with deletion correcting capability greater than two cannot exist. We have found a code with $q = 7$, $\ell = 7$ and $r = 2$, thus providing an explicit construction for that bound.

C. The Distribution of Deletion Correcting Capabilities

In cases where we have undertaken exhaustive enumerations the pro-portion of codes with particular deletion correcting capabilities is of interest. We can compare this with results in [16, Table 1]. For example, in [16] it was noted that 1% of all $\mathrm{GRS}_k(\ell, q, \boldsymbol{\alpha}, \mathbf{v})$ codes with $k = 2$, $q = 7$, $\ell = 4$ are capable of correcting 1 deletion. We have found that 45% of all $\tilde{G}(\ell, q, 2)$ codes with $q = 7$ and $\ell = 4$ are capable of correcting 1 deletion. The relative proportion of codes with higher deletion correcting capabilities supports the emphasis placed on RS codes. Similarly with $k = 2$, $q = 7$, $\ell = 5$ 60%[4] of $\mathrm{GRS}_k(\ell, q, \boldsymbol{\alpha}, \mathbf{v})$ codes can correct one deletion, against 88% of $\tilde{G}(\ell, q, k)$ codes.

D. Some "Nice" Codes

In this section, and in Appendix, we give the explicit codebooks for some of the smaller RS codes capable of correcting deletions, that is some of the $\mathbb{P}[\mathcal{Q}(\ell, r), \ell]_r$ codes (see Section IV-B). The codes are labelled in terms of q, ℓ, r and the selector $\boldsymbol{\alpha}$. In Appendix we use capitalized Latin letters to denote the numbers from $10(A)$ to $22(M)$, allowing us to more compactly represent the codewords for $q \geq 11$.

[4]The 60% and 40% in the second to last column of Table I of [16] should each be one row higher. That is for $q = 7$, $\ell = 5$, we have 60% with $s = 4$ $(r = 1)$ and 40% with $s = 5$ $(r = 0)$.

We note that the codes of interest over GF(q) have q^2 codewords. The first two examples are listed in the text below, the remaining examples can be found in Tables IV–VII.

0 0 0 0 0 0	1 1 1 1 1 1	2 2 2 2 2 2	3 3 3 3 3 3
4 4 4 4 4 4	5 5 5 5 5 5	6 6 6 6 6 6	0 1 6 5 3 4
1 2 0 6 4 5	2 3 1 0 5 6	3 4 2 1 6 0	4 5 3 2 0 1
5 6 4 3 1 2	6 0 5 4 2 3	0 2 5 3 6 1	1 3 6 4 0 2
2 4 0 5 1 3	3 5 1 6 2 4	4 6 2 0 3 5	5 0 3 1 4 6
6 1 4 2 5 0	0 3 4 1 2 5	1 4 5 2 3 6	2 5 6 3 4 0
3 6 0 4 5 1	4 0 1 5 6 2	5 1 2 6 0 3	6 2 3 0 1 4
0 4 3 6 5 2	1 5 4 0 6 3	2 6 5 1 0 4	3 0 6 2 1 5
4 1 0 3 2 6	5 2 1 4 3 0	6 3 2 5 4 1	0 5 2 4 1 6
1 6 3 5 2 0	2 0 4 6 3 1	3 1 5 0 4 2	4 2 6 1 5 3
5 3 0 2 6 4	6 4 1 3 0 5	0 6 1 2 4 3	1 0 2 3 5 4
2 1 3 4 6 5	3 2 4 5 0 6	4 3 5 6 1 0	5 4 6 0 2 1
6 5 0 1 3 2			

There is a single code in $\mathbb{P}[\mathcal{Q}(5,1) = 5,5]_1$. It is specified by the selector $\boldsymbol{\alpha} = (0\ 1\ 4\ 2\ 3)$ as shown in the first table at the bottom of the page.

There is a single code in $\mathbb{P}[\mathcal{Q}(6,2) = 7,6]_2$. It is specified by the selector $\boldsymbol{\alpha} = (0\ 1\ 6\ 5\ 3\ 4)$ as shown in the second table at the bottom of the page.

Another nice code, worthy of mention here, is the lone member of $\mathbb{P}[\mathcal{Q}(8,5) = 71,8]_5$, specified by the selector $\alpha = (0\ 1\ 64\ 42\ 70\ 48\ 40\ 41)$. Not only is this the only code in this equivalence set, there exist no GRS codes of this length with the same deletion correcting capability and a shorter length.

V. Summary and Discussion

We have presented an investigation into the classification of the deletion correcting capabilities of prime RS codes with dimension $k = 2$.

We have proven that the deletion correcting capability is invariant under affine transformations and reversal of the selector. Using these isomorphisms we have focused on inequivalent codes and enumerated $\tilde{G}(\ell, q, 2)$ classes with small parameter values. We have listed the inequivalent codes in cases where the sets are themselves small, and in some cases given the codebooks too.

We have proven that for $\tilde{G}(\ell, q, 2)$ codes $r \leq \ell - 3$. We have identified examples of codes meeting this bound for $\ell \leq 36$. For example, in Table II we give a length $\ell = 36$ code capable of correcting 33 deletions. This code is over a F_{274973}, with about 7.55×10^{10} codewords.

Let us conclude with some open questions for consideration. Firstly, How do we design codes with a particular deletion correcting capability? For the class of codes considered in this correspondence, the code is defined by a selector, and the question becomes; How do we choose a selector to provide a particular deletion correcting capability? Another related question is; Given a selector, how can we determine the deletion correcting capability of the code generated by the selector?

We note that the question; What is the deletion correcting capability of a code with specified q, ℓ and selector $\boldsymbol{\alpha}$?; can be closely related to the problem of decoding. This will be discussed in future work.

Closer to the direction of our work here, we would like to be able to answer, for a given $\tilde{G}(\ell, q, 2)$ class, Does there exist a selector specifying a code capable of correcting r deletions? Thus we want to more precisely determine the values of $\mathcal{Q}(\ell, r)$, that is the smallest q for which we have a code of length ℓ capable of correcting r deletions. This question is somewhat complicated by an unresolved proposition [16];

0 0 0 0 0	1 1 1 1 1	2 2 2 2 2	3 3 3 3 3	4 4 4 4 4
0 1 4 2 3	1 2 0 3 4	2 3 1 4 0	3 4 2 0 1	4 0 3 1 2
0 2 3 4 1	1 3 4 0 2	2 4 0 1 3	3 0 1 2 4	4 1 2 3 0
0 3 2 1 4	1 4 3 2 0	2 0 4 3 1	3 1 0 4 2	4 2 1 0 3
0 4 1 3 2	1 0 2 4 3	2 1 3 0 4	3 2 4 1 0	4 3 0 2 1

0 0 0 0 0 0	1 1 1 1 1 1	2 2 2 2 2 2	3 3 3 3 3 3
4 4 4 4 4 4	5 5 5 5 5 5	6 6 6 6 6 6	0 1 6 5 3 4
1 2 0 6 4 5	2 3 1 0 5 6	3 4 2 1 6 0	4 5 3 2 0 1
5 6 4 3 1 2	6 0 5 4 2 3	0 2 5 3 6 1	1 3 6 4 0 2
2 4 0 5 1 3	3 5 1 6 2 4	4 6 2 0 3 5	5 0 3 1 4 6
6 1 4 2 5 0	0 3 4 1 2 5	1 4 5 2 3 6	2 5 6 3 4 0
3 6 0 4 5 1	4 0 1 5 6 2	5 1 2 6 0 3	6 2 3 0 1 4
0 4 3 6 5 2	1 5 4 0 6 3	2 6 5 1 0 4	3 0 6 2 1 5
4 1 0 3 2 6	5 2 1 4 3 0	6 3 2 5 4 1	0 5 2 4 1 6
1 6 3 5 2 0	2 0 4 6 3 1	3 1 5 0 4 2	4 2 6 1 5 3
5 3 0 2 6 4	6 4 1 3 0 5	0 6 1 2 4 3	1 0 2 3 5 4
2 1 3 4 6 5	3 2 4 5 0 6	4 3 5 6 1 0	5 4 6 0 2 1
6 5 0 1 3 2			

TABLE III
THE CARDINALITIES OF INEQUIVALENCE SETS, FOR PRIMES FROM 5 TO 97 AND FOR VARIOUS LENGTHS

$q = 5$

$r\backslash\ell$	4	5
0	4	3
1		1

$q = 7$

$r\backslash\ell$	4	5	6	7
0	6	4	4	3
1	6	28	59	49
2			1	12

$q = 11$

$r\backslash\ell$	4	5	6	7	8	9	10	11
0	13	13	11	7	7	7	7	5
1	27	243	883	1235	1270	1094	793	363
2			642	6340	22938	32414	25478	10491
3				2	6121	57298	145440	106689
4						3	9914	64084

$q = 13$

$r\backslash\ell$	4	5	6	7	8	9	10	11	12
0	15	12	10	9	7	7	7	7	7
1	45	487	1488	1773	1813	1732	1544	1307	934
2		1	2502	25465	81044	110215	109713	89581	56822
3				513	83696	708752	2056085	2458946	1692824
4						11136	1160012	7375726	14179669
5								54593	4030064

