References

[1] J. C. Bourin, “Some inequalities for norms on matrices and operators,” *Linear Alg. Appl.*, vol. 292, pp. 139–154, 1999.

[2] J. C. Bourin, *Compressions, Dilations and matrix inequalities, RGMIA Monographs*. Victoria, Australia: Victoria University, 2004.

[3] M. V. Burnashev and A. S. Holevo, “On the reliability function for a quantum communication channel,” *Prob. Inf. Trans.*, vol. 34, no. 2, pp. 97–107, 1998.

[4] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York: Wiley, 1991.

[5] J. I. Fujii and M. Fujii, “Jensen’s Inequalities on any interval for operators,” in *Proc. 3rd Int. Conf. Nonlinear Anal. Convex Anal.*, 2004, pp. 29–39.

[6] J. I. Fujii, “A trace inequality arising from quantum information theory,” *Linear Alg. Appl.*, vol. 400, pp. 141–146, 2005.

[7] S. Furuichi, K. Yanagi, and K. Kuriyama, “A sufficient condition on concavity of the auxiliary function appearing in quantum reliability function,” *INFORMATION*, vol. 6, no. 1, pp. 71–76, 2003.

[8] R. G. Gallager, *Information theory and reliable communication*. New York: Wiley, 1968.

[9] F. Hansen and G. K. Pedersen, “Jensen’s operator inequality,” *Bull. London Math. Soc.*, vol. 35, pp. 553–564, 2003.

[10] A. S. Holevo, “The capacity of quantum channel with general signal states,” *IEEE. Trans. Inf. Theory*, vol. 44, no. 1, pp. 269–273, 1998.

[11] A. S. Holevo, “Reliability function of general classical-quantum channel,” *IEEE. Trans. Inf. Theory*, vol. 46, no. 6, pp. 2256–2261, 2000.

[12] T. Ogawa and H. Nagaoka, “Strong converse to the quantum channel coding theorem,” *IEEE. Trans. Inf. Theory*, vol. 45, no. 7, pp. 2486–2489, 1999.

[13] K. Yanagi, S. Furuichi, and K. Kuriyama, “On trace inequalities and their applications to noncommutative communication theory,” *Linear Alg. Appl.*, vol. 395, pp. 351–359, 2005.

On the Number of Encoder States of a Type of RLL Codes

Kui Cai and Kees A. Schouhamer Immink, *Fellow, IEEE*

***Abstract*—The relationship between the number of encoder states and the probable size of certain runlength-limited (RLL) codes is derived analytically. By associating the number of encoder states with (generalized) Fibonacci numbers, the minimum number of encoder states is obtained, which maximizes the rate of the designed code, irrespective of the codeword length.**

***Index Terms*—Fibonacci numbers, finite-state machine (FSM), generalized Fibonacci numbers, runlength-limited (RLL) codes, Shannon capacity.**

I. Introduction

Runlength-limited (RLL) codes, generically designated as (d, k) RLL codes, have been widely and successfully applied in magnetic and optical recording systems. Binary sequences generated by a (d, k) RLL encoder have at least d and at most k, $k > d$, ‘0’s’ between successive ‘1’s.’ Let the integers m and n denote the information word length and codeword length, respectively. The code rate, $R = m/n$, is a measure of the code’s efficiency. The maximum rate of a RLL code for given values of d and k, denoted by $C(d, k)$, is called the Shannon capacity [1].

Finite-state constrained encoders have become very popular in recording practice due to their high code rates [2]. The number of encoder states is key for the design of finite-state constrained codes, since it directly affects the coding efficiency as well as the encoding/decoding complexity. The *approximate eigenvector equation* guides a variety of code constructions, such as the renowned *state-splitting method* [3]. The sum of the components of the *approximate eigenvector* gives a loose upper bound on the number of encoder states, depending on the code constraints and designed rate.

In [4], Immink *et al.* have introduced a new family of efficient finite-state RLL codes with $d = 1$ or $d = 2$ constraints, whose rates are very close to the Shannon capacity. Unlike the *state-splitting method* starting with the *labeled graph*, they [4] propose simple and efficient finite-state machines (FSM), which specify the encoding/decoding principles for $d = 1$ and $d = 2$ codes directly. In [5], a general construction of capacity-approaching constrained parity-check codes has been proposed based on the same FSMs.

For finite-state constrained codes, there is not yet a definite solution on how to determine the minimum number of encoder states to maximize the code rate. In this correspondence, we focus on the necessary conditions for the design of the above capacity-approaching codes. Based on these conditions, a relationship between the number of encoder states and the probable size of the code can be derived. This guides the code design. The valid codewords are then assigned to different encoder states and the encoder and decoder may, thereby, be constructed.

Manuscript received April 18, 2005; revised October 18, 2005. The material in this correspondence was presented in part at the Twenty-Fifth Symposium on Information Theory on the Benelux, Kardrade, The Netherlands, June 2004.

K. Cai is with Data Storage Institute, Singapore, 117608, and National University of Singapore, Singapore, 119260(e-mail: Cai_Kui@dsi.a-star.edu.sg).

K. A. S. Immink is with the Institute for Experimental Mathematics, 45326 Essen, Germany, and Turing Machines Inc., 3016 DK Rotterdam, The Netherlands (e-mail: immink@iem.uni-due.de; kees@immink.nl).

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

Digital Object Identifier 10.1109/TIT.2006.876231

The brute-force way to find the relationship is to use an exhaustive search. Thus, for each desired codeword length, the probable code sizes corresponding to different choices of the number of encoder states need to be searched separately. In this work, we derive the relationship analytically for RLL codes with $d = 1$ constraint, which is used in the third generation optical recording systems (i.e., blu-ray disc (BD) and high-definition digital versatile disc (HD-DVD)) [6], [7]. We further obtain the minimum number of encoder states, which maximizes the probable size of the designed code, for any desired codeword length. These states also help to successfully assign codewords to various encoder states to maximize the code rate. Therefore, our analysis provides direct guidelines for choosing the number of encoder states (or equivalently, splitting the states of the encoder) for high-efficiency RLL codes with $d = 1$ constraint, as well as for other constraints that may be desired for future recording systems, such as the parity-check constraint and the maximum transition run (MTR) constraint [8].

