Optimal Block-Type-Decodable Encoders for Constrained Systems

Panu Chaichanavong, *Student Member, IEEE,* and Brian H. Marcus, *Fellow, IEEE*

***Abstract*—A constrained system is presented by a finite-state labeled graph. For such systems, we focus on block-type-decodable encoders, comprising three classes known as block, block-decodable, and deterministic encoders. Franaszek gives a sufficient condition which guarantees the equality of the optimal rates of block-decodable and deterministic encoders for the same block length. In this paper, we introduce another sufficient condition, called the straight-line condition, which yields the same result. Run-length limited RLL(d, k) and maximum transition run MTR(j, k) constraints are shown to satisfy both conditions. In general, block-type-decodable encoders are constructed by choosing a subset of states of the graph to be used as encoder states. Such a subset is known as a set of principal states. For each type of encoder and each block length, a natural problem is to find a set of principal states which maximizes the code rate. We show how to compute the asymptotically optimal sets of principal states for deterministic encoders and how they are related to the case of large but finite block lengths. We give optimal sets of principal states for MTR(j, k)-block-type-decodable encoders for all codeword lengths. Finally we compare the code rate of nonreturn to zero inverted (NRZI) encoders to that of corresponding nonreturn to zero (NRZ) and signed NRZI encoders.**

***Index Terms*—Constrained system, maximum transition run system, precoding, principal states, run-length-limited (RLL) system.**

I. Introduction

In modulation coding, one encodes arbitrary user data into sequences that satisfy some constraint that improves the performance of a communications or recording channel—in particular, a magnetic or optical recording channel. The best known constraint is the run-length-limited (RLL(d, k)) constraint on binary sequences, in which runs of zeros are bounded below by d and bounded above by k. Sequences satisfying the RLL(d, k) constraint correspond to consecutive edge labels in the finite-state machine shown in Fig. 1.

Run-length constraints help to mitigate problems of intersymbol interference and inaccurate clocking (the d-constraint for the former and the k-constraint for the latter). These constraints have been used in recording channels since the inception of the disk drive. They are still used today in some recording systems—in particular, those which rely on relatively simple detection methods, such as peak detection. This includes some magnetic tape drives and optical disk drives.

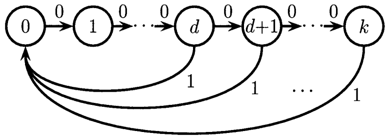

Fig. 1. RLL(d, k) constraint.

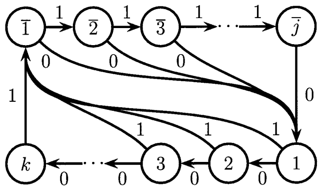

Fig. 2. MTR(j, k) constraint.

Of particular importance for today's high-density disk drives are constraints that enhance the performance of more complex detection methods, such as partial response maximum likelihood (PRML). One important example is the class of maximum transition run (MTR(j, k)) constraints [13], in which runs of zeros are bounded above by k and runs of ones are bounded above by j (see Fig. 2). The k-constraint plays the same role for clocking as mentioned above. The j-constraint is imposed in order to increase the minimum distance between distinct codewords and therefore provide error-correction coding gain. This constraint also helps the recording head to switch polarity sufficiently fast and yet still saturate the recording medium. Typical values of j and k can be roughly 4 and 15, but a wide range of values have been considered.

Other important constraints include asymmetrical RLL and multiple-spaced run-length constraints used in optical recording, charge constraints used in both recording and communications channels, and $(0, G/I)$ constraints for timing recovery and reduction of path memory in PRML.

It is well known that for any constraint, there exist encoders at any rate up to capacity. However, the corresponding decoder may propagate errors. For this reason, there has been much attention focused on block encoders. While block encoders are conceptually simplest, it may be possible to achieve higher rates using block-decodable codes (which still limits error propagation to one block) instead.

In general, for a given block length, determining the optimal rate of a block-decodable encoder can be very difficult. However, this problem is considerably more tractable for the class of

Manuscript received August 4, 2002; revised December 9, 2002. The material in this paper was presented in part at the IEEE International Symposium on Information Theory, Lausanne, Switzerland, June/July 2002.

P. Chaichanavong is with the Department of Electrical Engineering, Stanford University, Stanford, CA 94305 USA (e-mail: panu@stanford.edu).

B. H. Marcus was with IBM Almaden Research Center, San Jose, CA. He is now with the Department of Mathematics, University of British Columbia, Vancouver, BC V6T 1Z2 Canada (e-mail: marcus@math.ubc.ca).

Communicated by R. Urbanke, Associate Editor for Coding Techniques.

Digital Object Identifier 10.1109/TIT.2003.810634

deterministic encoders (see Section III for definitions of these encoders). While deterministic encoders do not necessarily have good error propagation properties, it turns out that for some natural constraints, including RLL(d, k) and MTR(j, k), the optimal rates of deterministic and block-decodable encoders coincide for every block length.

For some well-known families of constrained systems, optimal encoders have been completely characterized. Lee and Wolf [9], [10] explicitly computed optimal block encoders for RLL(d, k) constraints. Optimal block-decodable encoders for the same constraints are due to Gu and Fuja [5]. For $(0, G/I)$ constraints, optimal block encoders were found by Abdel-Ghaffar and Weber [1]. In this paper, we exhibit optimal encoders for MTR(j, k) constraints in Section VI.

An outline of our paper is as follows. In Section II, we summarize some necessary background material on constrained coding. In Section III, we give formal definitions of the three classes of encoders that we consider, and we give, in principle, a description of how the optimal encoder of each type can be constructed using Franaszek's notion of a set of principal states [2], [3]. We give a complete proof of a sufficient condition, due to Franaszek, for the equality of the optimal rates of block-decodable and deterministic encoders. Then we introduce a variation of this condition, called the straight-line condition. Several examples of constrained systems are shown to satisfy these conditions. Natural examples of systems where equality fails are also given. In Section IV, we study the asymptotic behavior of deterministic encoders. We provide a simple method to determine an asymptotically optimal set of principal states based on eigenvectors of an adjacency matrix corresponding to the constraint. In Section V, we apply the result in Section IV to characterize the asymptotically optimal block-decodable encoders for RLL(d, k) and MTR(j, k). We begin Section VI by a review of the results from Lee and Wolf [9], [10] on optimal block encoders and Gu and Fuja [5] for optimal block-decodable encoders for RLL(d, k) for all block lengths. Then, we present optimal block, block-decodable, and deterministic encoders for MTR(j, k) for all block lengths. Some proofs of the lemmas in this section are given in the Appendix. Finally, we study the effect of precoding on the code rate in Section VII.

II. Background

In this section, we summarize some necessary background and definitions. For more detail, the reader is referred to [8] and [11].

A *labeled graph* $G = (V, E, L)$ [11] consists of a finite set of states $V = V_G$, a finite set of edges $E = E_G$ where each edge has an initial state and a terminal state in V_G, and an edge labeling $L = L_G: E \rightarrow \Sigma$ where Σ is a finite alphabet. We will be concerned mainly with finite sequences, called *blocks* or *codewords;* a *q-block* is a block of length q. Formally, a *constrained system* or *constraint* $S = S(G)$ is the set of finite sequences obtained by reading the edge labels of a labeled graph G. Such a graph is called a *presentation* of the constraint. A presentation is called *deterministic* if at each state all outgoing edges carry distinct labels. It is well known that every constrained system has a deterministic presentation. A graph is *lossless* if any two distinct paths with the same initial state and terminal state have different labelings. This is a weaker property than deterministic.

A graph is *irreducible* if for any given pair u, v of states there is a path from u to v and a path from v to u. A graph is *primitive* if there exists a positive integer N such that for all pairs u, v of states there are paths from u to v and v to u of length N. A constraint is said to be irreducible (resp., primitive) if it has an irreducible (resp., primitive) presentation. Most constraints of practical interest, including RLL, MTR, and charge constraints, are irreducible; in fact, except for trivial cases, the RLL and MTR constraints are primitive. Moreover, any constraint can, in some sense, be broken down into irreducible pieces. For these reasons, we will consider only irreducible constraints (in fact, mostly primitive constraints). For an irreducible constraint, there is a unique minimal (in terms of number of states) deterministic presentation, called the *Shannon cover* [11].

Let G be a labeled graph. The *adjacency matrix* $A = A_G$ is the $|V_G| \times |V_G|$ matrix whose entry $(A_G)_{u,v}$ is the number of edges from state u to state v in G. The qth power of G, denoted G^q, is the labeled graph with the same set of states as G, but one edge for each path of length q in G, labeled by the q-block generated by that path. For a constrained system S presented by a labeled graph G, the qth power of S, denoted S^q, is the constrained system presented by G^q. If A is the adjacency matrix of G, it can be shown that the adjacency matrix of G^q is A^q.

It is well known that for any constraint, there exist encoders at any rate that does not exceed the (Shannon) *capacity*, $\mathrm{cap}(S)$, of the constraint; this capacity is defined as the asymptotic growth rate of the number of sequences allowed by the constraint, i.e.,

$$\mathrm{cap}(S) = \lim_{q\rightarrow\infty} \frac{1}{q} \log(N(q))$$

where $N(q)$ is the number of q-blocks in the constrained system S. It can also be computed as $\log(\lambda)$ where λ is the largest eigenvalue (also known as the *Perron eigenvalue*) of the adjacency matrix of any deterministic (or more generally, lossless) presentation of the constraint.

The encoders and the corresponding decoders can be implemented as finite-state machines that encode/decode using state information; the encoders are called *finite-state encoders*. More precisely, for a constrained system S and a positive integer n, an (S, n)-encoder is a labeled graph $\mathcal{E}$ such that

- each state of $\mathcal{E}$ has out-degree n,
- $S(\mathcal{E}) \subseteq S$,
- $\mathcal{E}$ is lossless.

The labels of the encoder are sometimes called *output labels*. A *tagged* (S, n)*-encoder* is an (S, n)-encoder whose outgoing edges from each state are assigned distinct *input tags* from an alphabet of size n. The *rate* of a tagged (S^q, n) encoder is $\frac{\log_2(n)}{q}$. So, a rate $p : q$ encoder for S is a tagged $(S^q, 2^p)$-encoder.

This structure is perfectly adequate for encoding. However, it is desirable for the decoder to be implemented as a *sliding-block decoder*, which makes a decision on a given received codeword on the basis of a local window consisting of the codeword itself, as well as a fixed number of preceding codewords (the *memory*), and a fixed number of upcoming codewords (the *anticipation*). In this way, any symbol error at the decoder input should give rise to a limited number of errors at the decoder output. For most constraints of interest, sliding-block decodability can be achieved at any rate up to capacity.

III. Block-Type-Decodable Encoders

Although sliding-block decoders enforce some limit on error propagation, the extent of error propagation may be too large for a given application. For this reason, there has been much attention focused on *block encoders*, which encode by mapping, in a one-to-one manner, unconstrained p-blocks to constrained q-blocks. Thus, block encoders are conceptually simpler than general finite-state encoders and, more importantly, have the advantage that error propagation is limited to one block.

In many circumstances, instead of using a block encoder, it may be possible to achieve a higher rate using a finite-state encoder that is *block decodable*, that is, sliding-block decodable with zero memory and zero anticipation. The corresponding decoder limits error propagation to the same extent as a block decoder: only one block. In this paper, we do not consider a more general notion of block decodability that can sometimes be achieved at the cost of replacing finite-state encoders with look-ahead encoders [7], [6].

In general, for a given block length, determining the optimal rate of a block-decodable encoder can be very difficult. However, this problem can be considerably more tractable for the class of *deterministic* encoders; these are finite-state encoders with deterministic output labeling. While deterministic encoders do not necessarily have good error propagation properties, it is well known that for RLL(d, k) constraints, the optimal rates of deterministic and block-decodable encoders coincide for every block length [2], [3], [5].

It is easy to see that a block encoder is block decodable which in turn is deterministic. For the latter, observe that for a block-decodable encoder, the deterministic tagging of input labels forces the output labeling to be deterministic. In this paper, we shall consider all of these three classes of encoders, which we call *block-type-decodable encoders*. Our goal is to determine, for a given constraint S, a given class $\mathcal{C}$ of encoders, and a given block length q, the optimal rate of an encoder for S in class $\mathcal{C}$.