$q = 17$

$r\backslash\ell$	4	5	6	7	8	9
0	22	21	19	19	19	17
1	90	1277	3631	4812	5689	6374
2		74	12814	147578	402158	609757
3				27855	1394663	11969412
4					111	3631480

$q = 19$

$r\backslash\ell$	4	5	6	7	8	9
0	24	22	22	21	19	19
1	120	1834	4717	6210	7535	8668
2		192	23933	269117	656491	993236
3				96044	3788517	29200726
4					4142	18807655

$q = 23$

$r\backslash\ell$	4	5	6	7	8
0	31	31	31	31	31
1	189	3280	8191	11374	14508
2		689	63777	693327	1611112
3			1	516388	17613045
4					299224

$q = 29$

$r\backslash\ell$	4	5	6	7
0	40	39	37	37
1	324	6298	14346	19599
2		2451	196480	1885334
3			67	2939142

$q = 31$

$r\backslash\ell$	4	5	6	7
0	42	40	38	33
1	378	7430	16380	22050
2		3506	268715	2431981
3			243	4671600

$q = 37$

$r\backslash\ell$	4	5	6	7
0	51	48	46	45
1	561	11836	24606	33228
2		7768	601467	4791257
3			2746	14653934

$q = 41$

$r\backslash\ell$	4	5	6	7
0	58	57	53	49
1	702	15302	31989	43932
2		12077	946172	7094296
3			9482	27407827

$q = 43$

$r\backslash\ell$	4	5	6	7
0	60	58	58	57
1	780	17065	34879	48323
2		14877	1164889	8322899
3			16174	36593361

$q = 47$

$r\backslash\ell$	4	5	6	7
0	67	67	67	67
1	945	21298	43641	61519
2		21227	1706722	11583603
3			38434	61661271
4				4

$q = 53$

$r\backslash\ell$	4	5	6
0	76	75	73
1	1224	28445	56550
2		33980	2831202
3			112175

$q = 59$

$r\backslash\ell$	4	5	6
0	85	85	85
1	1539	36509	71647
2		51214	4398356
3			271544

$q = 61$

$r\backslash\ell$	4	5	6
0	87	84	80
1	1653	39151	75387
2		58321	5030813
3			356856

$q = 67$

$r\backslash\ell$	4	5	6
0	96	94	94
1	2016	48350	92751
2		82528	7314081
3			719538

$q = 71$

$r\backslash\ell$	4	5	6
0	103	103	101
1	2277	55389	106688
2		101724	9192352
3			1077115

$q = 73$

$r\backslash\ell$	4	5	6
0	105	102	100
1	2415	58915	111166
2		112483	10243309
3			1307425

$q = 79$

$r\backslash\ell$	4	5	6
0	114	112	112
1	2850	70051	131978
2		149325	13884981
3			2225041

$q = 83$

$r\backslash\ell$	4	5	6
0	121	121	121
1	3159	78751	148602
2		177128	16789206
3			3030071

$q = 89$

$r\backslash\ell$	4	5	6
0	130	129	127
1	3654	91811	171674
2		226088	21846501
3			4696050

$q = 97$

$r\backslash\ell$	4	5	6
0	141	138	136
1	4371	110529	203372
2		304625	30051926
3			7951430

If $q_1 < q_2$ are prime powers then $\sigma(\ell, q_1, k) \geq \sigma(\ell, q_2, k)$.

That is, the deletion correction capability of codes does not decrease as q increases with fixed ℓ and k. In generating Table I we have not assumed this proposition to be true, we have explicitly identified codes in every case. The identification of codes in $\mathbb{P}[\mathcal{Q}(\ell, r), \ell]_r$ is also important.

TABLE IV

THERE ARE TWO CODES IN $\mathbb{P}[\mathcal{Q}(7,3)=11,7]_3$. ONE IS SPECIFIED BY THE SELECTOR $\boldsymbol{\alpha}=(0\ 1\ 2\ 8\ A\ 3\ 5)$

0 0 0 0 0 0 0	1 1 1 1 1 1 1	2 2 2 2 2 2 2	3 3 3 3 3 3 3	4 4 4 4 4 4 4	5 5 5 5 5 5 5	6 6 6 6 6 6 6	7 7 7 7 7 7 7
8 8 8 8 8 8 8	9 9 9 9 9 9 9	A A A A A A A	0 1 2 8 A 3 5	1 2 3 9 0 4 6	2 3 4 A 1 5 7	3 4 5 0 2 6 8	4 5 6 1 3 7 9
5 6 7 2 4 8 A	6 7 8 3 5 9 0	7 8 9 4 6 A 1	8 9 A 5 7 0 2	9 A 0 6 8 1 3	A 0 1 7 9 2 4	0 2 4 5 9 6 A	1 3 5 6 A 7 0
2 4 6 7 0 8 1	3 5 7 8 1 9 2	4 6 8 9 2 A 3	5 7 9 A 3 0 4	6 8 A 0 4 1 5	7 9 0 1 5 2 6	8 A 1 2 6 3 7	9 0 2 3 7 4 8
A 1 3 4 8 5 9	0 3 6 2 8 9 4	1 4 7 3 9 A 5	2 5 8 4 A 0 6	3 6 9 5 0 1 7	4 7 A 6 1 2 8	5 8 0 7 2 3 9	6 9 1 8 3 4 A
7 A 2 9 4 5 0	8 0 3 A 5 6 1	9 1 4 0 6 7 2	A 2 5 1 7 8 3	0 4 8 A 7 1 9	1 5 9 0 8 2 A	2 6 A 1 9 3 0	3 7 0 2 A 4 1
4 8 1 3 0 5 2	5 9 2 4 1 6 3	6 A 3 5 2 7 4	7 0 4 6 3 8 5	8 1 5 7 4 9 6	9 2 6 8 5 A 7	A 3 7 9 6 0 8	0 5 A 7 6 4 3
1 6 0 8 7 5 4	2 7 1 9 8 6 5	3 8 2 A 9 7 6	4 9 3 0 A 8 7	5 A 4 1 0 9 8	6 0 5 2 1 A 9	7 1 6 3 2 0 A	8 2 7 4 3 1 0
9 3 8 5 4 2 1	A 4 9 6 5 3 2	0 6 1 4 5 7 8	1 7 2 5 6 8 9	2 8 3 6 7 9 A	3 9 4 7 8 A 0	4 A 5 8 9 0 1	5 0 6 9 A 1 2
6 1 7 A 0 2 3	7 2 8 0 1 3 4	8 3 9 1 2 4 5	9 4 A 2 3 5 6	A 5 0 3 4 6 7	0 7 3 1 4 A 2	1 8 4 2 5 0 3	2 9 5 3 6 1 4
3 A 6 4 7 2 5	4 0 7 5 8 3 6	5 1 8 6 9 4 7	6 2 9 7 A 5 8	7 3 A 8 0 6 9	8 4 0 9 1 7 A	9 5 1 A 2 8 0	A 6 2 0 3 9 1
0 8 5 9 3 2 7	1 9 6 A 4 3 8	2 A 7 0 5 4 9	3 0 8 1 6 5 A	4 1 9 2 7 6 0	5 2 A 3 8 7 1	6 3 0 4 9 8 2	7 4 1 5 A 9 3
8 5 2 6 0 A 4	9 6 3 7 1 0 5	A 7 4 8 2 1 6	0 9 7 6 2 5 1	1 A 8 7 3 6 2	2 0 9 8 4 7 3	3 1 A 9 5 8 4	4 2 0 A 6 9 5
5 3 1 0 7 A 6	6 4 2 1 8 0 7	7 5 3 2 9 1 8	8 6 4 3 A 2 9	9 7 5 4 0 3 A	A 8 6 5 1 4 0	0 A 9 3 1 8 6	1 0 A 4 2 9 7
2 1 0 5 3 A 8	3 2 1 6 4 0 9	4 3 2 7 5 1 A	5 4 3 8 6 2 0	6 5 4 9 7 3 1	7 6 5 A 8 4 2	8 7 6 0 9 5 3	9 8 7 1 A 6 4
A 9 8 2 0 7 5							

TABLE V

THERE ARE THREE CODES IN $\mathbb{P}[\mathcal{Q}(9,4)=11,9]_4$. ONE IS SPECIFIED BY THE SELECTOR $\boldsymbol{\alpha}=(0\ 1\ 2\ A\ 4\ 6\ 5\ 8\ 9)$