This correspondence is organized as follows. Section II first reviews the techniques proposed in [4] to design capacity-approaching RLL codes with $d = 1$ constraint. The definitions of Fibonacci and generalized Fibonacci numbers, which are used to search the minimum number of encoder states for the desired codes are then given. Section III presents the analysis on the relationship between the number of encoder states and the probable size of the code. The minimum number of encoder states is determined in Section IV. Finally, concluding remarks are given in Section V.

II. Preliminaries

A. Design of Capacity-Approaching RLL Codes

The operation of the finite-state encoder proposed in [4] can be represented by an FSM, which is defined by the input set, the output set, the state set, and two logical functions: The output function and the next-state function. The principle for encoding/decoding $d = 1$ codes can be described as follows.

The input set B consists of m-digit (binary) information words, with a size $|B| = 2^m$. The tth information word in an input sequence is denoted by b_t, where t is an integer, denoting time.

The output set X consists of n-digit (binary) codewords. According to [4], the set of codewords X is divided into four subsets X_{00}, X_{01}, X_{10}, and X_{11}, which are characterized as follows. Codewords in X_{00} start and end with a '0,' codewords in X_{01} start with a '0' and end with a '1,' etc. The tth codeword in an output sequence is denoted by x_t.

The state set Σ consists of a total of $|\Sigma| = r$ encoder states. It is divided into two state subsets of a first and second type, denoted by Σ_1 and Σ_2, respectively. The encoder has $|\Sigma_1| = r_1$ states of the first type and $|\Sigma_2| = r_2 = r - r_1$ states of the second type. All codewords in the state subset Σ_1 must start with a '0,' while codewords in the state subset Σ_2 can start with either a '0' or a '1.' Here, the state at the time b_t is encoded is denoted by s_t.

The output function h has domain $\Sigma \times B$ and range X. It specifies the following translation

$$x_t = h(s_t, b_t).$$

The output function h can be defined by a simple table lookup [4], based on the input information word b_t and the encoder state s_t provided by the previous codeword. According to the definitions of the state subsets Σ_1 and Σ_2, we have

$$x_t \in \begin{cases} \{X_{00}^1 \cup X_{01}^1\} & \text{if } s_t \in \Sigma_1 \\ \{X_{10} \cup X_{11} \cup X_{00}^2 \cup X_{01}^2\} & \text{if } s_t \in \Sigma_2 \end{cases}$$

where the sets X_{ab}^c are such that $\{X_{00}^1 \cup X_{01}^1\}$ and $\{X_{00}^2 \cup X_{01}^2$ denote the sets of codewords starting with a '0' that are assigned to the first and second state subsets, respectively, and $\{X_{00}^2 \cup X_{01}^2\} = \{X_{00} \cup X_{01}\} \setminus \{X_{00}^1 \cup X_{01}^1\}$. As will become apparent in the following paragraphs, different states cannot contain the same codeword.

The next-state function f has domain $\Sigma \times B$ and range Σ, and it specifies the state of the encoder after transmitting the current codeword. Thus,

$$s_{t+1} = f(s_t, b_t).$$

To facilitate reuse of codewords, i.e., mapping the same codeword to more than one information word to achieve a high coding efficiency, each codeword may enter more than one encoder state. In particular, codewords that end with a '0,' i.e., codewords in subsets X_{00} and X_{10}, may enter any of the r encoder states. Codewords that end with a '1' may enter the r_1 states of the first state set only. This prohibits a codeword ending with '1' from entering states of the second type. Hence,

$$s_{t+1} \in \begin{cases} \Sigma & \text{if } x_t \in \{X_{00} \cup X_{10}\} \\ \Sigma_1 & \text{if } x_t \in \{X_{01} \cup X_{11}\}. \end{cases}$$

Due to the reuse of codewords in encoding, to ensure unique decodability, the set of codewords that belongs to a given state must be disjoint. This attribute implies that any codeword can be unambiguously identified to the state from which it emerged. During decoding, by observing both the current and the next codewords, the decoder can uniquely decide which information word was actually transmitted. Thus, the output function h is chosen such that

$$b_t = h^{-1}(x_t, x_{t+1}).$$

Obviously, the corresponding decoders are sliding-block decoders with zero memory and one codeword anticipation.

B. Definitions of Fibonacci and Generalized Fibonacci Numbers

Fibonacci numbers [9] play an important role in the study of constrained sequences, and particularly in searching the optimum number of encoder states of the FSM described previously. They satisfy the following recurrence relation:

$$\begin{aligned} &F(q) \equiv F(q-1) + F(q-2), \qquad q \geq 2 \\ &F(0) = 0, \quad F(1) = 1. \end{aligned} \tag{1}$$

The first few Fibonacci numbers are $0, 1, 1, 2, 3, 5, 8, 13, 21, \ldots$. It has been found that the number of distinct $d = 1$ sequences as a function of the sequence length, denoted by $\{N_1(q)\}$, is a Fibonacci sequence (FS) satisfying $N_1(q) = F(q+2)$ for $q \geq 0$ [2]. There are various types of generalized Fibonacci numbers. Here, we illustrate one type of numbers proposed by Horadam [10], which is related to the FSM studied in this correspondence. It is defined by

$$G(q) \equiv G(q-1) + G(q-2), \qquad q \geq 2. \tag{2}$$

Obviously, sequences defined by (2) are a generalization of the FS defined by (1), with arbitrary seeds $G(0)$ and $G(1)$. For instance, with $G(0) = 2$ and $G(1) = 1$, the numbers generated are called Lucas numbers, which are denoted by $L(q)$. Note that the FS can be considered as a special case of the previously defined generalized Fibonacci sequences (GFS).

