

GENERATING SERIES AND CAPACITY FOR CONSTRAINED SEQUENCES

Cecilio Pimentel

Communications Research Group - CODEC
 Department of Electronics and Systems, P.O. Box 7800
 Federal University of Pernambuco
 50711-970 - Recife - Brazil
 E-mail : cecilio@npd.ufpe.br

Resumo - Uma técnica bem estabelecida para a determinação da capacidade de seqüências discretas modeladas por um diagrama de transição de estados finitos (FSTD) é baseada no maior autovalor da matriz adjacência da FSTD. Neste artigo, o problema do cálculo da capacidade é colocado em uma perspectiva que permite o emprego de métodos de combinatória enumerativa para achar o número de seqüências de comprimento n satisfazendo uma determinada restrição. Vários exemplos são usados como ilustração.

Abstract - A well known technique for determining the capacity of constrained sequences modeled as a finite state transition diagram (FSTD) is based on the largest eigenvalue of the adjacent matrix of the FSTD. In this paper, we rephrase the calculation of capacity by putting it into a perspective which allows us to employ a standard combinatorial enumerative technique to find the number of sequences of length n satisfying a given constraint. Several examples are used as illustration.

Keywords: constrained codes, Shannon capacity, combinatorial methods, generating series.

1. INTRODUCTION

Runlength-limited sequences have been used in many magnetic and optical digital storage as well as data transmission systems. The constraints introduced into the sequence offer the possibility of achieving a desired spectral shaping, reducing the intersymbol interference and improving the synchronization ability of communication systems.

The problem of determining the Shannon capacity, or simply capacity of constrained sequences, originates with Shannon [1]. If S is the set of all binary sequences of arbitrary length satisfying a certain constraint, the capacity of the set S , denoted as C , is defined as:

$$C = \lim_{n \rightarrow \infty} \frac{\log_2 c_n}{n},$$

where c_n is the number of sequences in S of length n . The number of sequences in S grows exponentially for large n as 2^{nC} . Shannon showed that the capacity provides an upper bound on the code rate of any finite-state code that can be used to encode the unconstrained data sequence into the constrained sequence.

The technique proposed by Shannon for finding the capacity of constrained sequences that can be modeled by an FSTD is described in detail in reference [2]. The FSTD may be easily transformed into an adjacent matrix which indicates the allowed state transitions. Given the adjacent matrix A of the FSTD, it was shown by Shannon that the capacity is expressed as:

$$C = \log_2 \lambda,$$

where λ is the largest real eigenvalue of A . Equivalently, the capacity is the base two logarithm of the largest real root of the characteristic polynomial of A . The capacity can identify codes which are too constrained or inappropriate for a certain application.

In this paper a standard enumeration technique is given to find a simple formula for a formal power series, the generating series, for some classes of constrained sequences, such that the number of binary sequences in S , c_n , are the coefficients of appropriate power of indeterminates. Moreover, using an argument similar to Shannon's, the capacity is expressed as the base two logarithm of the inverse of the largest real root of the denominator polynomial in the polynomial ratio representation of the generating series. Several classes of binary constrained sequences of interest in digital data recording are used as illustration, including $RLL(r, s)$, $RLL_o(d, k)$, $ARLL(r, s)-(e, l)$, and $RLL_o(d, k, s)$ sequences. One example of M -ary sequences is considered as well.

We now define some notation for further reference. A generic binary sequence will be denoted by $\sigma = \sigma_1\sigma_2\cdots$. If A and B are sets of binary sequences, the concatenation product of A and B , defined as $AB = \{ab \mid a \in A, b \in B\}$ is the set formed by following any sequence in A with any sequence in B , $A^2 = AA$ and $A^* = \phi \cup A \cup A^2 \cup A^3 \dots$, where ϕ is the empty sequence, i.e., a sequence of length zero that possesses the property $\phi a = a\phi$. In particular, $\{0, 1\}^*$ is the set of all binary sequences. The Cartesian product of A and B , denoted as $A \times B$, is the set of all ordered pairs (a, b) where $a \in A$ and $b \in B$. For example, if $A = \{0, 1, 01\}$ and $B = \{1, 0, 10\}$, then

$$\begin{aligned} A \times B &= \{(0, 1), (0, 0), (0, 10), (1, 1), (1, 0), \\ &\quad (1, 10), (01, 1), (01, 0), (01, 10)\}; \\ AB &= \{01, 00, 010, 11, 10, 110, 011, 0110\}. \end{aligned}$$

In this particular example, the set $A \times B$ has 9 elements and the set AB has 8 elements. 1^r denotes a sequence of r consecutive ones, for example, $1^3=111$. The field of rational numbers is denoted by Q . The next section contains background on generating series in commuting indeterminates.