0 0 0 0 0 0 0 0 0	1 1 1 1 1 1 1 1 1	2 2 2 2 2 2 2 2 2	3 3 3 3 3 3 3 3 3	4 4 4 4 4 4 4 4 4	5 5 5 5 5 5 5 5 5
6 6 6 6 6 6 6 6 6	7 7 7 7 7 7 7 7 7	8 8 8 8 8 8 8 8 8	9 9 9 9 9 9 9 9 9	A A A A A A A A A	0 1 2 A 4 6 5 8 9
1 2 3 0 5 7 6 9 A	2 3 4 1 6 8 7 A 0	3 4 5 2 7 9 8 0 1	4 5 6 3 8 A 9 1 2	5 6 7 4 9 0 A 2 3	6 7 8 5 A 1 0 3 4
7 8 9 6 0 2 1 4 5	8 9 A 7 1 3 2 5 6	9 A 0 8 2 4 3 6 7	A 0 1 9 3 5 4 7 8	0 2 4 9 8 1 A 5 7	1 3 5 A 9 2 0 6 8
2 4 6 0 A 3 1 7 9	3 5 7 1 0 4 2 8 A	4 6 8 2 1 5 3 9 0	5 7 9 3 2 6 4 A 1	6 8 A 4 3 7 5 0 2	7 9 0 5 4 8 6 1 3
8 A 1 6 5 9 7 2 4	9 0 2 7 6 A 8 3 5	A 1 3 8 7 0 9 4 6	0 3 6 8 1 7 4 2 5	1 4 7 9 2 8 5 3 6	2 5 8 A 3 9 6 4 7
3 6 9 0 4 A 7 5 8	4 7 A 1 5 0 8 6 9	5 8 0 2 6 1 9 7 A	6 9 1 3 7 2 A 8 0	7 A 2 4 8 3 0 9 1	8 0 3 5 9 4 1 A 2
9 1 4 6 A 5 2 0 3	A 2 5 7 0 6 3 1 4	0 4 8 7 5 2 9 A 3	1 5 9 8 6 3 A 0 4	2 6 A 9 7 4 0 1 5	3 7 0 A 8 5 1 2 6
4 8 1 0 9 6 2 3 7	5 9 2 1 A 7 3 4 8	6 A 3 2 0 8 4 5 9	7 0 4 3 1 9 5 6 A	8 1 5 4 2 A 6 7 0	9 2 6 5 3 0 7 8 1
A 3 7 6 4 1 8 9 2	0 5 A 6 9 8 3 7 1	1 6 0 7 A 9 4 8 2	2 7 1 8 0 A 5 9 3	3 8 2 9 1 0 6 A 4	4 9 3 A 2 1 7 0 5
5 A 4 0 3 2 8 1 6	6 0 5 1 4 3 9 2 7	7 1 6 2 5 4 A 3 8	8 2 7 3 6 5 0 4 9	9 3 8 4 7 6 1 5 A	A 4 9 5 8 7 2 6 0
0 6 1 5 2 3 8 4 A	1 7 2 6 3 4 9 5 0	2 8 3 7 4 5 A 6 1	3 9 4 8 5 6 0 7 2	4 A 5 9 6 7 1 8 3	5 0 6 A 7 8 2 9 4
6 1 7 0 8 9 3 A 5	7 2 8 1 9 A 4 0 6	8 3 9 2 A 0 5 1 7	9 4 A 3 0 1 6 2 8	A 5 0 4 1 2 7 3 9	0 7 3 4 6 9 2 1 8
1 8 4 5 7 A 3 2 9	2 9 5 6 8 0 4 3 A	3 A 6 7 9 1 5 4 0	4 0 7 8 A 2 6 5 1	5 1 8 9 0 3 7 6 2	6 2 9 A 1 4 8 7 3
7 3 A 0 2 5 9 8 4	8 4 0 1 3 6 A 9 5	9 5 1 2 4 7 0 A 6	A 6 2 3 5 8 1 0 7	0 8 5 3 A 4 7 9 6	1 9 6 4 0 5 8 A 7
2 A 7 5 1 6 9 0 8	3 0 8 6 2 7 A 1 9	4 1 9 7 3 8 0 2 A	5 2 A 8 4 9 1 3 0	6 3 0 9 5 A 2 4 1	7 4 1 A 6 0 3 5 2
8 5 2 0 7 1 4 6 3	9 6 3 1 8 2 5 7 4	A 7 4 2 9 3 6 8 5	0 9 7 2 3 A 1 6 4	1 A 8 3 4 0 2 7 5	2 0 9 4 5 1 3 8 6
3 1 A 5 6 2 4 9 7	4 2 0 6 7 3 5 A 8	5 3 1 7 8 4 6 0 9	6 4 2 8 9 5 7 1 A	7 5 3 9 A 6 8 2 0	8 6 4 A 0 7 9 3 1
9 7 5 0 1 8 A 4 2	A 8 6 1 2 9 0 5 3	0 A 9 1 7 5 6 3 2	1 0 A 2 8 6 7 4 3	2 1 0 3 9 7 8 5 4	3 2 1 4 A 8 9 6 5
4 3 2 5 0 9 A 7 6	5 4 3 6 1 A 0 8 7	6 5 4 7 2 0 1 9 8	7 6 5 8 3 1 2 A 9	8 7 6 9 4 2 3 0 A	9 8 7 A 5 3 4 1 0
A 9 8 0 6 4 5 2 1					

TABLE VI

THERE IS ONE CODE IN $\mathbb{P}[\mathcal{Q}(5,2)=13,5]_2$. IT IS SPECIFIED BY THE SELECTOR $\boldsymbol{\alpha}=(0\ 1\ 7\ 6\ 2)$. A DELETION CORRECTING CAPABILITY OF TWO IS THE MAXIMUM ACHIEVABLE BY $\tilde{G}(\ell,q,2)$ CODES OF LENGTH 5 (SEE THEOREM 4)

0 0 0 0 0	1 1 1 1 1	2 2 2 2 2	3 3 3 3 3	4 4 4 4 4	5 5 5 5 5	6 6 6 6 6	7 7 7 7 7	8 8 8 8 8	9 9 9 9 9
A A A A A	B B B B B	C C C C C	0 1 7 6 2	1 2 8 7 3	2 3 9 8 4	3 4 A 9 5	4 5 B A 6	5 6 C B 7	6 7 0 C 8
7 8 1 0 9	8 9 2 1 A	9 A 3 2 B	A B 4 3 C	B C 5 4 0	C 0 6 5 1	0 2 1 C 4	1 3 2 0 5	2 4 3 1 6	3 5 4 2 7
4 6 5 3 8	5 7 6 4 9	6 8 7 5 A	7 9 8 6 B	8 A 9 7 C	9 B A 8 0	A C B 9 1	B 0 C A 2	C 1 0 B 3	0 3 8 5 6
1 4 9 6 7	2 5 A 7 8	3 6 B 8 9	4 7 C 9 A	5 8 0 A B	6 9 1 B C	7 A 2 C 0	8 B 3 0 1	9 C 4 1 2	A 0 5 2 3
B 1 6 3 4	C 2 7 4 5	0 4 2 B 8	1 5 3 C 9	2 6 4 0 A	3 7 5 1 B	4 8 6 2 C	5 9 7 3 0	6 A 8 4 1	7 B 9 5 2
8 C A 6 3	9 0 B 7 4	A 1 C 8 5	B 2 0 9 6	C 3 1 A 7	0 5 9 4 A	1 6 A 5 B	2 7 B 6 C	3 8 C 7 0	4 9 0 8 1
5 A 1 9 2	6 B 2 A 3	7 C 3 B 4	8 0 4 C 5	9 1 5 0 6	A 2 6 1 7	B 3 7 2 8	C 4 8 3 9	0 6 3 A C	1 7 4 B 0
2 8 5 C 1	3 9 6 0 2	4 A 7 1 3	5 B 8 2 4	6 C 9 3 5	7 0 A 4 6	8 1 B 5 7	9 2 C 6 8	A 3 0 7 9	B 4 1 8 A
C 5 2 9 B	0 7 A 3 1	1 8 B 4 2	2 9 C 5 3	3 A 0 6 4	4 B 1 7 5	5 C 2 8 6	6 0 3 9 7	7 1 4 A 8	8 2 5 B 9
9 3 6 C A	A 4 7 0 B	B 5 8 1 C	C 6 9 2 0	0 8 4 9 3	1 9 5 A 4	2 A 6 B 5	3 B 7 C 6	4 C 8 0 7	5 0 9 1 8
6 1 A 2 9	7 2 B 3 A	8 3 C 4 B	9 4 0 5 C	A 5 1 6 0	B 6 2 7 1	C 7 3 8 2	0 9 B 2 5	1 A C 3 6	2 B 0 4 7
3 C 1 5 8	4 0 2 6 9	5 1 3 7 A	6 2 4 8 B	7 3 5 9 C	8 4 6 A 0	9 5 7 B 1	A 6 8 C 2	B 7 9 0 3	C 8 A 1 4
0 A 5 8 7	1 B 6 9 8	2 C 7 A 9	3 0 8 B A	4 1 9 C B	5 2 A 0 C	6 3 B 1 0	7 4 C 2 1	8 5 0 3 2	9 6 1 4 3
A 7 2 5 4	B 8 3 6 5	C 9 4 7 6	0 B C 1 9	1 C 0 2 A	2 0 1 3 B	3 1 2 4 C	4 2 3 5 0	5 3 4 6 1	6 4 5 7 2
7 5 6 8 3	8 6 7 9 4	9 7 8 A 5	A 8 9 B 6	B 9 A C 7	C A B 0 8	0 C 6 7 B	1 0 7 8 C	2 1 8 9 0	3 2 9 A 1
4 3 A B 2	5 4 B C 3	6 5 C 0 4	7 6 0 1 5	8 7 1 2 6	9 8 2 3 7	A 9 3 4 8	B A 4 5 9	C B 5 6 A	

A further question requiring resolution is; Can arbitrary multiplier (GRS) codes provide better deletion correcting capabilities than unit multiplier (RS) codes?