III. Relationship Between the Number of Encoder States and the Probable Size of the Code

To determine the minimum number of encoder states for the designed codes, we start with specific conditions for the code construction, which are derived from the FSM described in Section II A. For $d = 1$ codes, we have [4]

$$r \mid X_{00} \mid + r_1 \mid X_{01} \mid \geq r_1 M \tag{3}$$

$$r\left(\mid X_{00} \mid + \mid X_{10} \mid\right) + r_1\left(\mid X_{01} \mid + \mid X_{11} \mid\right) \geq rM \tag{4}$$

where M is referred to as the probable size of the code, which is essentially the maximum number of information words that the encoder may accommodate, associated with a given number of encoder states. The aforementioned inequalities specify that for a fixed-length code with a probable size of M, the number of codewords leaving a state set, counting multiplicity, should be at least M times the number of states within the state set. Therefore, they are equivalent to the *approximate eigenvector equation*, and are necessary conditions for code construction. Thus, for a desired code rate and number of encoder states, if (3) or (4) fail, a code cannot be constructed. If on the other hand, (3) and (4) both hold, we can proceed to the next step to allocate the valid codewords to various encoder states. To do this, as described in Section II-A, the valid codewords (counting multiplicity) should be assigned to the encoder states such that there is no overlap of codewords between different states. The code construction will fail if such an allocation of codewords is not possible. In such cases, the actual number of information words that the encoder can accommodate will be smaller than M.

To facilitate a successful allocation of codewords and achieve a high coding efficiency, a large value of M, associated with a small number of encoder states, is highly desirable. This is due to the reason that, in general, the larger the value of M and/or the smaller the associated number of encoder states are/is, the more easy it will be to successfully allocate the codewords to the encoder states without incurring the overlap of codewords between different states, and vice versa. In addition, a large value of M may also help to impose other modulation constraints to the designed codes, such as a k constraint, dc-free constraint, MTR constraint, and parity-check constraint. Therefore, in the following sections, we focus on searching the minimum number of encoder states, which maximize the value of M, for any given codeword length.

Note that in (3) and (4), the choice of the number of encoder states r and r_1 determines the value of M, for a given codeword length n. Therefore, in this section, we explore all the possible choices for r and r_1, and derive common properties for these choices.

Proposition 1: For given positive integers r, r_1, and $r_2 = r - r_1$, there always exist a unique GFS $\{G(l)\}$ and a unique integer $q \geq 2$ with the following properties:

1) $G(q) = r$, $G(q-1) = r_1$, and $G(q-2) = r_2$;
2) $G(0) \geq G(1) > 0$.

Proof: Due to the two-term recurrence nature of the GFS (see (2)), for any given r and r_1, we can always define $G(q) = r$ and $G(q-1) = r_1$, and, thereby, generate a GFS based on these two numbers.

To expose the effect of various choices of r and r_1 on M, a judicious selection of indices for the corresponding GFS is crucial. Since r, r_1, and r_2 are all positive integers, without loss of any generality, we define $G(0) > 0$, $G(1) > 0$, and $G(-1) \leq 0$ for all the associated GFS. Thus, we have $G(0) \geq G(1) > 0$. In each of these GFS, with $q \geq 2$, any consecutive three numbers $[r_2 = G(q-2), r_1 = G(q-1)$, and $r = G(q)]$ represent a possible encoder state combination. □

Remarks:

- In Proposition 1, we consider r_1 and r_2 to be positive integers, and the obtained codes are sliding block codes [2]. In principle, the value of r_1 or r_2 can be taken as zero as well. However, in such cases, the encoder has only one type of state, and the resulting codes are block codes [2]. With $r_1 = 0$, conditions (3) and (4) reduced to

$$\mid X_{00} \mid + \mid X_{10} \mid \geq M.$$

The associated codewords are free to start with either '0' or '1,' but must end with a '0.' Similarly, with $r_2 = 0$, we obtain the condition

$$\mid X_{00} \mid + \mid X_{01} \mid \geq M.$$

The corresponding codewords start with a '0,' and end with either '0' or '1.' Obviously, the efficiency this type of codes will be lower than that of the sliding block codes obtained with r_1 and r_2 being positive integers, due to the lack of reuse of codewords during the code construction. The disadvantage of setting $r_1 = 0$ or $r_2 = 0$ on M will be further shown in Section IV.

We use $\{G^i(q)\}$ to denote the ith sequence in the set of all GFS $\mathbf{G}$ defined by Proposition 1, with $i = 1, 2, \ldots, K$ and $K = |\mathbf{G}|$. Note that the size K of $\mathbf{G}$ is finite, since the number of encoder states cannot be infinite. Note also that, hereafter, the GFS under consideration are all within the sequence set $\mathbf{G}$. The basic GFS, with $G^i(2) \leq 10$ and $0 \leq q \leq 11$, are shown in Table I. Here, sequences that are integer multiples of the basic GFS shown in Table I are not included since, as proved by Corollary 1 of Section IV, they produce the same M as the corresponding basic GFS. Note that the first sequence $\{G^1(q)\}$ corresponds to the shifted Fibonacci numbers $\{F'(q)\}$, with $F'(q) = F(q+1)$. The second sequence $\{G^2(q)\}$ corresponds to the Lucas numbers $\{L(q)\}$.

Lemma 1: For the GFS $\{G^i(q)\}$ associated with any given r and r_1 as defined by Proposition 1, there exist numbers $G^i(2)$ and $G^i(1)$ such that $G^i(2) \geq 2G^i(1) > 0$ for $i = 1, 2, \ldots, K$.