In order to quantify the optimality of block-type-decodable encoders, we need the following notations. Let u and v be any states in a labeled graph G. The *follower set* of u in G, denoted $\mathcal{F}_G(u)$, is the set of all finite words that can be generated from u in G. We shall use $\mathcal{F}_G^q(u, v)$ to denote the set of all words of length q in $\mathcal{F}_G(u)$ which end at state v. Similarly, $\mathcal{F}_G^q(u, P)$ denotes the set of all words of length q in $\mathcal{F}_G(u)$ which end at a state in the set P. The states of a labeled graph are naturally endowed with the *partial ordering* by inclusion of follower sets: $u \preceq v$ if $\mathcal{F}_G(u) \subseteq \mathcal{F}_G(v)$. We say that a set $P \subseteq V_G$ is *complete* if whenever u is in P and $u \preceq v$ then v is also in P. For a constrained system S, define $M_{\mathrm{blk}}(q)$ to be the maximum n such that there exists an (S^q, n)-block encoder. Similarly define $M_{\mathrm{blkdec}}(q)$ and $M_{\mathrm{det}}(q)$ for the block-decodable and deterministic class of encoders, respectively.

Let S be an irreducible constrained system and G be an irreducible, deterministic presentation of S. For each class $\mathcal{C}$ of block-type-decodable encoders, it can be shown that there exists an (S, n)-encoder in class $\mathcal{C}$ if and only if there exists such an encoder which is a subgraph of G (in particular, this holds when G is the Shannon cover) [11]. Thus, the problem of designing block-type-decodable encoders is equivalent to choosing a subgraph of G, in particular, a subset P of V_G, called a *set of principal states* (this terminology goes back to Franaszek [2] who used it only for the class of deterministic encoders). It follows that

$$M_{\mathrm{blk}}(q) = \max_{P \subseteq V_G} \left| \bigcap_{u \in P} \mathcal{F}_G^q(u, P) \right|$$

and

$$\begin{aligned} M_{\mathrm{det}}(q) &= \max_{P \subseteq V_G} \min_{u \in P} |\mathcal{F}_G^q(u, P)| \\ &= \max_{P \subseteq V_G} \min_{u \in P} \sum_{v \in P} (A_G^q)_{u,v}. \end{aligned}$$

We do not know of a formula for $M_{\mathrm{blkdec}}(q)$ as simple as those above, but, as with M_{blk} and M_{det}, it is a function of only an arbitrary irreducible, deterministic presentation of the constraint, such as the Shannon cover. For each class of block-type-decodable encoders, we shall refer to a subset P of states of the Shannon cover that achieves the maximum as an *optimal set of principal states*.

The *memory* of a labeled graph G is defined to be the smallest integer N such that the paths in G of length N that generate the same word all terminate at the same state. Freiman and Wyner show [4] that when G has finite memory $\leq q$, for the optimal set of principal states for block encoder, it suffices to consider sets P which are complete.

Definition 1: Let G be a graph and P be a subset of V_G. (G, P) is said to satisfy the *Franaszek condition* if the states in P can be ordered: $u_1, u_2, \ldots, u_{|P|}$ such that if $1 \leq i < j < k \leq |P|$ then

$$\mathcal{F}_G^1(u_i, P) \cap \mathcal{F}_G^1(u_k, P) \subseteq \mathcal{F}_G^1(u_j, P).$$

The following result is due to Franaszek [2]. Because of its importance, we give a complete proof here.

Proposition 1 [2]: Let S be a constrained system with a deterministic presentation G. Suppose that there exists a deterministic encoder, with a block length q, determined by a set of principal states P such that (G^q, P) satisfies the Franaszek condition. Then there exists a block-decodable encoder with the same block length, rate, and set of principal states.

Proof: First let us note that (G^q, P) satisfies the Franaszek condition if and only if there exists an ordering of the states in P: $u_1, \ldots, u_{|P|}$ such that if $i \leq j \leq k$ then

$$\mathcal{F}_G^q(u_i, P) \cap \mathcal{F}_G^q(u_k, P) \subseteq \mathcal{F}_G^q(u_j, P).$$

Suppose that there exists an (S^q, n)-deterministic encoder with a set of principal states P such that (G^q, P) satisfies the Franaszek condition. In order to show that there exists an (S^q, n)-block-decodable encoder, we shall show that we can assign consistent input tagging to a subgraph of G^q on the set of

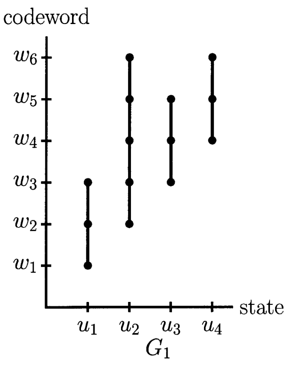

Fig. 3. The straight line condition.

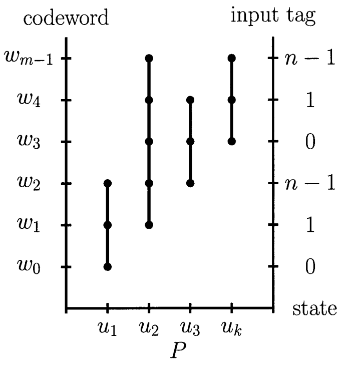

Fig. 4. Rolling input tag assignment.

states P. Formally, we show that there exists $H_i \subseteq \mathcal{F}_G^q(u_i, P)$ and a bijective mapping D_i: $H_i \rightarrow \{1, 2, \ldots, n\}$ such that

1) if a codeword $y \in H_i \cap H_j$ then $D_i(y) = D_j(y)$;
2) if $i < j$ then $H_i \cap \mathcal{F}_G^q(u_j, P) \subseteq H_j$.

We show this for each set $\{u_1, \ldots, u_m\}$, $m = 1, \ldots, |P|$. Clearly, this is true for $m = 1$. Next assume that this is true for m. Let

$$E_{m+1} = \left(\bigcup_{i=1}^{m} H_i\right) \cap \mathcal{F}_G^q(u_{m+1}, P).$$

We claim that in fact $E_{m+1} = H_m \cap \mathcal{F}_G^q(u_{m+1}, P)$. To see this, first let $y \in H_i \cap \mathcal{F}_G^q(u_{m+1}, P)$ for some $i \leq m$. By the Franaszek condition, $y \in \mathcal{F}_G^q(u_m, P)$, and, therefore, by Property 2, $y \in H_m$. Thus, $E_{m+1} = H_m \cap \mathcal{F}_G^q(u_{m+1}, P)$, as claimed.

Now, define D_{m+1} on E_{m+1} according to D_m which is injective and well defined by Property 1. Then, extend D_{m+1} by defining it on any subset of $\mathcal{F}_G^q(u_{m+1}, P) \setminus E_{m+1}$ of size $n-|E_{m+1}|$ to be a bijection onto $\{1, 2, \ldots, n\} - D_{m+1}(E_{m+1})$. This completes the construction of the bijection D_{m+1}.

What remains to be shown is that Properties 1 and 2 hold for $\{u_1, \ldots, u_{m+1}\}$. Property 1 is true because we define D_{m+1} from D_m. And if $i < m+1$ then

$$H_i \cap \mathcal{F}_G^q(u_{m+1}, P) \subseteq E_{m+1} \subseteq H_{m+1}$$

which proves Property 2. □

Next we shall give a different sufficient condition.

Definition 2: Let G be a graph and P be a subset of V_G. (G, P) is said to satisfy the *straight-line condition* if the codewords can be ordered such that for all u in P, $\mathcal{F}_G^1(u, P)$ is an interval (see Fig. 3).

We interpret Fig. 3 as saying: $\mathcal{F}_{G_1}^1(u_1) = \{w_1, w_2, w_3\}$, $\mathcal{F}_{G_2}^1(v_1) = \{z_1, z_2, z_3\}$, and so on. It can be seen that G_1 satisfies the straight-line condition, but G_2 does not (with respect to any ordering of the words). On the other hand, G_1 does not satisfy the Franaszek condition (with respect to any ordering of the states), but G_2 does.

Proposition 2: Let S be a constrained system with a deterministic presentation G. Suppose that there exists a deterministic encoder, with block length q, defined by a set of principal states P such that (G^q, P) satisfies the straight-line condition. Then there exists a block-decodable encoder with the same block length, rate, and set of principal states.

Proof: Suppose that there exists an (S^q, n)-deterministic encoder with a set of principal states P such that (G^q, P) satisfies the straight-line condition with ordering of the codewords of length q: $w_0, w_1, \ldots, w_{m-1}$. Then we assign the input tag of w_i to be $i \bmod n$ (called the *rolling assignment*; see Fig. 4), and delete excess codewords as necessary. It can be seen that for any input tag, every state can generate a codeword with that given input tag. This defines an (S^q, n)-block-decodable encoder. □

Corollary 1: Let q and n be positive integers. Suppose that for a deterministic presentation G of a constrained system S

1) G has memory at most q, and
2) (G^q, V_G) satisfies the Franaszek condition or the straight line condition.

Then there exists an (S^q, n)-block-decodable encoder if and only if there exists an (S^q, n)-deterministic encoder.

Proof: Since block decodability implies deterministic, it suffices to show that if S has a deterministic presentation G with memory at most q such that (G^q, V_G) satisfies either condition and there exists an (S^q, n)-deterministic encoder, then there exists an (S^q, n)-block-decodable encoder. This can be done by showing that if (G^q, V_G) satisfies either condition and has memory at most q, then (G^q, P) also satisfies the same condition for any $P \subseteq V_G$. Then the result follows from Propositions 1 and 2.

Let $P \subseteq V_G$ and $t^q(P)$ be the set of all words of length q that are labels of paths in G that end in P. Since the memory of G is at most q, it follows that for all states $u \in V_G$

$$\mathcal{F}_G^q(u, P) = \mathcal{F}_G^q(u, V_G) \cap t^q(P). \tag{1}$$

Suppose that (G^q, V_G) satisfies the Franaszek condition. Let $P = \{u_1, \ldots, u_{|P|}\}$ be the ordering of states in P inherited from the ordering on V_G that defines the Franaszek condition. If $i < j < k$, then by (1), we have

$$\begin{aligned}\mathcal{F}_G^q(u_i, P) \cap \mathcal{F}_G^q(u_k, P) &= \mathcal{F}_G^q(u_i, V_G) \cap \mathcal{F}_G^q(u_k, V_G) \cap t^q(P)\\ &\subseteq \mathcal{F}_G^q(u_j, V_G) \cap t^q(P)\\ &= \mathcal{F}_G^q(u_j, P)\end{aligned}$$

as desired.

Now, suppose that (G^q, V_G) satisfies the straight-line condition. Then, there is an ordering on words of length q such that for each state $u \in V_G$, $\mathcal{F}_G^q(u, V_G)$ is an interval. Restrict this

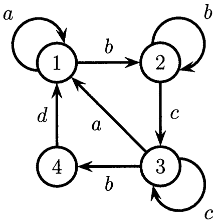

Fig. 5. An example in which block length is smaller than the memory and Corollary 1 fails.

ordering to $t^q(P)$. Then by (1), each $\mathcal{F}_G^q(u, P)$ is an interval. Intuitively, if a word w can be presented by a path that does not end in P, then the entire row corresponding to w can be removed from the straight-line diagram and the result still satisfies the straight-line condition. □

The following example shows that the conclusion of Corollary 1 can fail if Condition 1 does not hold.

Example 1: Let G be the labeled graph in Fig. 5. This graph has memory 2, and (G, V_G) satisfies both the Franaszek and straight-line conditions. But the only set of principal states which defines an $(S, 2)$-deterministic encoder is $P = \{1, 2, 3\}$, and it is easy to see that there is no consistent input tagging of this subgraph.

From Propositions 1 and 2, we can design a block-decodable encoder by first choosing a set of principal states P for a deterministic encoder: if (G^q, P) satisfies one of the conditions, then it is guaranteed that there exists a block-decodable encoder with the same rate and set of principal states. This is not such a good design criterion because, in general, the Franaszek and the straight-line conditions may hold for one P but not for another P. However, by Corollary 1, if we know that a constraint has a deterministic presentation G with memory at most q and (G^q, V_G) satisfies one of the conditions, then the existence of a deterministic encoder with codeword length q and *any* set of principal states assures the existence of a block-decodable encoder with the same block length, rate, and set of principal states.

Example 2: For the RLL(d, k) constraint with Shannon cover G, shown in Fig. 1, and all q, the Franaszek condition holds for (G^q, V_G) by virtue of the ordering: $0, 1, \ldots, d, d+1, \ldots, k$. The straight-line condition also holds: for $q > k$, see Fig. 6; for $q \leq k$, remove the codewords with prefix $0^q1, \ldots, 0^k1$ and add the word 0^q to the bottom of that diagram.