In this correspondence, we have only considered $k=2$, higher dimension codes need to be considered also. Some experimental results were given in [16]. It was also proven therein that the shortest subword

TABLE VII

THERE IS ONE CODE IN $\mathbb{P}[\mathcal{Q}(6,3) = 23, 6]_3$. IT IS SPECIFIED BY THE SELECTOR $\boldsymbol{\alpha} = (0\ 1\ G\ C\ 4\ 5)$. A DELETION CORRECTING CAPABILITY OF THREE IS THE MAXIMUM ACHIEVABLE BY $\tilde{G}(\ell, q, 2)$ CODES OF LENGTH 6 (SEE THEOREM 4)

0 0 0 0 0 0	1 1 1 1 1 1	2 2 2 2 2 2	3 3 3 3 3 3	4 4 4 4 4 4	5 5 5 5 5 5	6 6 6 6 6 6	7 7 7 7 7 7
8 8 8 8 8 8	9 9 9 9 9 9	A A A A A A	B B B B B B	C C C C C C	D D D D D D	E E E E E E	F F F F F F
G G G G G G	H H H H H H	I I I I I I	J J J J J J	K K K K K K	L L L L L L	M M M M M M	0 1 G C 4 5
1 2 H D 5 6	2 3 I E 6 7	3 4 J F 7 8	4 5 K G 8 9	5 6 L H 9 A	6 7 M I A B	7 8 0 J B C	8 9 1 K C D
9 A 2 L D E	A B 3 M E F	B C 4 0 F G	C D 5 1 G H	D E 6 2 H I	E F 7 3 I J	F G 8 4 J K	G H 9 5 K L
H I A 6 L M	I J B 7 M 0	J K C 8 0 1	K L D 9 1 2	L M E A 2 3	M 0 F B 3 4	0 2 9 1 8 A	1 3 A 2 9 B
2 4 B 3 A C	3 5 C 4 B D	4 6 D 5 C E	5 7 E 6 D F	6 8 F 7 E G	7 9 G 8 F H	8 A H 9 G I	9 B I A H J
A C J B I K	B D K C J L	C E L D K M	D F M E L 0	E G 0 F M 1	F H 1 G 0 2	G I 2 H 1 3	H J 3 I 2 4
I K 4 J 3 5	J L 5 K 4 6	K M 6 L 5 7	L 0 7 M 6 8	M 1 8 0 7 9	0 3 2 D C F	1 4 3 E D G	2 5 4 F E H
3 6 5 G F I	4 7 6 H G J	5 8 7 I H K	6 9 8 J I L	7 A 9 K J M	8 B A L K 0	9 C B M L 1	A D C 0 M 2
B E D 1 0 3	C F E 2 1 4	D G F 3 2 5	E H G 4 3 6	F I H 5 4 7	G J I 6 5 8	H K J 7 6 9	I L K 8 7 A
J M L 9 8 B	K 0 M A 9 C	L 1 0 B A D	M 2 1 C B E	0 4 I 2 G K	1 5 J 3 H L	2 6 K 4 I M	3 7 L 5 J 0
4 8 M 6 K 1	5 9 0 7 L 2	6 A 1 8 M 3	7 B 2 9 0 4	8 C 3 A 1 5	9 D 4 B 2 6	A E 5 C 3 7	B F 6 D 4 8
C G 7 E 5 9	D H 8 F 6 A	E I 9 G 7 B	F J A H 8 C	G K B I 9 D	H L C J A E	I M D K B F	J 0 E L C G
K 1 F M D H	L 2 G 0 E I	M 3 H 1 F J	0 5 B E K 2	1 6 C F L 3	2 7 D G M 4	3 8 E H 0 5	4 9 F I 1 6
5 A G J 2 7	6 B H K 3 8	7 C I L 4 9	8 D J M 5 A	9 E K 0 6 B	A F L 1 7 C	B G M 2 8 D	C H 0 3 9 E
D I 1 4 A F	E J 2 5 B G	F K 3 6 C H	G L 4 7 D I	H M 5 8 E J	I 0 6 9 F K	J 1 7 A G L	K 2 8 B H M
L 3 9 C I 0	M 4 A D J 1	0 6 4 3 1 7	1 7 5 4 2 8	2 8 6 5 3 9	3 9 7 6 4 A	4 A 8 7 5 B	5 B 9 8 6 C
6 C A 9 7 D	7 D B A 8 E	8 E C B 9 F	9 F D C A G	A G E D B H	B H F E C I	C I G F D J	D J H G E K
E K I H F L	F L J I G M	G M K J H 0	H 0 L K I 1	I 1 M L J 2	J 2 0 M K 3	K 3 1 0 L 4	L 4 2 1 M 5
M 5 3 2 0 6	0 7 K F 5 C	1 8 L G 6 D	2 9 M H 7 E	3 A 0 I 8 F	4 B 1 J 9 G	5 C 2 K A H	6 D 3 L B I
7 E 4 M C J	8 F 5 0 D K	9 G 6 1 E L	A H 7 2 F M	B I 8 3 G 0	C J 9 4 H 1	D K A 5 I 2	E L B 6 J 3
F M C 7 K 4	G 0 D 8 L 5	H 1 E 9 M 6	I 2 F A 0 7	J 3 G B 1 8	K 4 H C 2 9	L 5 I D 3 A	M 6 J E 4 B
0 8 D 4 9 H	1 9 E 5 A I	2 A F 6 B J	3 B G 7 C K	4 C H 8 D L	5 D I 9 E M	6 E J A F 0	7 F K B G 1
8 G L C H 2	9 H M D I 3	A I 0 E J 4	B J 1 F K 5	C K 2 G L 6	D L 3 H M 7	E M 4 I 0 8	F 0 5 J 1 9
G 1 6 K 2 A	H 2 7 L 3 B	I 3 8 M 4 C	J 4 9 0 5 D	K 5 A 1 6 E	L 6 B 2 7 F	M 7 C 3 8 G	0 9 6 G D M
1 A 7 H E 0	2 B 8 I F 1	3 C 9 J G 2	4 D A K H 3	5 E B L I 4	6 F C M J 5	7 G D 0 K 6	8 H E 1 L 7
9 I F 2 M 8	A J G 3 0 9	B K H 4 1 A	C L I 5 2 B	D M J 6 3 C	E 0 K 7 4 D	F 1 L 8 5 E	G 2 M 9 6 F
H 3 0 A 7 G	I 4 1 B 8 H	J 5 2 C 9 I	K 6 3 D A J	L 7 4 E B K	M 8 5 F C L	0 A M 5 H 4	1 B 0 6 I 5
2 C 1 7 J 6	3 D 2 8 K 7	4 E 3 9 L 8	5 F 4 A M 9	6 G 5 B 0 A	7 H 6 C 1 B	8 I 7 D 2 C	9 J 8 E 3 D
A K 9 F 4 E	B L A G 5 F	C M B H 6 G	D 0 C I 7 H	E 1 D J 8 I	F 2 E K 9 J	G 3 F L A K	H 4 G M B L
I 5 H 0 C M	J 6 I 1 D 0	K 7 J 2 E 1	L 8 K 3 F 2	M 9 L 4 G 3	0 B F H L 9	1 C G I M A	2 D H J 0 B
3 E I K 1 C	4 F J L 2 D	5 G K M 3 E	6 H L 0 4 F	7 I M 1 5 G	8 J 0 2 6 H	9 K 1 3 7 I	A L 2 4 8 J
B M 3 5 9 K	C 0 4 6 A L	D 1 5 7 B M	E 2 6 8 C 0	F 3 7 9 D 1	G 4 8 A E 2	H 5 9 B F 3	I 6 A C G 4
J 7 B D H 5	K 8 C E I 6	L 9 D F J 7	M A E G K 8	0 C 8 6 2 E	1 D 9 7 3 F	2 E A 8 4 G	3 F B 9 5 H
4 G C A 6 I	5 H D B 7 J	6 I E C 8 K	7 J F D 9 L	8 K G E A M	9 L H F B 0	A M I G C 1	B 0 J H D 2
C 1 K I E 3	D 2 L J F 4	E 3 M K G 5	F 4 0 L H 6	G 5 1 M I 7	H 6 2 0 J 8	I 7 3 1 K 9	J 8 4 2 L A
K 9 5 3 M B	L A 6 4 0 C	M B 7 5 1 D	0 D 1 I 6 J	1 E 2 J 7 K	2 F 3 K 8 L	3 G 4 L 9 M	4 H 5 M A 0
5 I 6 0 B 1	6 J 7 1 C 2	7 K 8 2 D 3	8 L 9 3 E 4	9 M A 4 F 5	A 0 B 5 G 6	B 1 C 6 H 7	C 2 D 7 I 8
D 3 E 8 J 9	E 4 F 9 K A	F 5 G A L B	G 6 H B M C	H 7 I C 0 D	I 8 J D 1 E	J 9 K E 2 F	K A L F 3 G
L B M G 4 H	M C 0 H 5 I	0 E H 7 A 1	1 F I 8 B 2	2 G J 9 C 3	3 H K A D 4	4 I L B E 5	5 J M C F 6
6 K 0 D G 7	7 L 1 E H 8	8 M 2 F I 9	9 0 3 G J A	A 1 4 H K B	B 2 5 I L C	C 3 6 J M D	D 4 7 K 0 E
E 5 8 L 1 F	F 6 9 M 2 G	G 7 A 0 3 H	H 8 B 1 4 I	I 9 C 2 5 J	J A D 3 6 K	K B E 4 7 L	L C F 5 8 M
M D G 6 9 0	0 F A J E 6	1 G B K F 7	2 H C L G 8	3 I D M H 9	4 J E 0 I A	5 K F 1 J B	6 L G 2 K C
7 M H 3 L D	8 0 I 4 M E	9 1 J 5 0 F	A 2 K 6 1 G	B 3 L 7 2 H	C 4 M 8 3 I	D 5 0 9 4 J	E 6 1 A 5 K
F 7 2 B 6 L	G 8 3 C 7 M	H 9 4 D 8 0	I A 5 E 9 1	J B 6 F A 2	K C 7 G B 3	L D 8 H C 4	M E 9 I D 5
0 G 3 8 I B	1 H 4 9 J C	2 I 5 A K D	3 J 6 B L E	4 K 7 C M F	5 L 8 D 0 G	6 M 9 E 1 H	7 0 A F 2 I
8 1 B G 3 J	9 2 C H 4 K	A 3 D I 5 L	B 4 E J 6 M	C 5 F K 7 0	D 6 G L 8 1	E 7 H M 9 2	F 8 I 0 A 3
G 9 J 1 B 4	H A K 2 C 5	I B L 3 D 6	J C M 4 E 7	K D 0 5 F 8	L E 1 6 G 9	M F 2 7 H A	0 H J K M G
1 I K L 0 H	2 J L M 1 I	3 K M 0 2 J	4 L 0 1 3 K	5 M 1 2 4 L	6 0 2 3 5 M	7 1 3 4 6 0	8 2 4 5 7 1
9 3 5 6 8 2	A 4 6 7 9 3	B 5 7 8 A 4	C 6 8 9 B 5	D 7 9 A C 6	E 8 A B D 7	F 9 B C E 8	G A C D F 9
H B D E G A	I C E F H B	J D F G I C	K E G H J D	L F H I K E	M G I J L F	0 I C 9 3 L	1 J D A 4 M
2 K E B 5 0	3 L F C 6 1	4 M G D 7 2	5 0 H E 8 3	6 1 I F 9 4	7 2 J G A 5	8 3 K H B 6	9 4 L I C 7
A 5 M J D 8	B 6 0 K E 9	C 7 1 L F A	D 8 2 M G B	E 9 3 0 H C	F A 4 1 I D	G B 5 2 J E	H C 6 3 K F
I D 7 4 L G	J E 8 5 M H	K F 9 6 0 I	L G A 7 1 J	M H B 8 2 K	0 J 5 L 7 3	1 K 6 M 8 4	2 L 7 0 9 5
3 M 8 1 A 6	4 0 9 2 B 7	5 1 A 3 C 8	6 2 B 4 D 9	7 3 C 5 E A	8 4 D 6 F B	9 5 E 7 G C	A 6 F 8 H D
B 7 G 9 I E	C 8 H A J F	D 9 I B K G	E A J C L H	F B K D M I	G C L E 0 J	H D M F 1 K	I E 0 G 2 L
J F 1 H 3 M	K G 2 I 4 0	L H 3 J 5 1	M I 4 K 6 2	0 K L A B 8	1 L M B C 9	2 M 0 C D A	3 0 1 D E B
4 1 2 E F C	5 2 3 F G D	6 3 4 G H E	7 4 5 H I F	8 5 6 I J G	9 6 7 J K H	A 7 8 K L I	B 8 9 L M J
C 9 A M 0 K	D A B 0 1 L	E B C 1 2 M	F C D 2 3 0	G D E 3 4 1	H E F 4 5 2	I F G 5 6 3	J G H 6 7 4
K H I 7 8 5	L I J 8 9 6	M J K 9 A 7	0 L E M F D	1 M F 0 G E	2 0 G 1 H F	3 1 H 2 I G	4 2 I 3 J H
5 3 J 4 K I	6 4 K 5 L J	7 5 L 6 M K	8 6 M 7 0 L	9 7 0 8 1 M	A 8 1 9 2 0	B 9 2 A 3 1	C A 3 B 4 2
D B 4 C 5 3	E C 5 D 6 4	F D 6 E 7 5	G E 7 F 8 6	H F 8 G 9 7	I G 9 H A 8	J H A I B 9	K I B J C A
L J C K D B	M K D L E C	0 M 7 B J I	1 0 8 C K J	2 1 9 D L K	3 2 A E M L	4 3 B F 0 M	5 4 C G 1 0
6 5 D H 2 1	7 6 E I 3 2	8 7 F J 4 3	9 8 G K 5 4	A 9 H L 6 5	B A I M 7 6	C B J 0 8 7	D C K 1 9 8
E D L 2 A 9	F E M 3 B A	G F 0 4 C B	H G 1 5 D C	I H 2 6 E D	J I 3 7 F E	K J 4 8 G F	L K 5 9 H G
M L 6 A I H							