Proof: According to Proposition 1, $G^i(0) \geq G^i(1) > 0$, for any $i = 1, 2, \ldots, K$. Further, $G^i(2) = G^i(0) + G^i(1)$. We thus obtain $G^i(2) \geq 2G^i(1) > 0$. □

Theorem 1: For given numbers of encoder states r, r_1, and the associated $G^i(q) = r$ and $G^i(q-1) = r_1$ with $q \geq 2$, as defined by Proposition 1,

$$M = \begin{cases} \left\lfloor \frac{G^i(n+q)}{G^i(q)} \right\rfloor, & \text{if } q \text{ is even} \\ \left\lfloor \frac{G^i(n+q-1)}{G^i(q-1)} \right\rfloor, & \text{if } q \text{ is odd.} \end{cases} \tag{5}$$

Proof: It can be easily shown [2] that

$$\mid X_{00} \mid = N_1(n-2) \tag{6}$$

$$\mid X_{01} \mid = \mid X_{10} \mid = N_1(n-3) \tag{7}$$

$$\mid X_{11} \mid = N_1(n-4). \tag{8}$$

Furthermore, with $G^i(q) = r$ and $G^i(q-1) = r_1$, we can rewrite (3) and (4) as

$$G^i(q)N_1(n-2) + G^i(q-1)N_1(n-3) \geq G^i(q-1)M \tag{9}$$

$$G^i(q)N_1(n-1) + G^i(q-1)N_1(n-2) \geq G^i(q)M. \tag{10}$$

By induction, it can be shown that [11]

$$G^i(q)N_1(n-1) + G^i(q-1)N_1(n-2) = G^i(n+q). \tag{11}$$

Combining (9), (10), and (11), we get

$$M = \min\left(\left\lfloor \frac{G^i(n+q-1)}{G^i(q-1)} \right\rfloor, \left\lfloor \frac{G^i(n+q)}{G^i(q)} \right\rfloor\right). \tag{12}$$

TABLE I
BASIC SEQUENCES FROM THE SET OF GENERALIZED FIBONACCI SEQUENCES (GFS)

$\{G^i(q)\}$ \ q	0	1	2	3	4	5	6	7	8	9	10	11
$\{G^1(q)\}$	1	1	2	3	5	8	13	21	34	55	89	144
$\{G^2(q)\}$	2	1	3	4	7	11	18	29	47	76	123	199
$\{G^3(q)\}$	3	1	4	5	9	14	23	37	60	97	157	254
$\{G^4(q)\}$	4	1	5	6	11	17	28	45	73	118	191	309
$\{G^5(q)\}$	5	1	6	7	13	20	33	53	86	139	225	364
$\{G^6(q)\}$	6	1	7	8	15	23	38	61	99	160	259	419
$\{G^7(q)\}$	7	1	8	9	17	26	43	69	112	181	293	474
$\{G^8(q)\}$	8	1	9	10	19	29	48	77	125	202	327	529
$\{G^9(q)\}$	9	1	10	11	21	32	53	85	138	223	361	584
$\{G^{10}(q)\}$	3	2	5	7	12	19	31	50	81	131	212	343
$\{G^{11}(q)\}$	5	2	7	9	16	25	41	66	107	173	280	453
$\{G^{12}(q)\}$	7	2	9	11	20	31	51	82	133	215	348	563
$\{G^{13}(q)\}$	4	3	7	10	17	27	44	71	115	186	301	487
$\{G^{14}(q)\}$	5	3	8	11	19	30	49	79	128	207	335	542
$\{G^{15}(q)\}$	7	3	10	13	23	36	59	95	154	249	403	652
$\{G^{16}(q)\}$	5	4	9	13	22	35	57	92	149	241	390	631

To compare the values of $\frac{G^i(n+q-1)}{G^i(q-1)}$ and $\frac{G^i(n+q)}{G^i(q)}$ for any given $q \geq 2$ and $n \geq 1$, we define

$$\begin{aligned}\zeta_q &\equiv G^i(n+q-1)G^i(q) - G^i(n+q)G^i(q-1)\\ &= G^i(q-2)G^i(n+q-3) + (G^i(q) - 2G^i(q-1))\\ &\quad G^i(n+q-2). \end{aligned} \tag{13}$$

When $q = 2$, we have

$$\begin{aligned}\zeta_2 &= G^i(n+1)G^i(2) - G^i(n+2)G^i(1)\\ &= G^i(0)G^i(n-1) + (G^i(2) - 2G^i(1))G^i(n).\end{aligned}$$

From Lemma 1, we get $G^i(2) \geq 2G^i(1)$ for the GFS associated with any given r and r_1. Furthermore, $G^i(n-1) > 0$ and $G^i(n) > 0$, for $n \geq 1$. Therefore, we obtain

$$\zeta_2 > 0, \qquad \text{for} \quad \geq 1. \tag{14}$$

Furthermore, since

$$\begin{aligned}\zeta_{q+1} &= G^i(n+q)G^i(q+1) - G^i(n+q+1)G^i(q)\\ &= -(G^i(n+q-1)G^i(q) - G^i(n+q)G^i(q-1)),\end{aligned}$$

we obtain

$$\zeta_{q+1} = -\zeta_q \qquad \text{for} \quad q \geq 2. \tag{15}$$

From (14) and (15), we get $\zeta_3 < 0$, $\zeta_4 > 0$, and so on. Therefore, we conclude that

$$\begin{cases} \zeta_q > 0, & \text{if } q \text{ is even}\\ \zeta_q < 0, & \text{if } q \text{ is odd}\end{cases} \tag{16}$$

for any $n \geq 1$. This proves (5). □

In particular, choosing the number of encoder states to be the Fibonacci numbers $r = F'(q)$ and $r_1 = F'(q-1)$, we can rewrite (13) as

$$\zeta'_q = F'(n+q-1)F'(q) - F'(n+q)F'(q-1). \tag{17}$$

Furthermore, according to d'Ocagne's identity [11], we have

$$\begin{aligned}&F'(n+q-1)F'(q) - F'(n+q)F'(q-1)\\ &= (-1)^q F'(n-1).\end{aligned} \tag{18}$$

Obviously, (18) coincides with (16). This provides another proof of (5), for M associated with the FS $\{F'(q)\}$.

IV. MINIMUM NUMBER OF ENCODER STATES

In this section, we search the minimum number of encoder states r and r_1 that maximizes M, for any desired codeword length. For ease of derivation, we use $M^i(q,n) = \min\left(\frac{G^i(n+q-1)}{G^i(q-1)}, \frac{G^i(n+q)}{G^i(q)}\right)$ to denote M in fractional format (i.e., without applying the floor operator $\lfloor\cdot\rfloor$ in (12) associated with the ith GFS in the sequence set $\mathbf{G}$, and for any given q and n.