Proposition 1 was first established for RLL constraints by Franaszek [3].

Example 3: The asymmetric-RLL(d_1, k_1, d_0, k_0) constraint is the set of binary sequences whose runs of 1's have length at least d_1 and no more than k_1 and runs of 0's have length at least d_0 and no more than k_0. In the case that $d_1 = d_0 = 1, k_1 = j, k_0 = k$, this constraint coincides with the MTR(j, k) constraint (see the Shannon cover G in Fig. 2). We claim that for all q, (G^q, V_G) satisfies the Franaszek condition with the ordering $\bar{j}, \ldots, \bar{1}, 1, \ldots, k$. To see this, divide the states into two groups, $U = \{1, 2, \ldots, k\}$ and $V = \{\bar{1}, \bar{2}, \ldots, \bar{j}\}$. For any three ordered states $u_1 < u_2 < u_3$ satisfying the ordering above, at least two of them must be from the same group, say $u_1, u_2 \in U$. It is easy to see that

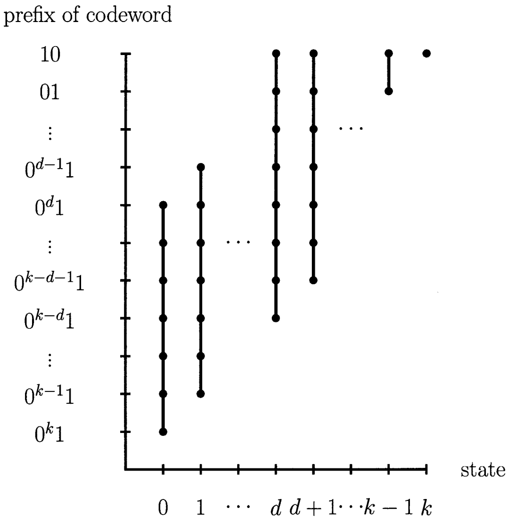

Fig. 6. RLL satisfies the straight-line condition.

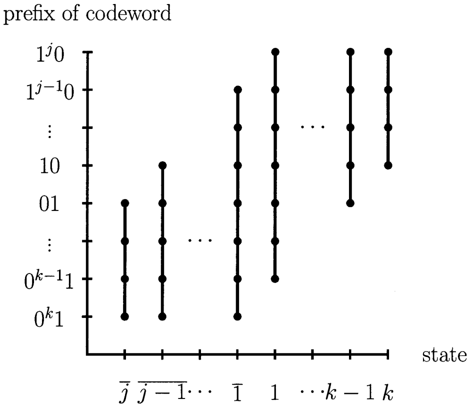

Fig. 7. MTR satisfies the straight-line condition.

$$\mathcal{F}_G^q(u_1) \cap \mathcal{F}_G^q(u_3) \subseteq \mathcal{F}_G^q(u_1) \subset \mathcal{F}_G^q(u_2)$$

hence, the Franaszek condition for (G^q, V_G) holds for all q. The straight-line condition for (G^q, V_G) also holds for all q: for $q > \max\{j, k\}$, see Fig. 7; for $q \leq \max\{j, k\}$, remove the codewords with prefix $1^q0, \ldots, 1^j0, 0^q1, \ldots, 0^k1$ and add 0^q to the bottom and/or 1^q to the top of that diagram.

In fact, for any deterministic presentation G, if the states can be divided into two groups, for each of which the corresponding follower sets are linearly ordered by inclusion, then for all q, (G^q, V_G) satisfies both the Franaszek and the straight-line conditions.

According to the examples above, the conclusion of Corollary 1 holds for RLL and MTR constraints when the block length q is greater than or equal to the memory. We claim that it holds even when the block length is smaller than the memory (i.e., $q < k$ for RLL and $q < \max\{j, k\}$ for MTR). To see this, we next verify that the straight-line condition holds for all block lengths q and all subsets P of states in the Shannon cover, and then apply Proposition 2 (we remark that this result fails for the Franaszek condition).

For the RLL constraint, as mentioned in Example 2, (G^q, V_G) satisfies the straight-line condition even for $q < k$. Next we show that for any $U \subseteq V_G$ and $u \in U$, if (G^q, U) satisfies the straight-line condition, so does $(G^q, U - \{u\})$. This implies that (G^q, P) satisfies the straight-line condition for all P. Suppose that (G^q, U) satisfies the condition and $u \in U$. To remove u, we do the following:

1) remove the column corresponding to u from the straight line diagram, this clearly preserves the straight-line condition;

2) remove all the points corresponding to edges of G^q that end at u and are labeled by a codeword $w \neq 0^q$; for the RLL constraints, any such codeword w has a unique terminal state, and so removing such a codeword ending at u means removing a whole row in the diagram, and hence the straight-line condition is preserved;

3) remove the point (if any) corresponding to 0^q codeword which ends at u; but again this does not destroy the straight-line property because the 0^q codeword is at the bottom of the diagram.

The same argument applies to the MTR constraint as 0^q and 1^q are at the bottom and the top of the diagram, respectively. Thus, we conclude that for the RLL and MTR constraints, (G^q, P) satisfies the straight-line condition for all q and P.

On the other hand, the conditions can fail for the asymmetric-RLL constraints, even for q larger than the memory; in fact, see the following example.

Example 4: Using the procedure below, one can show that for the asymmetric-RLL(1, 2, 2, 4), $M_{\text{det}}(6) = 9$. However, one can check that all the deterministic encoders that achieve M_{det} cannot be assigned consistent input tags. The best that we can achieve is $M_{\text{blkdec}}(6) = 8$.

As a final example, consider the following.

Example 5: The (d, k, s)-multiple-space-RLL constraint is the set of binary sequences whose runs of 0's satisfy the (d, k) constraint and have length equal to a multiple of s. It does not satisfy the conditions in general. But if $s = 2$ or the codeword length is large enough (say $q \geq k$) then the conditions are satisfied.

Next, we relate sets of principal states to approximate eigenvectors. Given a nonnegative square matrix A and a real number n, an (A, n)-*approximate eigenvector* is a nonnegative integer vector $x \neq 0$ such that $Ax \geq nx$ where the inequality holds componentwise. We say that x is a 0–1 *vector* if each of its entries is either 0 or 1. Let G be a graph and P be a subset of V_G; a 0–1 vector x of size $|V_G|$ is *supported* on P if $x_u = 1$ if $u \in P$ and 0 otherwise.

For the case of a deterministic encoder, let $P(q)$ denote the largest (by set inclusion) P that achieves $M_{\text{det}}(q)$. It can be shown that $P(q)$ is unique. Let $x(q)$ be the vector supported on $P(q)$. Note that for $M = M_{\text{det}}(q)$, $x(q)$ is the unique largest (entry-by-entry) 0–1 (A^q, M) approximate eigenvector.

Given integers M and q, and a matrix A, if a 0–1 (A^q, M) approximate eigenvector exists, we can find one by the well-known Franaszek algorithm [11] applied to the all-ones vector; the algorithm returns either the largest 0–1 approximate eigenvector (when M is small enough) or the zero vector (when M is too large). To find $M_{\text{det}}(q)$, we start with a small M and keep increasing it until the Franaszek algorithm returns the zero vector. Then set the last two values of M to be lower and upper bounds for $M_{\text{det}}(q)$. Once we have the bounds, we let the next M be the midpoint and apply the Franaszek algorithm. If it returns the zero vector, set M to be the next upper bound. If it returns an approximate eigenvector, set M to be the next lower bound. By repeating this process, the value of M will converge to $M_{\text{det}}(q)$. Once $M_{\text{det}}(q)$ is found, the Franaszek algorithm will return $x(q)$.

TABLE I
POSSIBLE RATES FOR MTR(j, k) CONSTRAINT WITH CODEWORD LENGTH 25

(j,k)	cap(S)	$\log_2(M_{det}(25))$
$(3,\infty)$	.9468	23.3436
$(4,4)$	.9468	23.3436
$(4,5)$	.9614	23.7492
$(4,6)$	.9684	23.9352
$(4,7)$	.9718	24.0390
$(4,\infty)$	.9752	24.1541

Example 6: We want to find a rate 24/25 MTR(j, k) block-decodable encoder, and we wish to determine possible values of (j, k). By Corollary 1, this is equivalent to finding values of (j, k) such that for $S =$ MTR(j, k) constraint, $\log_2(M_{\text{det}}(25)) \geq 24$. Some values are given in Table I. The capacities suggest that the smallest (j, k) is $(4, 5)$ because it has the capacity $0.9614 \geq 24/25$. However, for this j and k, $M_{\text{det}}(25)$ is not large enough. The smallest (j, k) which allows a rate 24/25 block-decodable encoder is $(4, 7)$. For this (j, k), we can show that $P(q)$ is $\{\overline{1}, \overline{2}, 1, 2, 3, 4\}$.

IV. Asymptotic Results for Deterministic Encoders

In order to achieve a code rate $p : q$ very close to the capacity, it is usually necessary to make codeword length q very large. Thus, it is of interest to study the asymptotic versions of $x(q)$, $P(q)$, and $M_{\text{det}}(q)$ described in the preceding section.

Let A be the adjacency matrix of an irreducible presentation of a constrained system. According to the Perron–Frobenius theorem [14], A has the following properties. It has a unique largest positive eigenvalue λ whose corresponding right and left eigenvectors, r and l, have all positive entries. In our context, r is a column vector and l is a row vector. Moreover, let c be the normalizing constant such that $c(l \cdot r) = 1$, and define $\Lambda = c(rl)$ a rank-one matrix. If A is primitive, then the Perron–Frobenius theory shows that

$$\lim_{q \to \infty} \frac{A^q}{\lambda^q} = \Lambda. \tag{2}$$

Let M^* be the largest real number n such that there exists a 0–1 (Λ, n)-approximate eigenvector. Equivalently

$$M^* = \max_{P \subseteq V_G} \min_{u \in P} \left[c r_u \left(\sum_{v \in P} l_v \right) \right]. \tag{3}$$

We let x^* be the unique, largest such approximate eigenvector and P^* be the corresponding set of principal states.

Proposition 3: For a primitive constrained system, the following hold.

1) For sufficiently large q, any 0–1 vector which achieves $M_{\text{det}}(q)$ also achieves M^*.

2)
$$M^* = \lim_{q\to\infty} \frac{M_{\text{det}}(q)}{\lambda^q}.$$

Proof: For a 0–1 vector x supported on a set P, define
$$M_q(x) = \min_{u\in P}\left(\frac{A^q}{\lambda^q}x\right)_u$$
and
$$M_\infty(x) = \min_{u\in P}(\Lambda x)_u.$$

Note that x^* maximizes $M_\infty(x)$ by definition.

Let $\overline{x}$ be a 0–1 vector which does not maximize $M_\infty(x)$, so $M_\infty(x^*) - M_\infty(\overline{x}) > 2\epsilon$ for some $\epsilon > 0$.

From (2), it follows that $M_q(x)$ converges to $M_\infty(x)$ for any x. Therefore, for q large enough
$$\begin{aligned} M_q(x^*) - M_q(\overline{x}) &> (M_\infty(x^*) - \epsilon) - (M_\infty(\overline{x}) + \epsilon) \\ &= M_\infty(x^*) - M_\infty(\overline{x}) - 2\epsilon > 0. \end{aligned}$$

This means that for large enough q, any 0–1 vector maximizing $M_q(x)$ also maximizes $M_\infty(x)$. This proves Part 1. Moreover, for any such vector x, since $M_q(x) = \frac{M_{\text{det}}(q)}{\lambda^q}$ and $M_\infty(x) = M^*$, and $M_q(x)$ converges to $M_\infty(x)$, it follows that $\frac{M_{\text{det}}(q)}{\lambda^q}$ converges to M^*, which concludes the proof. □

According to Proposition 3, for large q, $x(q)$ belongs to the set of 0–1 vectors achieving M^*; in particular, we have the following.

Corollary 2: If x^* is the unique 0–1 vector achieving M^* then $\lim_{q\to\infty} x(q) = x^*$.

The following result reduces the complexity of computing M^* and x^*.

Proposition 4: If we order the states of G according to the sizes of the entries of r in descending order $r_{u_1} \geq r_{u_2} \geq \cdots \geq r_{u_{|V_G|}}$, then
$$M^* = \max_{1\leq i\leq |V_G|} cr_{u_i}\left(\sum_{j\leq i} l_{u_j}\right).$$
And if i^* is the largest index which achieves M^*, then x^* is supported on $P^* = \{u_1, u_2, \ldots, u_{i^*}\}$.