length cannot become smaller if k is increased for fixed q and ℓ. Having identified optimal $k = 2$ codes it will be useful to check the deletion correcting capability of those codes obtained with the same parameters other than higher values of k.

It would also be useful to extend the results of [16] for RS codes fields of prime characteristic, in particular with prime characteristic 2. Codes over such fields have been previously found to have more practical application than those over prime fields.

VI. A Note on the Experimental Results

The tables in this correspondence do not present all our selective and exhaustive results. Experimental results will occasionally be updated at http://www.uow.edu.au/~lukemc/expt.html.

Appendix
Listings of Inequivalence Sets

In this Appendix, we list the inequivalence sets of Section IV-B with cardinalities of length than about 40. The codes are identified by the standard form selectors, which are listed without the first two elements since they are always 0 and 1.

In general a selector $\boldsymbol{\alpha}$ is related to a selector $\boldsymbol{\alpha}'$ under the combined reversal and affine transformation. If $\boldsymbol{\alpha}$ and $\boldsymbol{\alpha}'$ are equal, we mark the selector with a $*$. Otherwise, we list only the smaller of $\boldsymbol{\alpha}$ and $\boldsymbol{\alpha}'$, in the sense of $\boldsymbol{\alpha}$ being smaller than $\boldsymbol{\alpha}'$ if $\alpha_j < \alpha'_j$ for some j such that $\alpha_i = \alpha'_i, \forall\, i < j$.

$$\mathbb{P}[5,4]_0 = \{(2\;3)*,(2\;4),(3\;2),(3\;4)*\}.$$
$$\mathbb{P}[5,5]_0 = \{(2\;3\;4)*,(2\;4\;3),(3\;2\;4)\}.$$
$$\mathbb{P}[5,5]_1 = \{(4\;2\;3)*\}.$$
$$\mathbb{P}[7,4]_0 = \{(2\;3)*,(2\;4),(3\;2),(3\;6),(4\;3),(4\;6)\}.$$
$$\mathbb{P}[7,4]_1 = \{(2\;5),(2\;6),(3\;4)*,(4\;5)*,(5\;3),(5\;6)*\}.$$
$$\mathbb{P}[7,5]_0 = \{(2\;3\;4)*,(3\;2\;6),(4\;6\;3),(4\;6\;5)\}.$$

$$\begin{aligned}\mathbb{P}[7,5]_1 = \{&(2\;3\;5),(2\;3\;6),(2\;4\;3),(2\;4\;5),(2\;4\;6),(2\;5\;3)\\ &(2\;5\;4),(2\;5\;6),(2\;6\;4),(2\;6\;5),(3\;2\;4),(3\;2\;5)\\ &(3\;4\;2),(3\;4\;6),(3\;5\;2),(3\;5\;4),(3\;5\;6)*,(3\;6\;5)\\ &(4\;2\;3),(4\;2\;5),(4\;2\;6),(4\;5\;3),(5\;2\;3)*,(5\;3\;4)\\ &(5\;3\;6),(5\;6\;2),(6\;2\;4),(6\;4\;5)*\}.\end{aligned}$$
$$\mathbb{P}[7,6]_0 = \{(2\;3\;4\;5)*,(3\;2\;6\;4),(4\;6\;5\;2),(4\;6\;5\;3)\}.$$
$$\mathbb{P}[7,6]_2 = \{(6\;5\;3\;4)*\}.$$
$$\mathbb{P}[7,7]_0 = \{(2\;3\;4\;5\;6)*,(3\;2\;6\;4\;5),(4\;6\;5\;2\;3)\}.$$

$$\begin{aligned}\mathbb{P}[7,7]_2 = \{&(2\;3\;5\;6\;4),(2\;5\;4\;6\;3),(2\;5\;6\;3\;4),(2\;6\;5\;3\;4)\\ &(3\;4\;5\;6\;2),(3\;5\;6\;2\;4),(4\;3\;2\;6\;5),(4\;3\;5\;6\;2)\\ &(4\;5\;2\;3\;6),(4\;5\;3\;2\;6),(5\;3\;6\;2\;4),(6\;2\;5\;3\;4)*\}.\end{aligned}$$

$$\begin{aligned}\mathbb{P}[11,4]_0 = \{&(2\;3)*,(2\;4),(3\;7),(3\;9),(3\;10),(4\;2),(4\;5)*\\ &(5\;3),(6\;8),(6\;9),(7\;5),(8\;2),(8\;9)*\}.\end{aligned}$$