2. THE GENERATING SERIES

Let $w : S \rightarrow \{0, 1, 2, \dots\}$ be a weight function such that $w(\sigma)$ is the weight of σ with respect to w . The weight function w records designed information about the sequence σ . Let $Q[[x]]$ be the ring of all formal power series in commuting indeterminate x with coefficients taken from the field of rational numbers Q . The ordinary generating series for S with respect to weight function w , denoted by $[(S, w)]$, in the indeterminate x , is defined as:

$$[(S, w)] = \sum_{\sigma \in S} x^{w(\sigma)} \in Q[[x]].$$

The generating series $[(S, w)]$ is to be regarded as a formal power series in $Q[[x]]$ which retains combinatorial information on coefficients. We say that the indeterminate x marks w , and that $[(S, w)]$ enumerates S with respect to w . The reason for introducing this series is justified by the straightforward observation that if $w(\sigma)$ is the length of σ , the desired quantity c_n is the coefficient x^n in the formal power series $[(S, w)]$.

The generating series approach provides us with a strategy for solving problems involving sequences in a commutative ring. One possible strategy for determining the generating series is to select a weight preserving bijection that decomposes the set S into constituent subsets under the operation of disjoint union and Cartesian product, such that it is easier to determine the generating series for these subsets. The enumeration of the original set is given by the Sum and Product Lemmas [3], which we state in the following two lemmas.

Lemma 2..1 (The Sum Lemma) *Let S_1 and S_2 be a partition of S with weight function w . Then*

$$[(S, w)] = [(S_1, w)] + [(S_2, w)].$$

Lemma 2..2 (The Product Lemma) *Let B_1 and B_2 be sets with weight functions w_1 and w_2 , respectively. Let $S = B_1 \times B_2$ be the Cartesian product of B_1 and B_2 with weight function defined by $w(\sigma) = w_1(b_1) + w_2(b_2)$, for all $\sigma = (b_1, b_2) \in B_1 \times B_2$. Then*

$$[(B_1 \times B_2, w)] = [(B_1, w_1)] [(B_2, w_2)].$$

From now on, we will assume that each element in a concatenated set \mathcal{AB} is uniquely generated, that is, for each sequence in \mathcal{AB} we can determine uniquely which part came from \mathcal{A} and which part came from \mathcal{B} . This is the case, for example, when $\mathcal{A} = \{0, 00, 000\}$ and $\mathcal{B} = \{1, 11, 111\}$. Under this assumption the set \mathcal{AB} will effectively be the Cartesian product. Therefore, from the Product lemma we have:

$$[(\mathcal{AB}, w)] = [(\mathcal{A}, w)] [(\mathcal{B}, w)],$$

and consequently we can write:

$$[(S^*, w)] = 1 + [(S, w)] + [(S^2, w)] + \dots = (1 - [(S, w)])^{-1}.$$

For example, the generating series for the set of all $\{0, 1\}$ -strings, $\{0, 1\}^*$, with respect to the weight function length of σ is:

$$[(\{0, 1\}^*, w)] = (1 - [(0, 1), w])^{-1} = (1 - 2x)^{-1},$$

where the indeterminate x marks the number of 0's and 1's. The next section discusses the enumeration and capacity of the sequences under consideration.

3. SEQUENCE ENUMERATION AND CAPACITY

We define σ to be an RLL(r, s) sequence if all runs of zeros and ones of σ have length at least r and at most s . This class of sequences arises in magnetic recording systems where the maximum runlength s is usually imposed for reasons of clock recovery, while the r constraint is used to mitigate the effects of intersymbol interference. For example, $\sigma = (0001100111100)$ is an RLL(2,4) sequence of length 13. The required generating series for a generic set S with respect to the weight function length of σ is:

$$F_S(x) = \sum_{n=0}^{\infty} c_n x^n \in Q[[x]].$$

For the remainder of this section we consider examples of the calculation of $F_S(x)$ for some classes of constrained sequences. The key point to solve problems of this type is to find a bijection that expresses the set S in terms of concatenation products of binary strings.