From Theorem 1, we conclude the following.

Corollary 1: While comparing the values of M generated by different choices of the number of encoder states r and r_1, it is sufficient to use $n = 1$. The same trend in M exists for other codeword lengths also.

Proof: From Theorem 1, we know that for given $G^i(q) = r$ and $G^i(q-1) = r_1$, depending on whether q is even or odd, $M^i(q,n)$ is either

$$\frac{G^i(n+q)}{G^i(q)} \quad \text{or} \quad \frac{G^i(n+q-1)}{G^i(q-1)}.$$

For both cases, we have $M^i(q,0) = 1$. We further have

$$M^i(q,n) = M^i(q,n-1) + M^i(q,n-2), \qquad \text{for} \quad n \geq 2.$$

Thus, for a given q, the sequence $M^i(q,n)$ can be viewed as a GFS in the fractional format, with seeds $M^i(q,0) = 1$ and $M^i(q,1)$. Therefore, to compare M generated by different choices of r and r_1, it is sufficient to compare the corresponding $M^i(q,1)$. □

From Corollary 1, we can also conclude that for any given q and n, with the same $M^i(q,1)$, integer multiples of the basic GFS produce the same M as the corresponding basic sequences.

Corollary 2: With a given number of encoder states $r = r_1 + r_2$, by choosing $r_1 \geq r_2$, we always obtain a larger M than that with $r_1 < r_2$, for any codeword length n.

Proof: Assume two positive integers a and b with $a \geq b$. Without loss of generality, we prove Corollary 2 by comparing M of the encoder having $[r_2 = G^i(q_1-2) = a, r_1 = G^i(q_1-1) = b, r = G^i(q_1) = a+b]$ with that of the encoder having $[r_2 = G^j(q_2-2) = b, r_1 =$

$G^j(q_2-1) = a, r = G^j(q_2) = a+b]$, with $q_1, q_2 \geq 2$, $i,j = 1,2,\ldots,K$, and $i \neq j$.

For the encoder with $[r_2 = a, r_1 = b, r = a+b]$, we obtain $r \geq 2r_1$. Thus, $G^i(q_1) \geq 2G^i(q_1-1)$. By (13), we obtain $\zeta_{q_1} > 0$. Therefore,

$$M^i(q_1,1) = \frac{G^i(n+q_1)}{G^i(q_1)}|_{n=1} = \frac{a+b+b}{a+b} = 1 + \frac{b}{a+b}. \quad (19)$$

For the encoder with $[r_2 = b, r_1 = a, r = a+b]$, from Theorem 1, we obtain either

$$M^j(q_2,1) = \frac{G^i(n+q_2-1)}{G^i(q_2-1)}|_{n=1} = \frac{a+b}{a} = 1 + \frac{b}{a} \quad (20)$$

or

$$M^j(q_2,1) = \frac{G^i(n+q_2)}{G^i(q_2)}|_{n=1} = \frac{a+b+a}{a+b} = 1 + \frac{a}{a+b}. \quad (21)$$

Comparing (19) with (20) and (21) shows that $M^j(q_2,1) \geq M^i(q_1,1)$. The equality holds only when $a = b$. Therefore, we prove Corollary 2 for $n = 1$. Furthermore, using Corollary 1, we conclude that Corollary 2 is also true for any $n \geq 1$. □

Corollary 3: For a GFS $\{G^i(q)\}$ defined by $G^i(q) = r$ and $G^i(q-1) = r_1$, and with any codeword length n, increasing q from an even integer $q = 2p$, with $p \geq 1$, to an odd integer $q = 2p+1$, results in the same M.

Proof: From (5), it is straightforward to get

$$M^i(2p,n) = M^i(2p+1,n)$$

for all integers $p \geq 1$ and $n \geq 1$. □

Therefore, in each GFS shown in Table I, choosing r from the two numbers within each pair of columns (with $q \geq 2$) results in the same M, irrespective of the codeword length. That is, increasing the encoder complexity from $r = G^i(2p)$ to $r = G^i(2p+1)$ does not increase M.

Corollary 4: For all GFS $\{G^i(q)\}$ defined by $G^i(q) = r$ and $G^i(q-1) = r_1$, and with even integers $q = 2p$ and $p \geq 1$, M increases with increase in p, whether or not the new encoder states are still within the original GFS, and irrespective of the codeword length n.

Proof: What we want to prove is

$$M^i(2p+2,n) > M^j(2p,n) \quad (22)$$

for all integers $p \geq 1$ and $n \geq 1$, and with any $i,j = 1,2,\ldots,K$. We first prove (22) for the case $n = 1$ by induction. For $p = 1$, by Theorem 1, we have

$$M^i(2p+2,1)\,|_{p=1} = \frac{G^i(5)}{G^i(4)} = 1 + \frac{1}{1+\frac{1}{\frac{G^i(3)}{G^i(2)}}} \quad (23)$$

and

$$M^j(2p,1)\,|_{p=1} = \frac{G^j(3)}{G^j(2)} = 1 + \frac{1}{1+\frac{1}{\frac{G^j(1)}{G^j(0)}}}. \quad (24)$$

According to Proposition 1, $\frac{G^j(1)}{G^j(0)} \leq 1$. We further have $\frac{G^i(3)}{G^i(2)} > 1$, since $G^i(3) = G^i(2) + G^i(1)$, and $G^i(1) > 0$. We thus obtain

$$\frac{G^i(3)}{G^i(2)} > \frac{G^j(1)}{G^j(0)}. \quad (25)$$

Combining (23), (24), and (25), we get

$$M^i(2p+2,1) > M^j(2p,1), \qquad \text{for} \quad p = 1.$$

Now, for $p = l$, suppose

$$M^i(2p+2,1) > M^j(2p,1), \qquad \text{with} \quad l \geq 1.$$

Thus, we get

$$\frac{G^i(2l+3)}{G^i(2l+2)} > \frac{G^j(2l+1)}{G^j(2l)}. \quad (26)$$

Then, for $p = l+1$, we have

$$M^i(2p+2,1)\,|_{p=l+1} = \frac{G^i(2l+5)}{G^i(2l+4)} = 1 + \frac{1}{1+\frac{1}{\frac{G^i(2l+3)}{G^i(2l+2)}}} \quad (27)$$

and

$$M^j(2p,1)\,|_{p=l+1} = \frac{G^j(2l+3)}{G^j(2l+2)} = 1 + \frac{1}{1+\frac{1}{\frac{G^j(2l+1)}{G^i(2l)}}}. \quad (28)$$