$$\begin{aligned}\mathbb{P}[11,4]_1 = \{&(2\;5),(2\;6),(2\;7),(2\;8),(2\;9),(2\;10),(3\;2)\\ &(3\;4)*,(3\;6),(3\;8),(4\;3),(4\;6),(4\;8),(4\;9)\\ &(4\;10),(5\;2),(5\;4),(5\;6)*,(5\;8),(6\;2),(6\;5)\\ &(6\;7)*,(7\;4),(7\;8)*,(8\;10),(9\;5),(9\;10)*\}.\end{aligned}$$

$$\begin{aligned}\mathbb{P}[11,5]_0 = \{&(2\;3\;4)*,(2\;4\;8),(3\;7\;4),(3\;7\;10),(3\;9\;5)\\ &(4\;2\;5),(4\;2\;7),(5\;3\;4),(6\;9\;2),(6\;9\;8)\\ &(7\;5\;2),(8\;2\;4),(8\;2\;9)\}.\end{aligned}$$
$$\begin{aligned}\mathbb{P}[11,6]_0 = \{&(2\;3\;4\;5)*,(2\;4\;8\;5),(3\;7\;4\;9),(3\;7\;4\;10)\\ &(3\;9\;5\;4),(4\;2\;7\;5),(5\;3\;4\;9),(6\;9\;2\;8)\\ &(7\;5\;2\;3),(8\;2\;4\;7),(8\;2\;4\;9)\}.\end{aligned}$$

$$\begin{aligned}\mathbb{P}[11,7]_0 = \{&(2\;3\;4\;5\;6)*,(2\;4\;8\;5\;10),(3\;7\;4\;9\;8)\\ &(3\;7\;4\;9\;10).(7\;5\;2\;3\;10),(8\;2\;4\;7\;6)\\ &(8\;2\;4\;7\;9)\}.\end{aligned}$$

$$\mathbb{P}[11,7]_3 = \{(2\;8\;10\;3\;5),(7\;8\;4\;3\;9)\}.$$

$$\begin{aligned}\mathbb{P}[11,8]_0 = \{&(2\;3\;4\;5\;6\;7)*,(2\;4\;8\;5\;10\;9),(3\;7\;4\;9\;8\;6)\\ &(3\;7\;4\;9\;8\;10),(7\;5\;2\;3\;10\;4),(8\;2\;4\;7\;6\;9)\\ &(8\;2\;4\;7\;6\;10)\}.\end{aligned}$$

$$\begin{aligned}\mathbb{P}[11,9]_0 = \{&(2\;3\;4\;5\;6\;7\;8)*,(2\;4\;8\;5\;10\;9\;7),(3\;7\;4\;9\;8\;6\;2)\\ &(3\;7\;4\;9\;8\;6\;10),(7\;5\;2\;3\;10\;4\;6),(8\;2\;4\;7\;6\;10\;5)\\ &(8\;2\;4\;7\;6\;10\;9)\}.\end{aligned}$$

$$\mathbb{P}[11,9]_4 = \{(2\;10\;4\;6\;5\;8\;9),(2\;10\;8\;5\;6\;4\;9),(6\;7\;3\;2\;8\;9\;5)\}.$$

$$\begin{aligned}\mathbb{P}[11,10]_0 = \{&(2\;3\;4\;5\;6\;7\;8\;9)*,(2\;4\;8\;5\;10\;9\;7\;3)\\ &(3\;7\;4\;9\;8\;6\;2\;5),(3\;7\;4\;9\;8\;6\;2\;10)\\ &(7\;5\;2\;3\;10\;4\;6\;9),(8\;2\;4\;7\;6\;10\;5\;3)\\ &(8\;2\;4\;7\;6\;10\;5\;9)\}.\end{aligned}$$

$$\begin{aligned}\mathbb{P}[11,11]_0 = \{&(2\;3\;4\;5\;6\;7\;8\;9\;10)*,(2\;4\;8\;5\;10\;9\;7\;3\;6)\\ &(3\;7\;4\;9\;8\;6\;2\;5\;10),(7\;5\;2\;3\;10\;4\;6\;9\;8)\\ &(8\;2\;4\;7\;6\;10\;5\;3\;9)\}.\end{aligned}$$

$$\begin{aligned}\mathbb{P}[13,4]_0 = \{&(2\;3)*,(2\;4),(3\;7),(3\;9),(3\;12),(4\;3),(4\;6),(5\;4)\\ &(5\;8),(5\;12),(6\;3),(6\;5),(6\;10),(7\;4),(7\;5)\}.\end{aligned}$$

$$\begin{aligned}\mathbb{P}[13,5]_0 = \{&(2\;3\;4)*,(2\;4\;8),(3\;7\;2),(3\;7\;12),(4\;3\;12)\\ &(5\;8\;4),(5\;8\;7),(5\;12\;8),(6\;5\;3),(6\;10\;8)\\ &(7\;4\;5),(7\;4\;12)\}.\end{aligned}$$

$$\mathbb{P}[13,5]_2 = \{(7\;6\;2)\}.$$

$$\begin{aligned}\mathbb{P}[13,6]_0 = \{&(2\;3\;4\;5)*,(2\;4\;8\;3),(3\;7\;2\;5),(3\;7\;2\;12)\\ &(4\;3\;12\;9),(5\;8\;7\;3),(5\;8\;7\;4),(6\;10\;8\;9)\\ &(7\;4\;12\;5),(7\;4\;12\;8)\}.\end{aligned}$$

$$\begin{aligned}\mathbb{P}[13,7]_0 = \{&(2\;3\;4\;5\;6)*,(2\;4\;8\;3\;6),(3\;7\;2\;5\;11)\\ &(3\;7\;2\;5\;12),(4\;3\;12\;9\;10),(5\;8\;7\;3\;4)\\ &(6\;10\;8\;9\;2),(7\;4\;12\;8\;5),(7\;4\;12\;8\;10)\}.\end{aligned}$$

$$\begin{aligned}\mathbb{P}[13,8]_0 = \{&(2\;3\;4\;5\;6\;7)*,(2\;4\;8\;3\;6\;12),(3\;7\;2\;5\;11\;10)\\ &(3\;7\;2\;5\;11\;12),(6\;10\;8\;9\;2\;12),(7\;4\;12\;8\;10\;5)\\ &(7\;4\;12\;8\;10\;9)\}.\end{aligned}$$

$$\begin{aligned}\mathbb{P}[13,9]_0 = \{&(2\;3\;4\;5\;6\;7\;8)*,(2\;4\;8\;3\;6\;12\;11)\\ &(3\;7\;2\;5\;11\;10\;8),(3\;7\;2\;5\;11\;10\;12)\\ &(6\;10\;8\;9\;2\;12\;7),(7\;4\;12\;8\;10\;9\;3)\\ &(7\;4\;12\;8\;10\;9\;5)\}.\end{aligned}$$

$$\begin{aligned}\mathbb{P}[17,4]_0 = \{&(2\;3)*,(2\;4),(3\;7),(3\;9),(3\;16),(4\;8),(4\;13)\\ &(4\;16),(5\;4),(5\;8),(5\;11),(6\;4),(6\;14),(8\;13)\end{aligned}$$

$(9\ 5), (9\ 12), (10\ 15), (11\ 2), (11\ 9), (11\ 15)$
$(12\ 5), (12\ 14)\}.$

$\mathbb{P}[17,5]_0 = \{(2\ 3\ 4)*, (2\ 4\ 8), (3\ 7\ 15), (3\ 7\ 16), (3\ 9\ 10)$
$(4\ 13\ 6), (4\ 13\ 8), (4\ 16\ 13), (5\ 4\ 11), (5\ 8\ 6)$
$(6\ 14\ 3), (6\ 14\ 4), (8\ 13\ 2), (9\ 5\ 7), (9\ 5\ 12)$
$(10\ 15\ 14), (11\ 2\ 5), (11\ 9\ 6), (11\ 9\ 15), (12\ 14\ 2)$
$(12\ 14\ 5)\}.$

$\mathbb{P}[17,6]_0 = \{(2\ 3\ 4\ 5)*, (2\ 4\ 8\ 16), (3\ 7\ 15\ 14), (3\ 7\ 15\ 16)$
$(3\ 9\ 10\ 13), (4\ 13\ 6\ 2), (4\ 13\ 6\ 8), (5\ 8\ 6\ 13)$
$(6\ 14\ 3\ 4), (6\ 14\ 3\ 16), (8\ 13\ 2\ 16), (9\ 5\ 7\ 6)$
$(9\ 5\ 7\ 12), (10\ 15\ 14\ 4), (11\ 2\ 5\ 4), (11\ 9\ 6\ 10)$
$(11\ 9\ 6\ 15), (12\ 14\ 2\ 5), (12\ 14\ 2\ 6)\}.$