The generating series for RLL(r, s) sequences is obtained directly from the following decomposition of the set $\{0, 1\}^*$:

$$\{0, 1\}^* = 1^* \{00^* 11^*\}^* 0^*,$$

since any binary sequence begins with a sequence (possibly empty) of 1's, then alternating block's of 0's and 1's, and ends with a sequence (possibly empty) of 0's. It is now clear that the set of all RLL(r, s) sequences, denoted as $S_{r,s}$ (this set will also be referred to as the RLL(r, s) code) can be expressed as:

$$S_{r,s} = \{\phi, 1^r, 1^{r+1}, \dots, 1^s\} \{0^r, 0^{r+1}, \dots, 0^s, 1^r, 1^{r+1}, \dots, 1^s\}^* \{\phi, 0^r, 0^{r+1}, \dots, 0^s\}. \quad (1)$$

If the indeterminate x marks the length of the sequence, note that the generating series for the sets $\{\phi, 1^r, 1^{r+1}, \dots, 1^s\}$ or $\{\phi, 0^r, 0^{r+1}, \dots, 0^s\}$ is $1 + x^r + x^{r+1} + \dots + x^s$. Therefore, $F_{S_{r,s}}(x)$ is given by:

$$\begin{aligned} F_{S_{r,s}}(x) &= \\ &= (1 + x^r + x^{r+1} + \dots + x^s)^2 (1 - (x^r + x^{r+1} + \dots + x^s)^2)^{-1}; \\ &= \frac{1 + \frac{x^r - x^{s+1}}{1-x}}{1 - \frac{x^r - x^{s+1}}{1-x}}; \\ &= \frac{1 - x + x^r - x^{s+1}}{1 - x - x^r + x^{s+1}}, \end{aligned} \quad (2)$$

where we have used the fact that the generating series for the set S^* is $(1 - F_S(x))^{-1}$. The series expansion of Equation (2) yields the coefficients c_n , as we can see in the example below, for $r = 3$ and $s = 7$:

$$F_{S_{3,7}}(x) = 1 + 2x^3 + 2x^4 + 2x^5 + 4x^6 + 6x^7 + \dots + 20x^{11} + 28x^{12} + 42x^{13} + 58x^{14} + \dots$$

We can also readily seen from Equation (2) that the coefficients c_n for the RLL(r, s) code satisfy the following recurrence equation:

$$c_n - c_{n-1} - c_{n-r} + c_{n-s-1} = 0, \quad (3)$$

for $n > s + 1$, with initial conditions:

$$\begin{aligned} c_0 &= 1; \\ c_i &= 0, && \text{if } 1 \leq i \leq r - 1; \\ c_r &= 2 + c_{r-1}; \\ c_i &= c_{i-1} + c_{i-r}, && \text{if } r + 1 \leq i \leq s; \\ c_{s+1} &= -2 + c_s + c_{s+1-r}. \end{aligned}$$

We now consider the generating series for the set $S_{r,\infty}$, that is, sequences containing runs of zeros and ones of length at least r . This code is referred to as RLL(r, ∞). When s goes to infinity the term $(1 + x^r + x^{r+1} + \dots)$ has a closed form given by $1 + x^r / (1 - x)$. Therefore:

$$F_{S_{r,\infty}}(x) = \frac{(1 + \frac{x^r}{1-x})^2}{1 - (\frac{x^r}{1-x})^2} = \frac{1 - x + x^r}{1 - x - x^r}. \quad (4)$$

The asymptotic behavior of c_n is the main subject of the next section. More example of the calculation of generating series will also be given.

4. CAPACITY CALCULATION

To find the capacity of a constrained sequence we need to examine the asymptotic behavior of c_n for large n . For the class of sequences we are considering, the generating series $F_S(x)$ can be expressed as the ratio of two polynomials $F_S(x) = N(x)/D(x)$. On considering the partial fraction expansion of $F_S(x)$, we would expect c_n to be a linear combination of the n^{th} power of roots of the following polynomial equation:

$$c_n = A_1(n)\alpha_1^n + A_2(n)\alpha_2^n + \dots + A_T(n)\alpha_T^n,$$

where T is the degree of $D(x)$, $A_1(n), \dots, A_T(n)$ are polynomials in n , and the α_i 's are the roots of the reciprocal polynomial $D^*(x) = x^T D(1/x)$. Suppose that the root α_1 has the largest absolute value. Thus

$$\frac{c_n}{A_1(n)\alpha_1^n} = 1 + \frac{A_2(n)}{A_1(n)} \left(\frac{\alpha_2}{\alpha_1}\right)^n + \dots + \frac{A_T(n)}{A_1(n)} \left(\frac{\alpha_T}{\alpha_1}\right)^n.$$