Combining (26), (27), and (28), we obtain

$$M^i(2p+2,1) > M^j(2p,1), \qquad \text{for} \quad p = l+1.$$

Thus, we prove Corollary 4 for $n = 1$. According to Corollary 1, we conclude that the statement is true for other codeword lengths also. □

Corollary 5: By choosing the number of encoder states as $r = F'(q)$ and $r_1 = F'(q-1)$, we always obtain a larger M than that with r and r_1 being the qth and $(q-1)$th elements of other GFS, for any $q \geq 2$ and any codeword length n. Furthermore, the corresponding number of encoder states is always smaller than that resulting from other GFS.

Proof: Corollary 5 is proved based on the following properties of the FS $\{F'(q)\}$.

- *Property 1:* $\frac{F'(1)}{F'(0)} > \frac{G^{i\neq 1}(1)}{G^{i\neq 1}(0)}$.
  Proof: According to Proposition 1, in the set **G** of GFS, only the FS $\{F'(q)\}$ and its integer multiples satisfy $F'(0) = F'(1)$. All the other GFS result in $G^{i\neq 1}(0) > G^{i\neq 1}(1)$. □
- *Property 2:* $F'(q) < G^{i\neq 1}(q)$, with $q \geq 2$.
  Proof: This is due to the reason that the seeds, i.e., $F'(0) = 1$ and $F'(1) = 1$, are the smallest among all positive integers that can be used as seeds for the GFS in the set **G**. According to the recurrence relation (2) for GFS, the qth element of the FS is always smaller than the qth elements of other GFS in **G**, for $q \geq 2$. □

We first prove

$$M^1(q,n) > M^{i\neq 1}(q,n) \quad (29)$$

for any $q \geq 2$ and $n \geq 1$. We start with $n = 1$ and even q, by induction. For $q = 2$, by Theorem 1, we have

$$M^1(q,1)\,|_{q=2} = \frac{F'(3)}{F'(2)} = 1 + \frac{1}{1+\frac{1}{\frac{F'(1)}{F'(0)}}}$$

and

$$M^{i\neq 1}(q,1)\,|_{q=2} = \frac{G^{i\neq 1}(3)}{G^{i\neq 1}(2)} = 1 + \frac{1}{1+\frac{1}{\frac{G^{i\neq 1}(1)}{G^{i\neq 1}(0)}}}.$$

According to Property 1, we get

$$M^1(q,1) > M^{i\neq 1}(q,1), \qquad \text{for} \quad q = 2.$$

Now, for $q = 2l$, assume that

$$M^1(q,1) > M^{i\neq 1}(q,1), \qquad \text{with} \quad l \geq 1.$$

Then, we get

$$\frac{F'(2l+1)}{F'(2l)} > \frac{G^{i\neq 1}(2l+1)}{G^{i\neq 1}(2l)}. \tag{30}$$

For $q = 2l+2$, we have

$$M^1(q,1)\,|_{q=2l+2} = \frac{F'(2l+3)}{F'(2l+2)} = 1 + \frac{1}{1+\frac{1}{\frac{F'(2l+1)}{F'(2l)}}} \tag{31}$$

and

$$M^{i\neq 1}(q,1)\,|_{q=2l+2} = \frac{G^{i\neq 1}(2l+3)}{G^{i\neq 1}(2l+2)} = 1 + \frac{1}{1+\frac{1}{\frac{G^{i\neq 1}(2l+1)}{G^{i\neq 1}(2l)}}}. \tag{32}$$

Combining (30), (31), and (32), we obtain

$$M^1(q,1) > M^{i\neq 1}(q,1), \qquad \text{for} \quad q = 2l+2.$$

Thus, we prove (29) for the case of $n = 1$ and even integers $q = 2l$. It also holds for odd integers $q = 2l+1$, since $M^i(2l,1) = M^i(2l+1,1)$ according to Corollary 3. Further, using Corollary 1, we conclude that (29) is also true for other codeword lengths. Combining (29) and Property 2, we thus prove Corollary 5. □

Corollary 6: For given $r = F'(q_1)$, choosing $r_1 = F'(q_1-1)$ results in a larger M than that with r_1 being an element of other GFS (i.e., $r = G^{i\neq 1}(q_2) = F'(q_1)$ and $r_1 = G^{i\neq 1}(q_2-1)$, for any $q_1, q_2 \geq 2$), for any codeword length n.

Proof: With $r = G^{i\neq 1}(q_2) = F'(q_1)$, Property 2 of Corollary 5 gives $q_1 > q_2$. According to Corollaries 3 and 4, $M^1(q_1,n) \geq M^1(q_2,n)$, with $q_1 > q_2$. Furthermore, by Corollary 5, we have $M^1(q_2,n) > M^{i\neq 1}(q_2,n)$. Therefore, we obtain $M^1(q_1,n) > M^{i\neq 1}(q_2,n)$ for any $q_1,q_2 \geq 2$ and $n \geq 1$. □

Remarks:

- In the case of $r_1 = 0$ or $r_2 = 0$, the associated basic GFS $\{F''(q)\}$ can be defined as

q	0	1	2	3	4	5	6	$\cdots$
$\{F''(q)\}$	1	0	1	1	2	3	5	$\cdots$

  This is again a shifted FS defined by $F''(q) = F(q-1)$. The encoder state combinations associated with this FS for $r_1 = 0$ and $r_2 = 0$ are given by $[r_2 = F''(0), r_1 = F''(1), r = F''(2)]$ and $[r_2 = F''(1), r_1 = F''(2), r = F''(3)]$, respectively. Theorem 1 still holds for this sequence since $F''(2) > 2F''(1)$. Therefore, according to Corollary 3, we obtain

$$M''(2,n) = M''(3,n) \tag{33}$$

  with $M''(q,n) = \min\left(\frac{F''(n+q-1)}{F''(q-1)}, \frac{F''(n+q)}{F''(q)}\right)$. Further, since

$$\frac{G^i(1)}{G^i(0)} > \frac{F''(1)}{F''(0)}$$

  with any $i = 1,2,\ldots,K$, following a derivation similar to the proof of Corollary 5, we get

$$M^i(2,n) > M''(2,n) \tag{34}$$

  for any $n \geq 1$. In addition, from Corollaries 3 and 4, we have

$$M^i(q,n) \geq M^j(2,n) \tag{35}$$

  for all integers $q > 2$, $n \geq 1$, and with any $i,j = 1,2,\ldots,K$. Combining (33), (34), and (35), we conclude that choosing $r_1 = 0$ or $r_2 = 0$ results in the same M, which is smaller than that with r_1 and r_2 being positive integers, irrespective of the codeword length n.
- Corollaries 5 and 6 show the properties of M associated with the FS $\{F'(q)\}$. It may also be interesting to point out that among other GFS $\{G^{i\neq 1}(q)\}$ in the sequence set $\mathbf{G}$, and for a given $q_1 = 2p+2$ and $p \geq 1$, the Lucas sequence $\{L(q)\}$ provides an M larger than all those associated with $q_2 = 2p$, with the fewest encoder states. This is because, according to Corollary 4, we have $M^{i\neq 1}(2p+2,n) > M^j(2p,n)$. On the other hand, the seeds of the Lucas sequence, i.e., $L(0) = 2$ and $L(1) = 1$, are the smallest among all positive integers that can be used as the seeds for the GFS $\{G^{i\neq 1}(q)\}$ in $\mathbf{G}$. Due to the recurrence relation (2) of the GFS, we obtain $L(q) < G^{i\neq 1,2}(q)$ for any $q \geq 2$.

The aforementioned corollaries are compiled into Theorem 2.

Theorem 2: For any given codeword length n, choosing the number of encoder states to be the Fibonacci numbers $r = F'(q)$ and $r_1 = F'(q-1)$, where q is a positive even integer, always gives the local maximum M with the minimum number of encoder states. The value of M increases with increase in q, and saturates at the global maximum $\lfloor 2^{nC(1,\infty)} \rfloor$.

Proof: From Corollary 2, we know that for a given r, we should choose $r_1 \geq r_2 = r - r_1$ to get a larger M. From Corollary 3, we know that for all GFS defined by Proposition 1 in the sequence set $\mathbf{G}$, choosing the sequence index q such that $q = 2p$ and $p \geq 1$ results in the same M as that with $q = 2p+1$, and choosing $q = 2p$ will reduce the number of states. Furthermore, Corollary 4 shows that the value of M increases with increase in p, for all GFS in $\mathbf{G}$. From Corollary 5, we further know that by choosing the number of encoder states to be the Fibonacci numbers with $r = F'(q)$ and $r_1 = F'(q-1)$, we always obtain a larger M than that with r and r_1 being the qth and $(q-1)$th elements of other GFS in $\mathbf{G}$, for any $q \geq 2$. The associated number of encoder states is always smaller than that chosen from other GFS. Finally, Corollary 6 shows that with the same number of encoder states r, which is a Fibonacci number $F'(q)$, choosing r_1 to be the adjacent number $F'(q-1)$ of the same FS will result in a larger M than that with r_1 being an element of other GFS in $\mathbf{G}$. In addition, all these statements are true irrespective of the codeword length. Therefore, to achieve the maximum M (either locally or globally) with the minimum number of encoder states, we should choose the number of encoder states as $r = F'(2p)$ and $r_1 = F'(2p-1)$, with $p \geq 1$, for any desired codeword length.

The global maximum of M is given by

$$M_{\max} = \left\lfloor \lim_{q\to\infty} \frac{F'(n+q)}{F'(q)} \right\rfloor.$$

For the FS, we have [11]

$$\lim_{q\to\infty} \frac{F'(q+1)}{F'(q)} = \tau \tag{36}$$

where $\tau = \frac{1+\sqrt{5}}{2}$ is the golden ratio [11]. Following (36), we get

$$\lim_{q\to\infty} \frac{F'(n+q)}{F'(q)} = \lim_{q\to\infty} \frac{F'(q+1)}{F'(q)} \frac{F'(q+2)}{F'(q+1)} \cdots \frac{F'(q+n)}{F'(q+n-1)} = \tau^n.$$

With $C(1,\infty) = \log_2(\frac{1+\sqrt{5}}{2})$ [2], we then obtain

$$M_{\max} = \left\lfloor \left(\frac{1+\sqrt{5}}{2}\right)^n \right\rfloor = \lfloor 2^{nC(1,\infty)} \rfloor. \tag{37}$$

Note that in (37), $\lfloor 2^{nC(1,\infty)} \rfloor$ is indeed the theoretical limit of the size of $d = 1$ codes with length n. □