$\mathbb{P}[17,7]_0 = \{(2\ 3\ 4\ 5\ 6)*, (2\ 4\ 8\ 16\ 15), (3\ 7\ 15\ 14\ 12)$
$(3\ 7\ 15\ 14\ 16), (3\ 9\ 10\ 13\ 5), (4\ 13\ 6\ 2\ 7)$
$(4\ 13\ 6\ 2\ 8), (5\ 8\ 6\ 13\ 14), (6\ 14\ 3\ 16\ 4)$
$(6\ 14\ 3\ 16\ 13), (8\ 13\ 2\ 16\ 9), (9\ 5\ 7\ 6\ 12)$
$(9\ 5\ 7\ 6\ 15), (10\ 15\ 14\ 4\ 6), (11\ 2\ 5\ 4\ 10)$
$(11\ 9\ 6\ 10\ 15), (11\ 9\ 6\ 10\ 16), (12\ 14\ 2\ 6\ 5)$
$(12\ 14\ 2\ 6\ 16)\}.$

$\mathbb{P}[19,4]_0 = \{(2\ 3)*, (2\ 4), (3\ 7), (3\ 9), (3\ 18), (4\ 9), (4\ 13)$
$(4\ 16), (5\ 2), (5\ 6)*, (6\ 17), (7\ 5), (7\ 11), (7\ 15)$
$(8\ 3), (8\ 7), (9\ 5), (9\ 8), (9\ 16), (10\ 7), (10\ 15)$
$(14\ 6), (15\ 2), (15\ 16)*\}.$

$\mathbb{P}[19,5]_0 = \{(2\ 3\ 4)*, (2\ 4\ 8), (3\ 7\ 15), (3\ 7\ 18), (3\ 9\ 8)$
$(4\ 13\ 2), (4\ 13\ 9), (4\ 16\ 7), (5\ 2\ 6), (5\ 2\ 9)$
$(6\ 17\ 7), (7\ 5\ 12), (7\ 5\ 15), (8\ 7\ 18), (9\ 5\ 7)$
$(9\ 16\ 8), (9\ 16\ 15), (10\ 15\ 3), (10\ 15\ 7), (14\ 6\ 8)$
$(15\ 2\ 10), (15\ 2\ 16)\}.$

$\mathbb{P}[19,7]_0 = \{(2\ 3\ 4\ 5\ 6)*, (2\ 4\ 8\ 16\ 13), (3\ 7\ 15\ 12\ 6)$
$(3\ 7\ 15\ 12\ 18), (3\ 9\ 8\ 5\ 15), (4\ 13\ 2\ 7\ 3)$
$(4\ 13\ 2\ 7\ 9), (4\ 16\ 7\ 9\ 17), (5\ 2\ 9\ 18\ 6)$
$(5\ 2\ 9\ 18\ 16), (6\ 17\ 7\ 4\ 5), (7\ 5\ 12\ 16\ 2)$
$(7\ 5\ 12\ 16\ 15), (8\ 7\ 18\ 11\ 12), (9\ 5\ 7\ 6\ 16)$
$(9\ 16\ 15\ 7\ 8), (10\ 15\ 3\ 9\ 6), (10\ 15\ 3\ 9\ 7)$
$(14\ 6\ 8\ 17\ 10), (15\ 2\ 10\ 8\ 16), (15\ 2\ 10\ 8\ 18)\}.$

$\mathbb{P}[23,4]_0 = \{(2\ 3)*, (2\ 4), (3\ 7), (3\ 9), (3\ 22), (4\ 11), (4\ 13)$
$(4\ 16), (5\ 2), (5\ 15), (5\ 21), (6\ 8), (6\ 17), (7\ 3)$
$(8\ 11), (8\ 19), (9\ 12), (10\ 20), (10\ 22), (11\ 6)$
$(12\ 16), (12\ 18), (13\ 8), (14\ 21), (14\ 22), (15\ 18)$
$(16\ 11), (16\ 18), (17\ 13), (18\ 8), (18\ 10)\}.$

$\mathbb{P}[23,5]_0 = \{(2\ 3\ 4)*, (2\ 4\ 8), (3\ 7\ 15), (3\ 7\ 22), (3\ 9\ 4)$
$(4\ 13\ 11), (4\ 13\ 17), (4\ 16\ 18), (5\ 2\ 10), (5\ 21\ 15)$
$(5\ 21\ 16), (6\ 8\ 17), (6\ 8\ 18), (7\ 3\ 21), (8\ 11\ 9)$
$(8\ 11\ 19), (9\ 12\ 16), (10\ 22\ 15), (10\ 22\ 20)$
$(11\ 6\ 20), (12\ 18\ 15), (12\ 18\ 16), (13\ 8\ 12)$
$(14\ 22\ 11), (14\ 22\ 21), (15\ 18\ 17), (16\ 11\ 5)$
$(16\ 11\ 18), (17\ 13\ 14), (18\ 8\ 10), (18\ 8\ 22)\}.$

$\mathbb{P}[23,6]_0 = \{(2\ 3\ 4\ 5)*, (2\ 4\ 8\ 16), (3\ 7\ 15\ 8), (3\ 7\ 15\ 22)$
$(3\ 9\ 4\ 12), (4\ 13\ 17\ 6), (4\ 13\ 17\ 11), (4\ 16\ 18\ 3)$
$(5\ 2\ 10\ 4), (5\ 21\ 16\ 15), (5\ 21\ 16\ 19), (6\ 8\ 18\ 17)$
$(6\ 8\ 18\ 22), (7\ 3\ 21\ 9), (8\ 11\ 9\ 18), (8\ 11\ 9\ 19)$
$(9\ 12\ 16\ 6), (10\ 22\ 15\ 20), (10\ 22\ 15\ 21)$
$(11\ 6\ 20\ 13), (12\ 18\ 15\ 5), (12\ 18\ 15\ 16)$
$(13\ 8\ 12\ 18), (14\ 22\ 11\ 6), (14\ 22\ 11\ 21)$
$(15\ 18\ 17\ 2), (16\ 11\ 5\ 7), (16\ 11\ 5\ 18)$
$(17\ 13\ 14\ 8), (18\ 8\ 22\ 7), (18\ 8\ 22\ 10)\}.$

$\mathbb{P}[23,6]_3 = \{(16\ 12\ 4\ 5)*\}.$

$\mathbb{P}[23,7]_0 = \{(2\ 3\ 4\ 5\ 6)*, (2\ 4\ 8\ 16\ 9), (3\ 7\ 15\ 8\ 17)$
$(3\ 7\ 15\ 8\ 22), (3\ 9\ 4\ 12\ 13), (4\ 13\ 17\ 6\ 11)$
$(4\ 13\ 17\ 6\ 19), (4\ 16\ 18\ 3\ 12), (5\ 2\ 10\ 4\ 20)$
$(5\ 21\ 16\ 19\ 8), (5\ 21\ 16\ 19\ 15), (6\ 8\ 18\ 22\ 17)$
$(6\ 8\ 18\ 22\ 19), (7\ 3\ 21\ 9\ 17), (8\ 11\ 9\ 18\ 12)$
$(8\ 11\ 9\ 18\ 19), (9\ 12\ 16\ 6\ 8), (10\ 22\ 15\ 21\ 6)$
$(10\ 22\ 15\ 21\ 20), (11\ 6\ 20\ 13\ 5), (12\ 18\ 15\ 5\ 10)$
$(12\ 18\ 15\ 5\ 16), (13\ 8\ 12\ 18\ 4), (14\ 22\ 11\ 6\ 10)$
$(14\ 22\ 11\ 6\ 21), (15\ 18\ 17\ 2\ 7), (16\ 11\ 5\ 7\ 14)$
$(16\ 11\ 5\ 7\ 18), (17\ 13\ 14\ 8\ 21), (18\ 8\ 22\ 7\ 5)$
$(18\ 8\ 22\ 7\ 10)\}.$

$\mathbb{P}[29,4]_0 = \{(2\ 3)*, (2\ 4), (3\ 7), (3\ 9), (3\ 28), (4\ 13), (4\ 14)$
$(4\ 16), (5\ 19), (5\ 21), (5\ 25), (6\ 2), (6\ 7)*, (7\ 20)$
$(8\ 6), (8\ 24), (8\ 28), (9\ 4), (9\ 15), (9\ 23), (10\ 4)$
$(10\ 18), (12\ 28), (13\ 12), (13\ 21), (14\ 22), (15\ 8)$
$(15\ 20), (16\ 24), (17\ 12), (17\ 27), (18\ 5), (19\ 13)$
$(19\ 17), (19\ 24), (20\ 4), (20\ 8), (23\ 7), (24\ 2)$
$(24\ 25)*\}.$

$\mathbb{P}[29,5]_0 = \{(2\ 3\ 4)*, (2\ 4\ 8), (3\ 7\ 15), (3\ 7\ 28), (3\ 9\ 27)$
$(4\ 13\ 11), (4\ 13\ 14), (4\ 16\ 6), (5\ 21\ 19), (5\ 21\ 27)$
$(5\ 25\ 9), (6\ 2\ 7), (6\ 2\ 11), (7\ 20\ 24), (8\ 6\ 19)$
$(8\ 28\ 23), (8\ 28\ 24), (9\ 15\ 4), (9\ 15\ 5), (9\ 23\ 4)$
$(10\ 4\ 8), (10\ 4\ 18), (12\ 28\ 17), (13\ 12\ 21)$
$(14\ 22\ 18), (15\ 8\ 20), (15\ 8\ 26), (16\ 24\ 7)$
$(17\ 12\ 19), (17\ 12\ 27), (18\ 5\ 3), (19\ 13\ 15)$
$(19\ 24\ 17), (19\ 24\ 27), (20\ 4\ 8), (20\ 4\ 19)$
$(23\ 7\ 16), (24\ 2\ 18), (24\ 2\ 25)\}.$