It is clear that:

$$\lim_{n \rightarrow \infty} \frac{c_n}{A_1(n)\alpha_1^n} = 1,$$

since α_i/α_1 all have absolute value less than 1 for all $i = 2, \dots, T$. Thus, asymptotically, the capacity is given by:

$$C = \lim_{n \rightarrow \infty} \frac{\log_2 c_n}{n} = \lim_{n \rightarrow \infty} \frac{\log_2 A_1(n)\alpha_1^n}{n} = \log_2 \alpha_1.$$

Therefore, the capacity is given by the base two logarithm of the largest real root of the reciprocal polynomial $D^*(x)$. It turns out that the root of this equation with the largest magnitude is real and positive. The equivalence between the generating series and Shannon's approach is clear, $D^*(x)$ is the characteristic polynomial of the adjacency matrix of the FSTD that represents the constraints.

From Equation (2) we have that $D^*(x)$ for the RLL(r, s) code is $x^{s+1} - x^s - x^{s-r+1} + 1$. As an illustration, Table 1 shows the capacity of RLL($2, s$) codes for small values of s . The capacity of this code converges for large s to

Table 1: Capacity of RLL($2, s$) codes

s	C
3	.4057
4	.5515
5	.6174
6	.6569
7	.6690
8	.6793
9	.6853
10	.6899

$\log_2(1.618) = 0.6942$, since 1.618 is the largest real root of the polynomial $D^*(x) = x^2 - x - 1$ given by Equation (4).

In some optical recording channel applications the minimum and maximum run-length of 0's and 1's are required to be different [4]. This gives rise to a new constrained code called asymmetrical runlength-limited (ARLL) code [4] with parameters (r, s) - (e, l) , where (r, s) represent the minimum and maximum runlength of 1's, and (e, l) are the analogous parameters of 0's. By modifying Equations (1) and (2) accordingly, we conclude that the generating series for ARLL(r, s)- (e, l) codes is:

$$F_{S_{(r,s),(e,l)}}(x) = \frac{(1 - x + x^r - x^{s+1})(1 - x + x^e - x^{l+1})}{1 - 2x + x^2 - x^{s+l+2} + x^{s+e+1} + x^{r+l+1} - x^{r+e}} \quad (5)$$

The capacity follows readily from the denominator polynomial of the equation above. Moreover, the reciprocal denominator polynomial for the ARLL(r, ∞)- (e, ∞) code is $D^*(x) = x^{e+r} - 2x^{e+r-1} + x^{e+r-2} - 1$. It is also of interest to impose additional limitations on these codes by requiring that the number of consecutive ones must be of even length [4]. This modified ARLL(r, ∞)- (e, ∞) code is denoted MARLL(r, e). The allowed sequences of this code are:

$$S_{(r,e)} = \{ \phi, 1^r, 1^{r+2}, 1^{r+4}, \dots \} \{ 0^e, 0^{e+1}, 0^{e+2}, \dots, 1^r, 1^{r+2}, 1^{r+4}, \dots \}^* \{ \phi, 0^e, 0^{e+1}, 0^{e+2}, \dots \}.$$

Thus

$$\begin{aligned} F_{S_{(r,e)}} &= \frac{(1 + \frac{x^r}{1-x^2})(1 + \frac{x^e}{1-x^2})}{1 - \frac{x^{r+e}}{1-x^2} + \frac{x^{r+e}}{1-x^2} - \frac{x^{r+e}}{1-x^2}} \\ &= \frac{(1-x-x^r)(1-x^2-x^e)}{1-x-x^2+x^3-x^{r+e}}. \end{aligned}$$

The code MARLL($2, 1$) is the so-called even-marked-modulation (EMM) [4]. The capacity of this code is the largest root of $D^*(x) = x^2 - x - 1$, or $C=0.6942$, the same capacity as the RLL($2, \infty$) code.