Proof: Letting P be any set of principal states, consider the following definition which is adapted from (3):
$$M^*(P) = \min_{u\in P}\left[cr_u\left(\sum_{v\in P} l_v\right)\right]. \tag{4}$$
Then (3) becomes
$$M^* = \max_{P\subseteq V_G} M^*(P).$$
This proposition states that the maximizing P must have the form $P_i = \{u_1, u_2, \ldots, u_i\}$. To see this, let us compare $M^*(P_i)$ and $M^*(P_i \setminus \{u_k\})$ where $k < i$. Since r and l are positive vectors and r_{u_i} is the minimum among $r_{u_1}, \ldots, r_{u_i}$, we have
$$M^*(P_i) = cr_{u_i}\left(\sum_{j=1}^{i} l_j\right)$$
and
$$M^*(P_i \setminus \{u_k\}) = cr_{u_i}\left(\sum_{j=1,\, j\neq k}^{i} l_j\right).$$
Clearly, $M^*(P_i) > M^*(P_i \setminus \{u_k\})$ because l_k is positive. Removing additional states from P_i further decreases the expression. Therefore, we conclude that maximizing P must have the form $\{u_1, u_2, \ldots, u_i\}$. □

Example 7: Consider the asymmetric-RLL$(1, 3, 2, 5)$ constraint. The adjacency matrix of its Shannon cover is
$$A = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$
The corresponding right and left eigenvectors are as shown at the bottom of the page. Since $lr = 1$, the constant $c = 1$. We sort the entries of r and compute $r_{u_i}(\sum_{j\leq i} l_{u_j})$ for $1 \leq i \leq |V_G|$.

From Table II, $i^* = 6$ and $M^* = 0.7563$. Therefore, $P^* = \{5, 6, 1, 7, 4, 2\}$ and $x^* = [1\ 1\ 0\ 1\ 1\ 1\ 1\ 0]^T$.

The following proposition shows that a strong form of the converse of the first part of Proposition 3 holds for a special case where the graph has two states. However, this is not true in general (see Example 9).

Proposition 5: If an irreducible constrained system has a deterministic presentation with two states, any 0–1 vector which achieves M^* also achieves $M_{\text{det}}(q)$ for all q.

Proof: Let A be the adjacency matrix of the presentation. First consider the case when A is primitive. We claim that Λ can be written as $\alpha_1 A + \alpha_0 I$ where $\alpha_1 > 0$. To see this, observe that since A is a 2×2 matrix, it follows from the Cayley–Hamilton theorem that $A^q = \beta_1(q)A + \beta_0(q)I$ where $\beta_1(q) > 0$. Since A is primitive, $\frac{A^q}{\lambda^q} \to \Lambda$. Therefore, $\frac{\beta_1(q)}{\lambda^q}$ and $\frac{\beta_0(q)}{\lambda^q}$ converge to some real numbers α_1 and α_0 as desired.

It can be easily shown that $A^q x \geq Mx$ if and only if $\Lambda x \geq Cx$ where
$$C = \frac{\alpha_1(M - \beta_0(q))}{\beta_1(q)} + \alpha_0.$$

$$r = [0.3719\quad 0.3020\quad 0.1875\quad 0.3069\quad 0.5025\quad 0.4507\quad 0.3659\quad 0.2272]^T$$
$$l = [0.5837\quad 0.3565\quad 0.2178\quad 0.7073\quad 0.4320\quad 0.2639\quad 0.1612\quad 0.0985].$$

TABLE II
COMPUTATION OF M^* AND x^* FOR ASYMMETRIC-RLL$(1, 3, 2, 5)$

i	1	2	3	4	5	6	7	8
state u_i	5	6	1	7	4	2	8	3
r_{u_i}	.5025	.4507	.3719	.3659	.3069	.3020	.2272	.1875
$r_{u_i}\left(\sum_{j\leq i} l_{u_j}\right)$	.2171	.3136	.4759	.5272	.6593	.7563	.5913	.5288

Since C is strictly increasing with M, any vector achieving M^* also achieves $M_{\text{det}}(q)$.

Next let A be an irreducible matrix but not primitive. Then A must have the form

$$A = \begin{bmatrix} 0 & a \\ b & 0 \end{bmatrix}$$

where a and b are positive integers. By direct computation, we find that $[1\ 1]^T$ is the only 0, 1-approximate eigenvector for Λ. This vector also achieves $M_{\text{det}}(q)$ for all q. □

It follows from the proof of Proposition 5 that for a two-state primitive graph, a 0–1 vector achieves $M_{\text{det}}(q)$ if and only if it achieves M^*. The following example demonstrates that the "only if" part of this statement as well as the assertion

$$M^* = \lim_{q\to\infty} \frac{M_{\text{det}}(q)}{\lambda^q}$$

(which holds for all primitive graphs by Proposition 3) are not true for graphs which are merely irreducible.

Example 8: Consider the following adjacency matrix:

$$A = \begin{bmatrix} 0 & 4 \\ 1 & 0 \end{bmatrix}.$$

The largest eigenvalue is 2 and the corresponding Λ is

$$\Lambda = \frac{1}{4}\begin{bmatrix} 2 & 4 \\ 1 & 2 \end{bmatrix}.$$

The resulting M^* is $3/4$ and the unique achieving vector is $[1\ 1]^T$. However,

$$A^q = \begin{cases} \begin{bmatrix} 2^q & 0 \\ 0 & 2^q \end{bmatrix}, & \text{if } q \text{ is even} \\[2ex] \begin{bmatrix} 0 & 2^{q+1} \\ 2^{q-1} & 0 \end{bmatrix}, & \text{if } q \text{ is odd.} \end{cases}$$

We can see that $[1\ 0]^T$ and $[0\ 1]^T$ achieve $M_{\text{det}}(q)$ for even q, and $[1\ 1]^T$ achieves $M_{\text{det}}(q)$ for all q. Moreover

$$\frac{M_{\text{det}}(q)}{\lambda^q} = \begin{cases} 1, & \text{if } q \text{ is even} \\ \frac{1}{2}, & \text{if } q \text{ is odd.} \end{cases}$$

Hence, $\lim_{q\to\infty} \frac{M_{\text{det}}(q)}{\lambda^q}$ does not exist.

The following example shows that even in the primitive case, $x(q)$ can fail to stabilize; in fact, in this example, there is no single 0–1 approximate eigenvector x that achieves $M_{\text{det}}(q)$ for all sufficiently large q.

Example 9: Consider the following adjacency matrix:

$$A = \begin{bmatrix} 10 & 17 & 10 \\ 8 & 1 & 8 \\ 3 & 3 & 3 \end{bmatrix}.$$

The right and left eigenvectors are $r = [4\ 2\ 1]^T$ and $l = [1\ 1\ 1]$. Therefore,

$$\Lambda = \frac{1}{7}\begin{bmatrix} 4 & 4 & 4 \\ 2 & 2 & 2 \\ 1 & 1 & 1 \end{bmatrix}$$

and $M^* = 4/7$. The only 0–1 approximate eigenvectors that achieve M^* are $x_1 = [1\ 0\ 0]^T$ and $x_2 = [1\ 1\ 0]^T$. From Proposition 3, these are the only vectors that can achieve $M_{\text{det}}(q)$ for large q.

A can be diagonalized as

$$A = \frac{1}{7}\begin{bmatrix} 4 & -1 & -1 \\ 2 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}\begin{bmatrix} 21 & 0 & 0 \\ 0 & -7 & 0 \\ 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} 1 & 1 & 1 \\ -2 & 5 & -2 \\ -1 & -1 & 6 \end{bmatrix}.$$

Therefore,

$$A^q = \frac{1}{7}\begin{bmatrix} 4\cdot 21^q+2(-7)^q & 4\cdot 21^q-5(-7)^q & 4\cdot 21^q+2(-7)^q \\ 2\cdot 21^q-2(-7)^q & 2\cdot 21^q+5(-7)^q & 2\cdot 21^q-2(-7)^q \\ 21^q & 21^q & 21^q \end{bmatrix}.$$

From this, it is straightforward to compute

$$A^q x_1 = \frac{1}{7}\begin{bmatrix} 4\cdot 21^q + 2(-7)^q \\ 2\cdot 21^q - 2(-7)^q \\ 21^q \end{bmatrix}$$

$$A^q x_2 = \frac{1}{7}\begin{bmatrix} 8\cdot 21^q - 3(-7)^q \\ 4\cdot 21^q + 3(-7)^q \\ 2\cdot 21^q \end{bmatrix}.$$

We can see from the preceding expression for A^q that for all q, there is a unique vector $x(q)$ that achieves $M_{\text{det}}(q)$. When q is odd, $x(q) = x_1$ and $M_{\text{det}}(q) = \frac{4}{7}21^q + \frac{2}{7}(-7)^q$. When q is even, $x(q) = x_2$ and $M_{\text{det}}(q) = \frac{4}{7}21^q + \frac{3}{7}(-7)^q$.

In Section V, we will see that the pathological behavior of this example does not occur for RLL and MTR constraints: in particular, in those cases $x(q)$ does stabilize to x^*.

Example 10: Empirically, it seems that the convergence in Part 2 of Proposition 3 is very fast, and so M^* can be very useful for estimating $M_{\text{det}}(q)$ and the rate of the optimal deterministic code for a given codeword length. For a primitive constraint and large codeword length

$$\begin{aligned} \frac{M_{\text{det}}(q)}{\lambda^q} &\approx M^* \\ M_{\text{det}}(q) &\approx M^*\lambda^q \qquad (5) \\ \log(M_{\text{det}}(q)) &\approx \log(M^*) + q\log(\lambda) \end{aligned}$$

TABLE III
APPROXIMATION OF THE NUMBER OF CODEWORDS AND THE CODE RATE FOR RLL$(2, 7)$

q	$M_{det}(q)$	$M^*\lambda^q$	$\frac{\lfloor\log_2(M_{det}(q))\rfloor}{q}$	$\frac{\lfloor\log_2(M^*)+q\mathrm{cap}(S)\rfloor}{q}$
6	6	5.79	0.3333	0.3333
8	12	11.86	0.3750	0.3750
10	24	24.30	0.4000	0.4000
12	50	49.78	0.4167	0.4167
15	146	145.98	0.4667	0.4667
20	877	877.03	0.4500	0.4500
25	5269	5269.05	0.4800	0.4800
30	31656	31655.65	0.5000	0.5000
40	1142586	1142585.96	0.5000	0.5000

$$\frac{\lfloor\log(M_{\det}(q))\rfloor}{q} \approx \frac{\lfloor\log(M^*) + q\,\mathrm{cap}(S)\rfloor}{q}. \tag{6}$$

Equations (5) and (6) can be used to estimate $M_{\det}(q)$ and the code rate, respectively. In particular, the rate approaches the capacity as $\mathrm{cap}(S) + \frac{\log(M^*)}{q}$. Note that $M^* \le 1$ (because the largest eigenvalue of Λ is 1), and so $\log(M^*)$ is nonpositive. Table III shows this for RLL$(2, 7)$.

V. Asymptotic Results on RLL and MTR Constraints

For a particular constrained system, the asymptotically optimal set of principal states for deterministic encoders can be computed systematically based on Proposition 4. But, in fact, some families of constrained systems have enough structure that we can completely and explicitly characterize the optimal set of principal states analytically, for all members of the family simultaneously. In Propositions 6 and 7, we show that this holds for two specific families: the RLL(d, k) and MTR(j, k) constraints. For RLL, this result can be derived as a consequence of a result of Gu and Fuja [5], who show that, in fact, the asymptotically optimal set of principal states is optimal for all finite q. However, our proof gives an algebraic perspective on (albeit a weaker version of) the Gu–Fuja result. It turns out that for MTR(j, k) constraints, the asymptotically optimal set of principal states is not optimal for all q (see Example 11).

Note that by virtue of Corollary 1, the results in this section also characterize the asymptotically optimal sets of principal states for block-decodable encoders. However, these state sets generally differ from those for asymptotically optimal block encoders [10], [12].