$\mathbb{P}[29,6]_0 = \{(2\ 3\ 4\ 5)*, (2\ 4\ 8\ 16), (3\ 7\ 15\ 2), (3\ 7\ 15\ 28)$
$(3\ 9\ 27\ 23), (4\ 13\ 11\ 5), (4\ 13\ 11\ 14), (4\ 16\ 6\ 24)$
$(5\ 21\ 27\ 19), (5\ 21\ 27\ 22), (5\ 25\ 9\ 16), (6\ 2\ 11\ 7)$
$(6\ 2\ 11\ 27), (7\ 20\ 24\ 23), (8\ 6\ 19\ 7), (8\ 28\ 23\ 17)$
$(8\ 28\ 23\ 24), (9\ 15\ 5\ 4), (9\ 15\ 5\ 12), (9\ 23\ 4\ 7)$

$$(10\ 4\ 8\ 15), (10\ 4\ 8\ 18), (14\ 22\ 18\ 20)$$
$$(15\ 8\ 26\ 17), (15\ 8\ 26\ 20), (16\ 24\ 7\ 25)$$
$$(17\ 12\ 19\ 15), (17\ 12\ 19\ 27), (18\ 5\ 3\ 25)$$
$$(19\ 13\ 15\ 24), (19\ 24\ 27\ 17), (19\ 24\ 27\ 23)$$
$$(20\ 4\ 19\ 8), (20\ 4\ 19\ 14), (23\ 7\ 16\ 20)$$
$$(24\ 2\ 18\ 9), (24\ 2\ 18\ 25)\}.$$

$$\mathbb{P}[47,7]_4 = \{(8\ 23\ 42\ 16\ 18), (15\ 46\ 28\ 14\ 41)$$
$$(16\ 27\ 42\ 26\ 29), (27\ 2\ 40\ 22\ 35)\}.$$

ACKNOWLEDGMENT

The authors appreciate the valuable comments of the thorough reviewers. They have helped us to significantly improve the readability and consistency of this correspondence.

REFERENCES

[1] P. A. H. Bours, "On the construction of perfect deletion—Correcting codes using design theory," *Designs, Codes, Cryptogr.*, vol. 6, pp. 5–20, 1995.

[2] L. Calabi and W. E. Hartnett, "Some general results of coding theory with applications to the study of codes for the correction of synchronisation errors," *Inf. Contr.*, vol. 15, pp. 235–249, 1969.

[3] V. Guruswami and M. Sudan, "Improved decoding of Reed—Solomon and algebraic—Geometry codes," *IEEE Trans. Inf. Theory*, vol. 45, pp. 1757–1767, 1999.

[4] A. Klein, "On perfect deletion-correcting codes," *J. Comb. Des.*, vol. 12, no. 1, pp. 72–77, 2004.

[5] T. Kløve, "Codes correcting a single insertion/deletion of a zero or a single peak-shift," *IEEE Trans. Inf. Theory*, vol. 41, pp. 279–283, 1995.

[6] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions and reversals," *Soviet Phys.—Doklady*, vol. 10, no. 8, pp. 707–710, 1966.

[7] V. I. Levenshtein, "One method of constructing quasilinear codes providing synchronisation in the presence of errors," *Probl. Inf. Transm.*, vol. 7, no. 3, pp. 215–222, 1971.

[8] A. Mahmoodi, "Existence of perfect 3—Deletion-correcting codes," *Designs, Codes, Cryptogr.*, vol. 14, pp. 81–87, 1998.

[9] R. Mathon and T. van Trung, "Directed t—Packings and directed t—Steiner systems," *Designs, Codes, Cryptogr.*, vol. 18, pp. 187–198, 1999.

[10] I. Reed and G. Solomon, "Polynomial codes over certain finite fields," *SIAM J. Appl. Math.*, vol. 8, no. 2, pp. 300–304, 1960.

[11] R. Safavi-Naini and Y. Wang, "Traitor tracing for shortened and corrupted fingerprints," in *Proc. ACM-DRM'02, LNCS*, 2003, vol. 2696, pp. 81–100.

[12] N. Shalaby, J. Wang, and J. Yin, "Existence of perfect 4—Deletion-Correcting codes with length six," *Designs, Codes, Cryptogr.*, vol. 27, pp. 145–156, 2002.

[13] N. J. A. Sloane*, On Single-Deletion—Correcting Codes' in Codes and Designs*. Columbus, OH: Math. Res. Inst. Publications, Ohio Univ., 2002, vol. 10, pp. 273–291.

[14] E. Tanaka and T. Kasai, "Synchronisation and substitution error correcting codes for the Levenshtein metric," *IEEE Trans. Inf. Theory*, vol. 22, pp. 156–162, 1976.

[15] R. R. Varshamov and G. M. Tenengolts, "Codes which correct single asymmetric errors (in Russian)," (in Russian) *Avtomatika i Telemekhanika*, vol. 26, no. 2, pp. 288–292, 1965.

[16] Y. Wang, L. McAven, and R. Safavi-Naini*, Deletion Correcting Using Generalised Reed-Solomon Codes.' t Coding, Cryptography and Combinatorics*, K. Q. Feng, H. Niederreiter, and C. Xing, Eds. Basel: Birkhäuser, 2004.

[17] J. Yin, "A combinatorial construction for perfect deletion-correcting codes," *Designs, Codes and Cryptogr.*, vol. 23, pp. 99–110, 2001.

Bounds on Key Appearance Equivocation for Substitution Ciphers

Yuri L. Borissov and Moon Ho Lee*, Senior Member, IEEE*

***Abstract*—The average conditional entropy of the key given the message and its corresponding cryptogram, $H(\mathbf{K}|\mathbf{M},\mathbf{C})$, which is refer as a key appearance equivocation, was proposed as a theoretical measure of the strength of the cipher system under a known plaintext attack by Dunham in 1980. In the same work (among other things), lower and upper bounds for $H(\mathcal{S}_{\mathcal{M}}|\mathbf{M}^L\mathbf{C}^L)$ are found and its asymptotic behavior as a function of cryptogram length L is described for simple substitution ciphers, i.e., when the key space $\mathcal{S}_{\mathcal{M}}$ is the symmetric group acting on a discrete alphabet $\mathcal{M}$. In the present paper we consider the same problem when the key space is an arbitrary subgroup $\mathcal{K} \triangleleft \mathcal{S}_{\mathcal{M}}$ and generalize Dunham's result.**

***Index Terms*—Key equivocation, known plaintext attack, memoryless message source, message equivocation, simple substitution ciphers.**

I. INTRODUCTION

Shannon, in his seminal paper [2], showed that the conditional entropies of the key and message given the cryptogram can be used as a theoretical measure of strength of the cipher system when assuming unlimited cryptanalytic computational capabilities. These conditional entropies are called the key and message equivocation, respectively.

In general it is difficult to calculate these equivocations explicitly. For that Shannon established in [2] a general lower bound and introduced a random cipher model which would approximate the behavior of complex practical ciphers. Afterward, Hellman [3] reviewed and extended Shannon's information-theoretic approach and showed that random cipher model is conservative in that a randomly chosen cipher is essentially the worst possible. Later on Blom [5] obtained exponentially tight bounds on the key equivocation for simple substitution ciphers. In [1] to derive bounds for simple substitution ciphers on the message equivocation in terms of the key equivocation, Dunham derived such bounds for so-called key appearance equivocation. This author pointed out also, that it can be considered as a theoretical measure of the strength of the cipher system under known plaintext attack. Another contribution of this subject is the Sgarro's work [7].

In this paper we consider a situation where the key space is confined to a subgroup $\mathcal{K}$ of the group $\mathcal{S}_{\mathcal{M}}$ of all permutations acting on a discrete alphabet $\mathcal{M}$. Apart from simple substitution ciphers, some other classical cipher systems (e.g., transposition cipher with fixed period, matrix system from [2, Example 4.6, p. 667], etc.) can be studied in this model. Other examples are given in [4] and [6].

The paper is organized as follows. In Section II, we present the assumptions and background of substitution ciphers and key appearance equivocation. In Section III, we state a theorem which gives the

Manuscript received December 11, 2006; revised February 19, 2007. This work was supported in part by Ministry of Information and Communication (MIC) Korea under the IT Foreign Specialist Inviting Program (ITFSIP), ITSOC, ITRC, International Cooperative Research by the Ministry of Science and Technology, KOTEF, and 2nd stage Brain Korea 21.

Y. L. Borissov is with the Institute of Mathematics and Informatics, Bulgarian Academy of Sciences, Sofia 1113, Bulgaria (e-mail: yborisov@moi.math.bas.bg).

M. H. Lee is with the Institute of Information and Communication, Chonbuk National University, Jeonju 561-756, Republic of Korea (e-mail: moonho@chonbuk.ac.kr).

Communicated by E. Okamoto, Associate Editor for Complexity and Cryptography.

Digital Object Identifier 10.1109/TIT.2007.896865