5. CAPACITY OF RLL_o(d, k) SEQUENCES

We refer to σ to be an RLL_o(d, k) sequence if between consecutive ones there are at least d and at most k zeros. The IBM RLL_o($2, 7$) code [5] constitutes an example of practical use of these codes. The RLL_o(d, k) code is asymptotically equivalent to the set of sequences that ends with a 1, where each run of zeroes satisfies an RLL_o(d, k) constraint. This set

is expressed as $S_{d,k} = \{(0^d, \dots, 0^k)1\}^*$. So

$$F_{S_{d,k}}(x) = \frac{1-x}{1-x-x^{d+1}-x^{k+2}}. \quad (6)$$

From Equations (2) and (6) we can check the known result that the capacity of $RLL_o(d, k)$ and $RLL(d+1, k+1)$ codes is the same [6]. The recursion given by Equation (3) is also valid for $RLL_o(d, k)$ codes where $r = d+1$ and $s = k+1$, but with different initial conditions. Bounds on the capacity of $RLL_o(d, k)$ codes are given in reference [7].

An example of a more complicated system is the code $RLL_o(d, k, s)$ [2] where the parameter s indicates that the number of consecutive zeros must be of the form $d + is$, for $i = 0, 1, \dots$. For example, when $d = s = 2$, only even number of zeros are allowed. The denominator polynomial of the generating series for $RLL_o(d, k, s)$ codes is $D(x) = 1 - x^s + x^{1+d+s+s[\frac{k-d}{s}]} - x^{d+1}$.

Recently, practical applications of M -ary modulation schemes can be achieved on some optical and magnetic media [8]. This offers the possibility of achieving large information density on the channel. An M -ary $RLL_o(d, k)$ code, denoted as $RLL_o(M, d, k)$, is one where at least d and at most k zeroes occur between non-zero symbols. Similarly to the development above, the set of all sequences is $S_{M,d,k} = \{(0^d, \dots, 0^k)1, 2, \dots, (M-1)\}^*$. Consequently, $F_{S_{M,d,k}}(x)$ is written as:

$$F_{S_{M,d,k}}(x) = \frac{1 - (x^d + \dots + x^k)(M-1)x^{-1}}{1 - x + (M-1)x^{2+k} - (M-1)x^{1+d}}. \quad (7)$$

The reciprocal denominator polynomial of the generating series $F_{S_{M,d,k}}(x)$ is:

$$D^*(x) = x^{k+2} - x^{k+1} - (M-1)x^{k-d+1} + M - 1.$$

6. CONCLUSIONS

We have presented an enumerative approach to address the problem of finding the capacity of constrained sequences that can be expressed as the concatenation product of M -ary strings. The capacity is expressed as the base two logarithm of the inverse largest root of a polynomial whose coefficients are found directly from the parameters of the constraints. The examples shown here constitute in most cases well-known results but it forms the basis of an approach that can be expanded. Perhaps of more importance than these particular results are the enumeration techniques used to derive them.

REFERENCES

- [1] C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, vol. 27, pp.379-423, July 1948.
- [2] B. H. Marcus and P. H. Siegel and J. K. Wolf, "Finite-state modulation codes for data storage," *IEEE Journal on Selected Areas in Communications*, vol.10, pp. 5-37, January 1992.

- [3] I. P. Goulden and D. M. Jackson, *Combinatorial Enumeration*, John Wiley & Sons, 1983.
- [4] R. Karabed and P. H. Siegel, "Matched spectral-null codes for partial-response channels", *IEEE Transactions on Information Theory*, vol. 37, pp. 818-855, May 1991.
- [5] T. Howell, "Analysis of correctable errors in the IBM 3380 disk file," *IBM J. Research development*, vol. 28, pp. 206-211, March 1984.
- [6] V. D. Kolesnik and V. Y. Krachkovsky, "Generating functions and lower bounds on rates for limited error-correcting codes," *IEEE Transactions on Information Theory*, vol. 37, pp. 779-788, May 1991.
- [7] E. Zehavi and J. K. Wolf, "On runlength codes," *IEEE Transactions on Information Theory*, vol. 34, pp. 45-54, January 1988.
- [8] S. W. McLaughlin and J. Luo and Q. Xie, "On the capacity of M -ary runlength-limited codes," *IEEE Transactions on Information Theory*, vol. 41, pp. 1508-1511, September 1995.

Cecilio Pimentel was born in Recife, Brazil, on June 7, 1966. He received the B.Sc degree from the Federal University of Pernambuco, Brazil, in 1987, the M.Sc degree from the Catholics University of Rio de Janeiro, Brazil, in 1990 and the Ph.D. degree from the University of Waterloo, Ontario, Canada, in 1996, all in electrical engineering. Since October 1996, he has been with the Department of Electronics and Systems at the Federal University of Pernambuco. His research interests include digital communications, information theory, and error correcting coding.