Proposition 6: For an RLL(d, k) constraint

$$P^* = \begin{cases} \{0, 1, \ldots, \lceil \frac{k}{2} \rceil\}, & \text{if } d = 0 \\ \{0, 1, \ldots, k-1\}, & \text{if } 1 \le d \le k-2 \\ \{0, 1, \ldots, k\}, & \text{if } d = k-1. \end{cases}$$

Proof: Suppose we have the ordering of

$$r_u : r_{n_0} \ge r_{n_1} \ge \cdots \ge r_{n_k}.$$

Define

$$M_\infty(r_{n_i}) = c r_{n_i} \left(\sum_{j \le i} l_{n_j} \right).$$

By Proposition 4, it is enough to determine P^* by comparing all $M_\infty(r_{u_i})$.

From the adjacency matrix A of the RLL(d, k) constraint, we can find r and l explicitly in terms of λ

$$r_i = \begin{cases} \lambda^{k+i+1}, & \text{if } 0 \le i \le d \\ \lambda^k + \cdots + \lambda^i, & \text{if } d \le i \le k \end{cases} \tag{7}$$

and $l_i = \lambda^{k-i}$. This can be verified as follows:

$$(Ar)_i = \begin{cases} r_{i+1} = \lambda^{k+i+2} = \lambda r_i, & \text{if } 0 \le i \le d-1 \\ r_0 + r_{i+1} = \lambda^{k+1} + \lambda^k & \\ \quad + \cdots + \lambda^{i+1} = \lambda r_i, & \text{if } d \le i \le k-1 \\ r_0 = \lambda^{k+1} = \lambda r_i, & \text{if } i = k \end{cases}$$

$$(lA)_i = \begin{cases} l_d + \cdots + l_k = \lambda^{k-d} & \\ \quad + \cdots + 1 = \lambda^{k+1} = \lambda l_i, & \text{if } i = 0 \\ l_{i-1} = \lambda^{k-i+1} = \lambda l_i, & \text{if } 1 \le i \le k, \end{cases}$$

since $\lambda^{k+1} = \lambda^{k-d} + \cdots + 1$ is the characteristic equation of A.

First consider the case $d \ge 1$. We shall show that r_0 maximizes $M_\infty(r_{n_i})$. For this, we first observe that $r_k = \lambda^k$ is the smallest entry and $r_0 = \lambda^{k+1}$ is the second smallest entry of r. The former is obvious from (7). For the latter, we need only verify that $\lambda^k + \lambda^{k-1} > \lambda^{k+1}$. But this is an immediate consequence of the fact that $x + 1 = x^2$ is the characteristic equation of the adjacency matrix for the Shannon cover of the RLL$(1, \infty)$ constraint.

Now, we have

$$M_\infty(r_0) = c\lambda^{k+1}(\lambda^k + \cdots + \lambda)$$
$$M_\infty(r_k) = c\lambda^k(\lambda^k + \cdots + 1)$$

then

$$M_\infty(r_0) - M_\infty(r_k) = c\lambda^k(\lambda^{k+1} - \lambda - 1) \ge 0$$

with equality if and only if $d = k - 1$ (because $x^{k+1} - x - 1$ is the characteristic polynomial of the RLL$(k-1, k)$ constraint).

Next we shall compare $M_\infty(r_0)$ with $M_\infty(r_{n_i})$ for $n_i \ne 0, k$. Since r_k and r_0 are the two smallest entries and r_d is clearly the largest entry of r, we can assume that $r_0 < r_{n_i} \le r_d$, and so $\lambda^{k+j} < r_{n_i} \le \lambda^{k+j+1}$ for some $1 \le j \le d$. It follows that

$$\begin{aligned} M_\infty(r_{n_i}) &\le c\lambda^{k+j+1}((\lambda^k + \cdots + 1) \\ &\qquad - (\lambda^k + \cdots + \lambda^{k-j+1} + 1)) \\ &= c\lambda^{k+j+1}(\lambda^{k-j} + \cdots + \lambda) \\ &= c\lambda^{k+1}(\lambda^k + \cdots + \lambda^{j+1}) < M_\infty(r_0). \end{aligned}$$

Therefore, r_0 maximizes $M_\infty(r_{n_i})$. So, $P^* = \{0, 1, \ldots, k-1\}$ if $d \le k-2$, and $P^* = \{0, 1, \ldots, k\}$ if $d = k-1$.

We have only the case $d = 0$ remaining. In this case

$$r_0 > r_1 > \cdots > r_k$$

and

$$M_\infty(r_i) = (\lambda^k + \cdots + \lambda^i)(\lambda^k + \cdots + \lambda^{k-i}).$$

It is not hard to show that this quantity is maximized at $i = \lceil \frac{k}{2} \rceil$ and $i = \lfloor \frac{k}{2} \rfloor$, and so $P^* = \{0, 1, \ldots, \lceil \frac{k}{2} \rceil\}$. □

The reader should note that in the foregoing result we did not need to establish the entire ordering of entries of r. Furthermore, the maximizing sets of principal states are not unique for $d = 0$ and $d = k - 1$. From Proposition 3, for large q, $P(q)$ must be one of these maximizing sets. Thus, the above proposition does not uniquely determine $P(q)$. However, it can be shown that all maximizing sets give the same code rate for all q, and so in particular $\lim_{q\to\infty} x(q) = x^*$.

Case 1 ($d = 0$): From the proof of Proposition 6, the two possible choices for $P(q)$ (large q) are $P_1 = \{0, 1, \dots, \lfloor \frac{k}{2} \rfloor\}$ and $P_2 = \{0, 1, \dots, \lceil \frac{k}{2} \rceil\}$. The number of codewords corresponding to P_1 is the number of codewords starting at state $\lfloor \frac{k}{2} \rfloor$ and ending at any state in P_1. These codewords can begin with 0 to $\lceil \frac{k}{2} \rceil$ zeros and end with 0 to $\lfloor \frac{k}{2} \rfloor$ zeros. On the other hand, the number of codewords corresponding to P_2 is the number of codewords beginning with 0 to $\lfloor \frac{k}{2} \rfloor$ zeros and ending with 0 to $\lceil \frac{k}{2} \rceil$ zeros. These two numbers of codewords are equal; hence, P_1 and P_2 produce the same code rate.

Case 2 ($d = k - 1$): Again from the proof of Proposition 6, the two possibilities for $P(q)$ (large q) are $P_1 = \{0, 1, \dots, k-1\}$ and $P_2 = \{0, 1, \dots, k\}$. The ambiguity arises because

$$r_0(l_0 + \cdots + l_{k-1}) = r_k(l_0 + \cdots + l_k).$$

Therefore, we must show that $A^q_{0,0} + \cdots + A^q_{0,k-1}$ is equal to $A^q_{k,0} + \cdots + A^q_{k,k}$. Since $d = k - 1$, we can derive from the adjacency matrix that

$$A^q_{0,0} = A^{q-1}_{0,k-1} + A^{q-1}_{0,k}$$
$$A^q_{0,u} = A^{q-1}_{0,u-1}, \qquad \text{for } u = 1, 2, \dots, k-1$$

and

$$A^q_{k,u} = A^{q-1}_{0,u}, \qquad \text{for all } u.$$

Thus,

$$\begin{aligned} A^q_{0,0} + A^q_{0,1} + \cdots + A^q_{0,k-1} &= \left(A^{q-1}_{0,k-1} + A^{q-1}_{0,k}\right) \\ &\quad + A^{q-1}_{0,0} + \cdots + A^{q-1}_{0,k-2} \\ &= A^{q-1}_{0,0} + \cdots + A^{q-1}_{0,k} \\ &= A^q_{k,0} + \cdots + A^q_{k,k}. \end{aligned}$$

Proposition 7: For MTR(j, k) constraint with $j \le k$, if $k \ge j + \lfloor \frac{j}{2} \rfloor + 1$ and $(j, k) \ne (1, 2)$ then

$$P^* = \left\{ \bar{1}, \bar{2}, \dots, \overline{\left\lceil \frac{j}{2} \right\rceil}, 1, 2, \dots, k - \left\lfloor \frac{j}{2} \right\rfloor - 1 \right\} \qquad (8)$$

if $j < k \le j + \lfloor \frac{j}{2} \rfloor$ or $(j, k) = (1, 2)$ then

$$P^* = \left\{ \bar{1}, \bar{2}, \dots, \overline{\left\lceil \frac{j}{2} \right\rceil}, 1, 2, \dots, k - \left\lfloor \frac{j}{2} \right\rfloor \right\} \qquad (9)$$

and if $j = k$ then

$$P^* = \left\{ \bar{1}, \bar{2}, \dots, \overline{\left\lfloor \frac{j}{2} \right\rfloor + 1}, 1, 2, \dots, \left\lfloor \frac{k}{2} \right\rfloor + 1 \right\}. \qquad (10)$$

And for $j \ge k$, due to the symmetry, reverse the roles of j and k in the above expression.

Outline of Proof: We only give an outline of proof here (since the proposition can be deduced from a stronger result stated and proved in Section VI-B). The proof follows a similar approach to that of Proposition 6.

First assume that $j \le k$. Let λ be the largest eigenvalue of the adjacency matrix of the graph in Fig. 2. We can show the following properties of λ

1) $\lambda^{j+k} = (\lambda^{j-1} + \cdots + 1)(\lambda^{k-1} + \cdots + 1)$
2) $\lambda^j \ge \lambda^{j-1} + \cdots + 1$ and $\lambda^k \le \lambda^{k-1} + \cdots + 1$, with equality if and only if $j = k$.

The first property is in fact the characteristic equation of the adjacency matrix and the second property follows from the first property and the assumption $j \le k$.

With the states named as in Fig. 2, it can be shown that

$$\begin{aligned} r_i &= \frac{(\lambda^k - \lambda^{i-1})(\lambda^j - 1)}{(\lambda - 1)^2}, && \text{for } 1 \le i \le k \\ r_{\bar{i}} &= \frac{\lambda^k(\lambda^j - \lambda^{i-1})}{\lambda - 1}, && \text{for } 1 \le i \le j \\ l_i &= \lambda^{j+k-i+1}, && \text{for } 1 \le i \le k \\ l_{\bar{i}} &= \frac{\lambda^{j-i+1}(\lambda^k - 1)}{\lambda - 1}, && \text{for } 1 \le i \le j. \end{aligned}$$

With these expressions and the properties of λ mentioned above, we can prove the following ordering of the entries of r:

$$\begin{aligned} &r_k \le r_{\bar{j}} < r_{k-1} \le r_{\overline{j-1}} < \\ &\qquad \cdots < r_{k-j+1} \le r_{\bar{1}} < r_{k-j} < r_{k-j-1} < \cdots < r_2 < r_1 \end{aligned}$$

with equality if and only if $j = k$.

Case 1 ($j < k$): We prove that $r_{k-\lfloor \frac{j}{2} \rfloor}$ maximizes $M_\infty(r_i)$ for $k - j \le i \le k$ by computing the difference $M_\infty(r_{i+1}) - M_\infty(r_i)$. We show that this difference is decreasing in i, negative for $k - \lfloor \frac{j}{2} \rfloor \le i \le k - 1$, and positive for $k - j \le i \le k - \lfloor \frac{j}{2} \rfloor - 1$. It then follows that $r_{k-\lfloor \frac{j}{2} \rfloor}$ maximizes $M_\infty(r_i)$ for $k - j \le i \le k$ as desired.

By using a similar approach, it can be shown that $r_{\overline{\lceil \frac{j}{2} \rceil}}$ maximizes $M_\infty(r_{\bar{i}})$ for $1 \le i \le j$ and $r_{\lceil \frac{k}{2} \rceil}$ maximizes $M_\infty(r_i)$ for $1 \le i \le k - j$ (provided that $1 \le \lceil \frac{k}{2} \rceil \le k - j$; otherwise, the maximizing state is $k - j$). Therefore, we have three candidates for the maximal $M_\infty(r_u)$; namely,

$$u = \left\lceil \frac{j}{2} \right\rceil, k - \left\lfloor \frac{j}{2} \right\rfloor, \overline{\left\lceil \frac{j}{2} \right\rceil}.$$

We compare these three and it turns out that $M_\infty(r_{\overline{\lceil \frac{j}{2} \rceil}})$ is largest when $k \ge j + \lfloor \frac{j}{2} \rfloor + 1$. And $M_\infty(r_{k-\lfloor \frac{j}{2} \rfloor})$ is largest when $j < k \le j + \lfloor \frac{j}{2} \rfloor$.

Case 2 ($j = k$): In this case, $r_i = r_{\bar{i}}$. Similar to the case $d = 0$ in Proposition 6, we can show that $r_{\lceil \frac{k}{2} \rceil}$ and $r_{\lfloor \frac{k}{2} \rfloor + 1}$ maximizes $M_\infty(r_u)$.