In summary, we conclude that choosing r and r_1 as the Fibonacci numbers, with $[r_2 = 1, r_1 = 1, r = 2]$, $[r_2 = 2, r_1 = 3, r = 5]$, $[r_2 = 5, r_1 = 8, r = 13]$, etc., gives a locally maximum M, with the minimum number of encoder states. The value of M increases with increase in r, until it saturates at the global maximum $M_{\max} = \lfloor 2^{nC(1,\infty)} \rfloor$. These choices of encoder states also help to successfully allocate the valid codewords to the encoder states to maximize the code rate. The final choice of r depends on the desired code rate, the code constraints, and the affordable implementation complexity. For example, based on the previous analysis, we find that with a codeword length of $n = 13$, a five-state encoder with $[r_2 = 2, r_1 = 3, r = 5]$ provides $M = \lfloor M^1(4,13) \rfloor = 516$. It can be verified that these states enable an effective allocation of codewords to accommodate $|B| = M = 516$ information words. As a result, a rate $9/13$ $(1,18)$ code [4] can be designed, whose rate is 3.85% higher than that of the rate $2/3$ $d = 1$ codes [6][7] used for BD and HD-DVD. Furthermore, a 13-state encoder with $[r_2 = 5, r_1 = 8, r = 13]$ generates $M = \lfloor M^1(6,13) \rfloor = 520$. This results in a code with size $|B| = M = 520$, which approaches the theoretical limit of $\lfloor 2^{13C(1,\infty)} \rfloor = 521$. The excess codewords can be used to reduce the k constraint of the rate $9/13$ code to $k = 14$. With $n = 12$, we also find that using a 13-state encoder with $[r_2 = 5, r_1 = 8, r = 13]$, we can construct a $d = 1$ code, whose size achieves the maximum code size of $\lfloor 2^{12C(1,\infty)} \rfloor = 321$. These codes are supposedly the most efficient in terms of the code rate.

V. Conclusion

In this correspondence, we have analytically investigated the relationship between the number of encoder states and the probable size of certain RLL codes. We have found that the number of encoder states can always be associated with generalized Fibonacci numbers. Choosing the number of encoder states to be specific Fibonacci numbers maximizes the probable size of the designed code with the minimum number of states, for any desired codeword length. These states, in general, also enable the successful allocation of codewords to the encoder states to maximize the code rate. Our analysis provides direct guidelines for the design of capacity-approaching RLL codes with $d = 1$ constraint. This analysis can be generalized to other finite-state constrained codes as well.

Acknowledgment

The authors would like to thank Dr. G. Mathew and Prof. J. W. M. Bergmans for their insights and help in preparation of this paper. They would also like to thank the anonymous reviewers and the Associate Editor for their careful reading and detailed critique of the manuscript. Their suggestions have helped greatly to improve the original draft.

References

[1] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, pp. 379–423, Jul. 1948.
[2] K. A. S. Immink, *Codes for Mass Data Storage Systems*. Den Haag, The Netherlands: Shannon Foundation, 1999.
[3] R. L. Adler, D. Coppersmith, and M. Hassner, "Algorithms for sliding block codes: An application of symbolic dynamics to information theory," *IEEE Trans. Inform. Theory*, vol. IT-29, no. 1, pp. 5–22, Jan. 1983.
[4] K. A. S. Immink, J. Y. Kim, S. W. Suh, and S. K. Ahn, "Efficient Dc-free RLL codes for optical recording," *IEEE Trans. Commun.*, vol. 51, no. 3, pp. 326–331, Mar. 2003.
[5] K. Cai and K. A. S. Immink, "Method and system for encoding and decoding information with modulation constraints and error control," Singapore, PCT application PCT/SG2004/000357, filed on Oct. 26, 2004.
[6] T. Narahara, S. Kobayashi, Y. Shimpuku, G. van den Enden, J. Kahlman, M. van Dijk, and R. van Woudenberg, "Optical disc system for digital video recording," *Japan J. Appl. Phys.*, vol. 39, no. 2B, pt. 1, pp. 912–919, 2000.
[7] T. Iwanaga, S. Ogkubo, M. Nakano, M. Kubota, H. Honma, T. Ide., and R. Katayama, "High-density recording systems using paritial response maximum likelihood with blue lader diode," *Japan J. Appl. Phys.*, vol. 42, no. 2B, pt. 1, pp. 1042–1043, Feb. 2003.
[8] T. Nishiya, K. Tsukano, T. Hirai, S. Mita, and T. Nara, "Rate 16/17 maximum transition run (3;11) code on an EEPRML channel with an error-correcting postprocessor," *IEEE Trans. Magnetics*, vol. 35, no. 5, pp. 4378–4386, Sep. 1999.
[9] N. N. Vorob'ev, *Fibonacci Numbers*. New York: Blaisdel, 1961.
[10] A. F. Horadam, "Generating functions for powers of certain generalized sequence of numbers," *Duke Math. J.*, vol. 32, pp. 437–446, 1965.
[11] S. Vajda, *Fibonacci and Lucas Numbers, and the Golden Section: Theory and Applications*. New York: Halsted , 1989.

On the Conjectures of SU(3) and AB Unitary Space–Time Codes

Hsiao-Feng (Francis) Lu, *Member, IEEE*

***Abstract*—Proofs to the conjectures made by Jing and Hassibi on having fully diverse (3 × 3) SU(3) and AB unitary space–time codes are presented in this correspondence. We first prove that the SU(3) codes are fully diverse if and only if the design parameters P, Q, R, and S are all odd integers, and in addition, are relatively prime. For the type I AB codes, it is shown that full diversity can be achieved if and only if the integers P, Q, R, and S are relatively prime. Finally, we show that such condition is also sufficient for having fully diverse type II AB codes.**

***Index Terms*—Algebraic number theory, cyclotomic number field, Lie group, multiple-antenna system, unitary space–time code.**

I. Introduction

The significance of using multiple transmit and receive antennas to communicate over Rayleigh flat-fading channels with higher data rate and better reliability has been well recognized in [1], [2], [3]. Codes specifically designed for this multiple-antenna scenario are termed *space–time codes* [1]. Among all the space–time codes currently available in the literatures, the $(M \times M)$ unitary space–time codes are the codes consisting of $(M \times M)$ unitary matrices and are designed specifically for the system with M transmit antennas. Analogous to the concept of differential PSK modulation used in conventional digital communication systems [4], these unitary codes are usually

Manuscript received June 2, 2004; revised December 5, 2005. This work was supported by the Taiwan National Science Council under Grants NSC 93-2218-E-194-012, NSC 94-2213-E-194-013, and NSC 94-2213-E-194-019.

The author is with the Department of Communication Engineering, National Chung-Cheng University, Min-Hsiung, Chia-Yi, 621 Taiwan, R.O.C.(e-mail: francis@ccu.edu.tw).

Communicated by Ø. Ytrehus, Associate Editor for Coding Theory.
Digital Object Identifier 10.1109/TIT.2006.876233