The exceptional case $(j, k) = (1, 2)$ arises from the fact that both (8) and (9) achieve M^* in this case. □

Similar to Proposition 6, there are some cases that the maximizing set of principal states is not unique. These comprise $(j, k) = (1, 2)$ and $j = k$. MTR$(1, 2)$ is the same as RLL$(1, 2)$ (see Case 2 after Proposition 6). We shall see in Section VII that MTR(k, k) is obtained after precoding RLL$(0, k-1)$; thus, we can apply Case 1: $d = 0$ following Proposition 6.

Finally, we mention that, in contrast to RLL constraints, for MTR constraints and finite q, even of size at least the memory, the maximizing set for M^* need not be a maximizing set for $M_{\mathrm{det}}(q)$.

Example 11: For the MTR$(2, 4)$ constraint, one can show that $P(4) = \{\bar{1}, 1, 2, 3\}$ and $M_{\mathrm{det}}(4) = 8$. This is different from $P^* = \{\bar{1}, 1, 2\}$ given above, which is the unique maximizing set for M^*, but achieves $M = 7$ for $q = 4$.

VI. Finite Results for MTR Constraints

In this section, we shift our interest to the case of finite codeword length. For completeness, we state Lee's result [9], [10] on optimal block encoders for RLL constraint in Proposition 8. Gu and Fuja's result [5] on optimal block-decodable encoders for RLL constraints is described in the previous section and is stated again in Proposition 9. Theorems 1 and 2 consider optimal block encoders and block-decodable encoders for MTR constraints, respectively. Recall that for RLL and MTR constraints, optimal block-decodable encoders are equivalent to optimal deterministic encoders in terms of optimal code rate and supporting set of principal states for any given block length. Therefore, Proposition 9 and Theorem 2 apply to deterministic encoders as well. In fact, we prove Theorem 2 from the deterministic encoders point of view.

Proposition 8 [9], [10]: The following are optimal complete sets of principal states for RLL(d, k)-block encoders and all codeword lengths:

$$\left\{i,\, i+1,\, \ldots,\, i+\left\lfloor\frac{k-d}{2}\right\rfloor\right\}$$

and

$$\left\{i,\, i+1,\, \ldots,\, i+\left\lceil\frac{k-d}{2}\right\rceil\right\}$$

where $i \in \{0, 1, \ldots, d\}$.

Proposition 9 [5], [15]: The following are optimal sets of principal states for RLL(d, k)-block-decodable encoders and all codeword lengths:

$$\begin{array}{ll} \left\{0,\, 1,\, \ldots,\, \left\lfloor\frac{k}{2}\right\rfloor\right\} \text{ and } \left\{0,\, 1,\, \ldots,\, \left\lceil\frac{k}{2}\right\rceil\right\}, & \text{if } d = 0 \\ \{0,\, 1,\, \ldots,\, k-1\}, & \text{if } 1 \le d \le k-2 \\ \{0,\, 1,\, \ldots,\, k-1\} \text{ and } \{0,\, 1,\, \ldots,\, k\}, & \text{if } d = k-1. \end{array}$$

We remark that for RLL$(0, k)$

$$M_{\mathrm{blk}}(q) = M_{\mathrm{blkdec}}(q) = M_{\mathrm{det}}(q)$$

because the partial ordering of states in the Shannon cover is actually a total ordering: $0 \succeq 1 \succeq \cdots \succeq k$.

For the MTR(j, k) constraint, we shall use combinatorial methods to enumerate and compare the number of valid sequences for each choice of set of principal states. As usual, $A^q_{1,\,1}$ denotes the $(1, 1)$ element of A^q. In particular, $A^0_{1,\,1} = 1$; we take $A^q_{1,\,1}$ to be 0 for $q < 0$. Therefore, $A^q_{1,\,1} \ge 0$ for all q.

The following lemma is a useful tool in establishing the results of this section.

Lemma 1: Let $d,\, e,\, f$ be nonnegative integers such that $d \le e \le f$ and $d \le f - e$. Then

$$\sum_{a=1}^{e}\sum_{b=1}^{f-e} A^{q-a-b}_{1,\,1} - \sum_{a=1}^{d}\sum_{b=1}^{f-d} A^{q-a-b}_{1,\,1} = \sum_{a=1}^{e-d}\sum_{b=d+1}^{f-e} A^{q-a-b}_{1,\,1}.$$

Proof:

$$\begin{aligned} &\sum_{a=1}^{e}\sum_{b=1}^{f-e} A^{q-a-b}_{1,\,1} - \sum_{a=1}^{d}\sum_{b=1}^{f-d} A^{q-a-b}_{1,\,1} \\ &\overset{\text{(a)}}{=} \sum_{a=1}^{d}\sum_{b=1}^{f-e} A^{q-a-b}_{1,\,1} + \sum_{a=d+1}^{e}\sum_{b=1}^{f-e} A^{q-a-b}_{1,\,1} \\ &\quad - \sum_{a=1}^{d}\sum_{b=1}^{f-e} A^{q-a-b}_{1,\,1} - \sum_{a=1}^{d}\sum_{b=f-e+1}^{f-d} A^{q-a-b}_{1,\,1} \\ &\overset{\text{(b)}}{=} \sum_{a=1}^{e-d}\sum_{b=d+1}^{f-e+d} A^{q-a-b}_{1,\,1} - \sum_{a=f-e+1}^{f-e+d}\sum_{b=1}^{e-d} A^{q-a-b}_{1,\,1} \\ &= \sum_{a=1}^{e-d}\sum_{b=d+1}^{f-e} A^{q-a-b}_{1,\,1}. \end{aligned}$$

(a) Break each term in the left-hand side into two terms.

(b) The first and the third terms in the last equation cancel; change index of the remaining terms. □

A. Optimal Block Encoder

The following result corrects part of a statement made in [12, p. 1863].

Theorem 1: The following are optimal sets of principal states for MTR(j, k)-block encoders and all codeword lengths:

$$\left\{\bar{1},\, \bar{2},\, \ldots,\, \overline{\left\lceil\frac{j}{2}\right\rceil},\, 1,\, 2,\, \ldots,\, \left\lfloor\frac{k}{2}\right\rfloor\right\}$$

and

$$\left\{\bar{1},\, \bar{2},\, \ldots,\, \overline{\left\lfloor\frac{j}{2}\right\rfloor},\, 1,\, 2,\, \ldots,\, \left\lceil\frac{k}{2}\right\rceil\right\}.$$

Proof: Since we have the partial ordering of the states

$$\bar{1} \succeq \bar{2} \succeq \cdots \succeq \bar{j} \quad \text{and} \quad 1 \succeq 2 \succeq \cdots \succeq k$$

we claim that we can assume that the optimal complete set has the form $\{\bar{1}, \bar{2}, \ldots, \bar{s}, 1, 2, \ldots, t\}$ where $0 \le s \le j$ and $0 \le t \le k$. For $q \ge$ memory, this follows from the result of Frieman and Wyner [4] cited in Section III. For $q <$ memory, this still holds provided that for each codeword w all paths in the Shannon cover that are labeled by w end in the same state. The only words for which this may not hold are 0^q and 1^q, neither of which can be codewords in a block encoder for MTR(j, k).

For a complete set of this form, a valid sequence begins with 1 to $k - t$ zeros or 1 to $j - s$ ones and ends with 1 to t zeros or 1 to s ones. Define $B_{a,\,b}$ to be the number of valid sequences of length q beginning with a zeros and ending with b zeros and let $C_{a,\,b}$ denote the number of valid sequences of length q beginning with a ones and ending with b ones. Moreover, from the state diagram, we see that the number of valid sequences of length q starting with a zeros and ending with b ones is $A^{q-a-b}_{\bar{1},\,\bar{1}}$.

Similarly, the number of valid sequences of length q starting with a ones and ending with b zeros is $A_{1,1}^{q-a-b}$.

We claim that $A_{\overline{1},\overline{1}}^{q} = A_{1,1}^{q}$ for any q. To see this, consider a sequence starting from state $\overline{1}$ and ending at state $\overline{1}$. This sequence must pass through state 1 at least once. If we cyclically shift the first phase from $\overline{1}$ to 1, we get a sequence starting from state 1 and ending at state 1. So there is a 1–1 relationship between the cycles at state 1 and the cycles at state $\overline{1}$.

Using the above facts, the number of valid sequences of length q corresponding to the complete set

$$\{\overline{1}, \overline{2}, \ldots, \overline{s}, 1, 2, \ldots, t\}$$

is

$$\sum_{a=1}^{k-t}\sum_{b=1}^{t} B_{a,b} + \sum_{a=1}^{j-s}\sum_{b=1}^{s} C_{a,b} + \sum_{a=1}^{k-t}\sum_{b=1}^{s} A_{1,1}^{q-a-b} + \sum_{a=1}^{j-s}\sum_{b=1}^{t} A_{1,1}^{q-a-b}. \quad (11)$$

In order to compare this number for different values of s and t, we transform $B_{a,b}$ and $C_{a,b}$ into sums of the (1, 1)-entry of powers of A. For a valid sequence of length q starting with a zeros and ending with b zeros, the first run of ones can have length 1 to j. Therefore,

$$B_{a,b} = \sum_{c=1}^{j} A_{1,1}^{q-a-b-c}.$$

Similarly

$$C_{a,b} = \sum_{c=1}^{k} A_{1,1}^{q-a-b-c}.$$

Thus, (11) becomes

$$\sum_{a=1}^{k-t}\sum_{b=1}^{t}\sum_{c=1}^{j} A_{1,1}^{q-a-b-c} + \sum_{a=1}^{j-s}\sum_{b=1}^{s}\sum_{c=1}^{k} A_{1,1}^{q-a-b-c} + \sum_{a=1}^{k-t}\sum_{b=1}^{s} A_{1,1}^{q-a-b} + \sum_{a=1}^{j-s}\sum_{b=1}^{t} A_{1,1}^{q-a-b}.$$

Let us denote this by $\phi(s, t)$. Our goal is to find s and t that maximize ϕ. First, we fix s and see how ϕ changes when we change t to $t+1$.

Case 1: $t \le \frac{k}{2} - 1$

$$\begin{aligned}
&\phi(s, t+1) - \phi(s, t) \\
&\overset{(a)}{=} \sum_{a=1}^{1}\sum_{b=t+1}^{k-t-1}\sum_{c=1}^{j} A_{1,1}^{q-a-b-c} - \sum_{b=1}^{s} A_{1,1}^{q-k+t-b} \\
&\quad + \sum_{a=1}^{j-s} A_{1,1}^{q-a-t-1} \\
&\overset{(b)}{\ge} \sum_{a=t+3}^{k-t+j} A_{1,1}^{q-a} - \sum_{b=k-t+1}^{k-t+s} A_{1,1}^{q-b} \\
&\overset{(c)}{\ge} 0.
\end{aligned}$$

(a) The first term comes from Lemma 1 using $d = t$, $e = t+1$, and $f = k$.

(b) Neglect the last term of the previous equation; index of the first term runs from $t+3$ to $k-t+j$.

(c) Recall that in this case $t \le \frac{k}{2} - 1$, hence $t+3 \le k-t+1$.

Case 2: $t \ge \frac{k}{2}$

$$\begin{aligned}
&\phi(s, t+1) - \phi(s, t) \\
&\overset{(a)}{=} -\sum_{a=1}^{1}\sum_{b=k-t}^{t}\sum_{c=1}^{j} A_{1,1}^{q-a-b-c} - \sum_{b=1}^{s} A_{1,1}^{q-k+t-b} \\
&\quad + \sum_{a=1}^{j-s} A_{1,1}^{q-a-t-1} \\
&\overset{(b)}{\le} -\sum_{a=k-t+2}^{j+t+1} A_{1,1}^{q-a} + \sum_{a=t+2}^{j-s+t+1} A_{1,1}^{q-a} \\
&\overset{(c)}{\le} 0.
\end{aligned}$$

(a) Use Lemma 1 with $d = k-t-1$, $e = k-t$, and $f = k$.

(b) Neglect the second term of the previous equation; index of the first term is from $k-t+2$ to $j+t+1$.

(c) In this case $t \ge \frac{k}{2}$ therefore, $k-t+2 \le t+2$.

Case 3: $t = \frac{k-1}{2}$

$$\phi(s, t+1) - \phi(s, t) = -\sum_{b=1}^{s} A_{1,1}^{q-\frac{k+1}{2}-b} + \sum_{a=1}^{j-s} A_{1,1}^{q-\frac{k+1}{2}-a}. \quad (12)$$

We can see that the maximum number of valid sequences is obtained when $t = \frac{k}{2}$ if k is even. If k is odd, the optimal t will be either $\frac{k-1}{2}$, or $\frac{k+1}{2}$, or both depending on the sign of (12). The same analysis applies when we fix t and vary s. So we are left with four possible choices of (s, t): $(\lfloor\frac{k}{2}\rfloor, \lfloor\frac{j}{2}\rfloor)$, $(\lceil\frac{k}{2}\rceil, \lceil\frac{j}{2}\rceil)$, $(\lfloor\frac{k}{2}\rfloor, \lceil\frac{j}{2}\rceil)$, and $(\lceil\frac{k}{2}\rceil, \lfloor\frac{j}{2}\rfloor)$. If both j and k are even, then all of these are the same. If only one of them, say j, is even, then the optimal s is $\frac{j}{2}$ and (12) will take value zero. Therefore, $t = \lfloor\frac{k}{2}\rfloor$ and $t = \lceil\frac{k}{2}\rceil$ will both be optimal. Finally, consider the case when both j and k are odd. If we pick $s = \lfloor\frac{j}{2}\rfloor$ then (12) will take positive value and we must choose $t = \lceil\frac{k}{2}\rceil$. On the other hand, if we pick $s = \lceil\frac{j}{2}\rceil$, we must choose $t = \lfloor\frac{k}{2}\rfloor$. These two choices give the same number of valid sequences as can be verified by substituting into (11). □

B. Optimal Block-Decodable Encoder

For MTR constraints, $M_{\text{blkdec}}(q) = M_{\text{det}}(q)$ for all q. Therefore, we shall prove the optimality for deterministic encoders and apply the result to block-decodable encoders.

For simplicity, we assume that $j < k$ throughout this section. We shall treat the case $j = k$ in Section VII (Corollary 5).

We know that there exists (S^q, n) deterministic encoder if and only if there exists a 0–1 approximate eigenvector v

$$A^q v \ge nv$$

and the states corresponding to $v_i = 1$ are the principal states. Our goal is to find a set of principal states that gives the largest n for each value of j, k, and q. We do this assuming $q \ge k+2$.

Note that an optimal set of principal states must be of the form

$$P(s, t) = \{\overline{1}, \ldots, \overline{s}, 1, \ldots, t\}$$

where $0 \le s \le j$ and $0 \le t \le k$; and the state that determines the number of codewords n is either $\overline{s}$ or t. This follows from the ordering of the states, $\overline{1} \succeq \overline{2} \succeq \cdots \succeq \overline{j}$ and $1 \succeq 2 \succeq \cdots \succeq k$.

Let $\psi(s, t)$ be the number of codewords generated from state t and ending at a state in $P(s, t)$. These codewords must start with 0 to $k-t$ zeros and end with 1 to t zeros or 1 to s ones. By using the same method as in Section VI-A, we find that

$$\psi(s,\,t) = \sum_{a=0}^{k-t}\sum_{b=1}^{t}\sum_{c=1}^{j} A_{1,\,1}^{q-a-b-c} + \sum_{a=0}^{k-t}\sum_{b=1}^{s} A_{1,\,1}^{q-a-b}. \quad (13)$$

Similarly, define $\omega(s,\,t)$ be the number of codewords generated from state $\overline{s}$ and ending at a state in $P(s,\,t)$. The codewords must start with 0 to $j-s$ ones and end with 1 to s ones or 1 to t zeros. Therefore,

$$\omega(s,\,t) = \sum_{a=0}^{j-s}\sum_{b=1}^{s}\sum_{c=1}^{k} A_{1,\,1}^{q-a-b-c} + \sum_{a=0}^{j-s}\sum_{b=1}^{t} A_{1,\,1}^{q-a-b}. \quad (14)$$

Hence,

$$M_{\text{det}}(q) = \max_{s,\,t} \min\{\psi(s,\,t),\, \omega(s,\,t)\}.$$

Theorem 2: The following are optimal sets of principal states for MTR$(j,\,k)$-block-decodable encoders, $j<k$:

$$P_1 = \left\{\overline{1},\,\ldots,\,\overline{\left\lceil \frac{j}{2}\right\rceil},\,1,\,\ldots,\,k-\left\lfloor\frac{j}{2}\right\rfloor - 1\right\}, \qquad \text{if } k \ge j + \left\lfloor\frac{j}{2}\right\rfloor + 1$$

$$P_2 = \left\{\overline{1},\,\ldots,\,\overline{\left\lceil \frac{j}{2}\right\rceil},\,1,\,\ldots,\,k-\left\lfloor\frac{j}{2}\right\rfloor\right\}, \qquad \text{if } j < k \le j + \left\lfloor\frac{j}{2}\right\rfloor$$

for $q \ge k+2$.

By modifying the proof of Theorem 2, we can get a characterization for shorter block lengths as well. We state this without proof.

Let P_1 and P_2 be as defined in Theorem 2. For $q \le k+1$, if

1) $k \ge j + \left\lfloor \frac{j}{2}\right\rfloor + 1$ and $q \ge j+3$—or

2) $k - q \ge \left\lfloor \frac{j}{2}\right\rfloor$

then an optimal set of principal states is P_1; otherwise, an optimal set of principal states is P_2.

Before we give a formal proof of Theorem 2, let us describe the idea of how to find optimal s and t and state the required lemmas. Consider the following fact.

Fact 1: Suppose X is a set and ψ and ω are functions from X to $\mathbb{R}$. Let $M(x) = \min\{\psi(x),\,\omega(x)\}$ for $x \in X$. Suppose that $x^* \in X$ satisfies:

- $\psi(x^*) \le \omega(x^*)$
- x^* maximizes ψ.

Then x^* maximizes M and $\max_x M(x) = \psi(x^*)$.

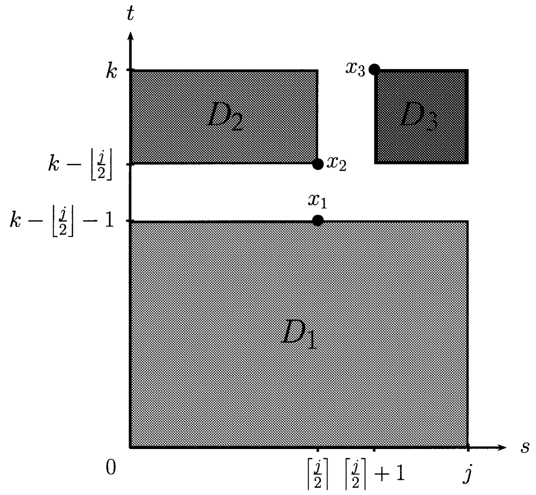
t
k
x_3
D_2
D_3
$k-\left\lfloor\frac{j}{2}\right\rfloor$
x_2
x_1
$k-\left\lfloor\frac{j}{2}\right\rfloor-1$
D_1
0
$\left\lceil\frac{j}{2}\right\rceil$
$\left\lceil\frac{j}{2}\right\rceil+1$
j
s

Fig. 8. s and t plane.

Fig. 8 shows the plane of s and t. Each point in the plane corresponds to a set of principal states $P(s,\,t)$. The whole plane is divided into three disjoint regions D_1, D_2, and D_3. We shall show that x_1 maximizes ω in D_1, x_2 maximizes ψ in D_2, and x_3 maximizes ω in D_3 (Lemma 8). Next we show that $\omega(x_1) \le \psi(x_1)$ (Lemma 4). Thus, it follows from Fact 1 that x_1 maximizes M_{det} in region D_1 and $M_{\text{det}} = \omega(x_1)$. Similarly, we show that $\omega(x_2) \ge \psi(x_2)$ (Lemma 5). Therefore, x_2 maximizes M_{det} in region D_2 and $M_{\text{det}} = \psi(x_2)$. Then we move on by showing that $\omega(x_3) \le \psi(x_2)$ (Lemma 6). This implies that M_{det} in region D_3 cannot be greater than M_{det} in region D_2. Finally, we compare $M_{\text{det}}(x_1)$ and $M_{\text{det}}(x_2)$ (Lemma 7).

Lemmas 2 and 3 are basic tools which will be used for proving subsequent lemmas.

Lemma 2:

$$A_{1,\,1}^{q} = \sum_{a=1}^{j}\sum_{b=1}^{k} A_{1,\,1}^{q-a-b} \quad (15)$$

for $q \ne 0$, and

$$A_{1,\,1}^{q} \ge \sum_{a=1}^{j}\sum_{b=1}^{k} A_{1,\,1}^{q-a-b} \quad (16)$$

for all q.

Proof: Let $q > 0$. The labeling of any cycle at state 1 of length q must have the form

$$\underbrace{00\cdots0}_{b}\underbrace{11\cdots1}_{a}0\,\underbrace{xx\cdots x}_{q-a-b-1}$$

where $0 \le b \le k-1$, $1 \le a \le j$, and $xx\cdots x$ denotes a cycle at state 1 of length $q-a-b-1 \ge 0$. Therefore, according to our convention for $A_{1,\,1}^{q}$, $q \le 0$, the total number of cycles of length q is

$$A_{1,\,1}^{q} = \sum_{a=1}^{j}\sum_{b=0}^{k-1} A_{1,\,1}^{q-a-b-1} = \sum_{a=1}^{j}\sum_{b=1}^{k} A_{1,\,1}^{q-a-b}.$$

This establishes (15) for $q > 0$. For $q < 0$, (15) is trivially true, as is (16) for $q = 0$. □

Note that, alternatively, for $q \ge j+k$, (15) follows from Cayley–Hamilton theorem applied to the characteristic equation of the adjacency matrix of the Shannon cover of the MTR$(j,\,k)$ constraint.

Lemma 3:

$$\sum_{a=1}^{j} A_{1,1}^{q-a} \le A_{1,1}^{q} \le \sum_{a=1}^{k} A_{1,1}^{q-a}$$

for $q \ge 2$.

Proof:

$$\begin{aligned} &A_{1,1}^{q} - \sum_{a=1}^{j} A_{1,1}^{q-a} \\ &\overset{(a)}{=} \sum_{a=1}^{j}\sum_{b=1}^{k} A_{1,1}^{q-a-b} - \sum_{a=1}^{j} A_{1,1}^{q-a} \\ &= \sum_{a=1}^{j}\sum_{b=2}^{k} A_{1,1}^{q-a-b} + \sum_{a=1}^{j} A_{1,1}^{q-a-1} - \sum_{a=1}^{j} A_{1,1}^{q-a} \\ &= \sum_{a=1}^{j}\sum_{b=1}^{k-1} A_{1,1}^{q-a-b-1} - A_{1,1}^{q-1} + A_{1,1}^{q-j-1} \\ &\overset{(b)}{=} \sum_{a=1}^{j}\sum_{b=1}^{k-1} A_{1,1}^{q-a-b-1} - \sum_{a=1}^{j}\sum_{b=1}^{k} A_{1,1}^{q-1-a-b} + A_{1,1}^{q-j-1} \\ &= -\sum_{a=1}^{j} A_{1,1}^{q-a-k-1} + A_{1,1}^{q-j-1} \\ &\overset{(c)}{\ge} -\sum_{a=1}^{j}\sum_{b=1}^{k} A_{1,1}^{q-j-1-a-b} + A_{1,1}^{q-j-1} \\ &\overset{(d)}{\ge} 0. \end{aligned} \tag{17}$$

(a) Use Lemma 2 (true for $q \ge 1$).

(b) Use Lemma 2 with the second term (true for $q \ge 2$).

(c) The exponents of the double sum run from $q-j-3$ to $q-2j-k-1$; while the exponents of the sum of the previous line are from $q-k-2$ to $q-j-k-1$; the inequality follows from the assumption that $j < k$.

(d) Use Lemma 2.

For the proof of the other inequality, we follow the same steps until (17)

$$A_{1,1}^{q} - \sum_{a=1}^{k} A_{1,1}^{q-a} = -\sum_{a=1}^{k} A_{1,1}^{q-a-j-1} + A_{1,1}^{q-k-1}.$$

This is clearly ≤ 0 since the exponents of the sum run from $q-j-2$ to $q-k-j-1$ which must include $q-k-1$ because of the assumption $j < k$. □

Lemmas 4–7 will be proven in the Appendix.

Lemma 4:

$$\omega\left(\left\lceil \frac{j}{2} \right\rceil, k - \left\lfloor \frac{j}{2} \right\rfloor - 1\right) \le \psi\left(\left\lceil \frac{j}{2} \right\rceil, k - \left\lfloor \frac{j}{2} \right\rfloor - 1\right)$$

for $q \ge k+1$.

Lemma 5:

$$\omega\left(\left\lceil \frac{j}{2} \right\rceil, k - \left\lfloor \frac{j}{2} \right\rfloor\right) \ge \psi\left(\left\lceil \frac{j}{2} \right\rceil, k - \left\lfloor \frac{j}{2} \right\rfloor\right)$$

for $q \ge k+2$.

Lemma 6:

$$\omega\left(\left\lceil \frac{j}{2} \right\rceil + 1, k\right) \le \psi\left(\left\lceil \frac{j}{2} \right\rceil, k - \left\lfloor \frac{j}{2} \right\rfloor\right)$$

for $q \ge 2$.

Lemma 7:

$$\begin{aligned} \omega\left(\left\lceil \frac{j}{2} \right\rceil, k - \left\lfloor \frac{j}{2} \right\rfloor - 1\right) &\le \psi\left(\left\lceil \frac{j}{2} \right\rceil, k - \left\lfloor \frac{j}{2} \right\rfloor\right), \\ &\qquad \text{if } j \le k \le j + \left\lfloor \frac{j}{2} \right\rfloor \\ \omega\left(\left\lceil \frac{j}{2} \right\rceil, k - \left\lfloor \frac{j}{2} \right\rfloor - 1\right) &\ge \psi\left(\left\lceil \frac{j}{2} \right\rceil, k - \left\lfloor \frac{j}{2} \right\rfloor\right), \\ &\qquad \text{if } k \ge j + \left\lfloor \frac{j}{2} \right\rfloor + 1 \end{aligned}$$

for $q \ge k+2$.

Lemma 8: For any fixed t, ω is nondecreasing for $0 \le s < \left\lceil \frac{j}{2} \right\rceil$ and nonincreasing for $\left\lceil \frac{j}{2} \right\rceil \le s \le j$. On the other hand, for any fixed s, ψ is nondecreasing for $0 \le t < \left\lceil \frac{k}{2} \right\rceil$ and nonincreasing for $\left\lceil \frac{k}{2} \right\rceil \le t \le k$. Hence, ω is maximized at $s = \left\lceil \frac{j}{2} \right\rceil$ and ψ is maximized at $t = \left\lceil \frac{k}{2} \right\rceil$.

Proof:

$$\begin{aligned} &\omega(s+1, t) - \omega(s, t) \\ &\quad = \sum_{c=0}^{k-1}\left[\sum_{a=1}^{j-s}\sum_{b=1}^{s+1} A_{1,1}^{q-a-b-c} - \sum_{a=1}^{j-s+1}\sum_{b=1}^{s} A_{1,1}^{q-a-b-c}\right] \\ &\quad\quad - \sum_{b=1}^{t} A_{1,1}^{q-j+s-b}. \end{aligned}$$

If $s \le (j-1)/2$ then use Lemma 1 with $d = s$, $e = s+1$, $f = j+1$.

$$\begin{aligned} &\omega(s+1, t) - \omega(s, t) \\ &\quad = \sum_{c=0}^{k-1}\left[\sum_{a=1}^{1}\sum_{b=s+1}^{j-s} A_{1,1}^{q-a-b-c}\right] - \sum_{b=j-s+1}^{j-s+t} A_{1,1}^{q-b} \\ &\quad \overset{(a)}{\ge} \sum_{a=s+2}^{j-s+k} A_{1,1}^{q-a} - \sum_{b=j-s+1}^{j-s+t} A_{1,1}^{q-b} \\ &\quad \ge 0. \end{aligned}$$

(a) The exponents of the first term in the previous equation are from $q-s-2$ to $q-j+s-k$.

If $s = j/2$ then

$$\omega(s+1, t) - \omega(s, t) = -\sum_{b=1}^{t} A_{1,1}^{q-\frac{j}{2}-b} \le 0.$$

If $s \ge (j+1)/2$ then use Lemma 1 with $d = j-s$, $e = j-s+1$, $f = j+1$

$$\begin{aligned} &\omega(s+1, t) - \omega(s, t) \\ &\quad = -\sum_{c=0}^{k-1}\left[\sum_{a=1}^{1}\sum_{b=j-s+1}^{s} A_{1,1}^{q-a-b-c}\right] - \sum_{b=1}^{t} A_{1,1}^{q-j+s-b} \\ &\quad \le 0. \end{aligned}$$

The case of ψ can be proved similarly. □

Proof of Theorem 2: Let $D = \{(s, t)\colon 0 \le s \le j, 0 \le t \le k\}$ be the domain of ψ and ω. We break D into three regions $D = D_1 \cup D_2 \cup D_3$ where

$$\begin{aligned} D_1 &= \left\{(s, t) \in D\colon t \le k - \left\lfloor \frac{j}{2} \right\rfloor - 1\right\} \\ D_2 &= \left\{(s, t) \in D\colon s \le \left\lceil \frac{j}{2} \right\rceil, t \ge k - \left\lfloor \frac{j}{2} \right\rfloor\right\} \\ D_3 &= \left\{(s, t) \in D\colon s \ge \left\lceil \frac{j}{2} \right\rceil + 1, t \ge k - \left\lfloor \frac{j}{2} \right\rfloor\right\}. \end{aligned}$$

It is easy to see that D_1, D_2, and D_3 are disjoint. We shall apply Fact 1 to regions D_1 and D_2.

First note that for any fixed s, ω is increasing in t (see (14)). This fact, together with Lemma 8 imply that $(s_1, t_1) = (\lceil \frac{j}{2} \rceil, k - \lfloor \frac{j}{2} \rfloor - 1)$ maximizes ω in D_1. Moreover, we have $\omega(s_1, t_1) \le \psi(s_1, t_1)$ from Lemma 4. Therefore, (s_1, t_1) maximizes $M_{\rm det}$ in region D_1 from Fact 1.

Similarly, ψ is increasing in s (see (13)). Thus, we have from Lemma 8 that $(s_2, t_2) = (\lceil \frac{j}{2} \rceil, k - \lfloor \frac{j}{2} \rfloor)$ maximizes ψ in D_2. Lemma 5 states that $\omega(s_2, t_2) \ge \psi(s_2, t_2)$; from Fact 1 we have (s_2, t_2) maximizes $M_{\rm det}$ in region D_2.

Again from Lemma 8 and the fact that ω is increasing in t, ω is maximized at $(s_3, t_3) = (\lceil \frac{j}{2} \rceil + 1, k)$ in region D_3. Therefore, in this region, $M_{\rm det} \le \omega(s_3, t_3)$. However, we know from Lemma 6 that $\omega(s_3, t_3) \le \psi(s_2, t_2)$. This means that $M_{\rm det}$ in D_3 is not larger than $M_{\rm det}$ in D_2. Therefore, we have only two candidates for optimal (s, t): (s_1, t_1) and (s_2, t_2). Finally, Lemma 7 compares these two choices and we have that (s_1, t_1) is optimal when $k \ge j + \lfloor \frac{j}{2} \rfloor + 1$ and (s_2, t_2) is optimal when $j < k \le j + \lfloor \frac{j}{2} \rfloor$. □

The case $q \le k + 1$ can be proved by first showing that (s_1, t_1) maximizes $M_{\rm det}$ in region D_1 and (s_2, t_2) maximizes $M_{\rm det}$ in region D_2 for all q. Then for each value of j, k, and q, we compare $M_{\rm det}$ obtained from (s_1, t_1) and (s_2, t_2) by modifying Lemma 7.

VII. Optimality and Precoding

In magnetic recording, binary data is often represented in two ways: nonreturn to zero inverted (NRZI) and nonreturn to zero (NRZ). We usually use $\{0, 1\}$ and $\{+, -\}$ as alphabets for NRZI and NRZ, respectively. In the NRZ domain, $+$ and $-$ represent the two directions of magnetic polarity. In the NRZI domain, 1 represents a transition from one polarity to the other and 0 represents no transition. Codes are usually designed in the NRZI domain and then $1/(1 \oplus D)$-precoded to NRZ before recording. For more detail on precoding, the reader is referred to [16], [11]. In this section, we shall investigate the effect of the precoder on the optimality (code rate and $M(q)$) of block-type-decodable encoders. We shall use the symbol S to denote a constraint in NRZI and $\tilde{S}$ to denote its NRZ precoded version. Furthermore, let $M_C(S, q)$ denote the maximum number of codewords of length q satisfying constraint S which can be generated from class C encoder, where C can be $\rm blk$, $\rm blkdec$, or $\rm det$.

Let A be the adjacency matrix of the Shannon cover G of a binary constraint S. We can write $A = A_0 + A_1$ where A_0 contains the edges with output label 0 and A_1 contains the edges with output label 1. We now show that there is a deterministic presentation $\tilde{G}$ of $\tilde{S}$ with an adjacency matrix

$$\tilde{A} = \begin{bmatrix} A_0 & A_1 \\ A_1 & A_0 \end{bmatrix} \tag{18}$$

(although $\tilde{G}$ need not be the Shannon cover of $\tilde{S}$). To see this, for each $u \in V_G$, define u^+ and u^- to be counterparts of u where the last symbol of the codeword ending at u^+ must be $+$ and the last symbol of the codeword ending at u^- must be $-$. Let us define $U^+ = \{u^+ | u \in V_G\}$ and $U^- = \{u^- | u \in V_G\}$. We construct a presentation $\tilde{G}$ of $\tilde{S}$ with $V_{\tilde{G}} = U^+ \cup U^-$. With the adjacency matrix $\tilde{A}$ in (18), we think of the first $|V_G|$ states of $\tilde{G}$ as U^+ and the other half as U^-. The edges corresponding to the top left A_0 and the bottom left A_1 have label $+$ and the others have label $-$. Suppose there is an edge with label 0 from u to v in G. Then there is an edge with label $+$ from u^+ to v^+ and another edge with label $-$ from u^- to v^- in $\tilde{G}$. On the other hand, if there is an edge with label 1 from u to v in G, then there is an edge with label $+$ from u^- to v^+ and another edge with label $-$ from u^+ to v^- in $\tilde{G}$.

Proposition 10: Let S be an irreducible constraint. Then

$$M_{\rm det}(S, q) = M_{\rm det}(\tilde{S}, q).$$

Proof: First we claim that

$$\tilde{A}^q = \begin{bmatrix} B_q & C_q \\ C_q & B_q \end{bmatrix}$$

where B_q and C_q are $|V_G| \times |V_G|$ matrices such that $B_q + C_q = A^q$. We prove the claim by induction on q. Clearly, this is true for $q = 1$. Suppose it is true for q then

$$\begin{aligned} \tilde{A}^{q+1} &= \begin{bmatrix} B_q & C_q \\ C_q & B_q \end{bmatrix} \begin{bmatrix} A_0 & A_1 \\ A_1 & A_0 \end{bmatrix} \\ &= \begin{bmatrix} B_q A_0 + C_q A_1 & B_q A_1 + C_q A_0 \\ C_q A_0 + B_q A_1 & C_q A_1 + B_q A_0 \end{bmatrix}. \end{aligned}$$

Therefore, $B_{q+1} = B_q A_0 + C_q A_1$ and $C_{q+1} = B_q A_1 + C_q A_0$ and

$$\begin{aligned} B_{q+1} + C_{q+1} &= (B_q + C_q)(A_0 + A_1) \\ &= A^q A = A^{q+1}. \end{aligned}$$

This proves the claim.

Next we shall show that $M_{\rm det}(S, q) \le M_{\rm det}(\tilde{S}, q)$. Let x be a 0–1 vector such that

$$A^q x \ge M_{\rm det}(S, q) x.$$

Let

$$y = \begin{bmatrix} x \\ x \end{bmatrix}.$$

Then

$$\begin{aligned} \tilde{A}^q y &= \begin{bmatrix} B_q & C_q \\ C_q & B_q \end{bmatrix} \begin{bmatrix} x \\ x \end{bmatrix} \\ &= \begin{bmatrix} B_q x + C_q x \\ C_q x + B_q x \end{bmatrix} \\ &= \begin{bmatrix} A^q x \\ A^q x \end{bmatrix} \ge M_{\rm det}(S, q) \begin{bmatrix} x \\ x \end{bmatrix} = M_{\rm det}(S, q) y. \end{aligned}$$

Therefore,

$$M_{\rm det}(\tilde{S}, q) \geq M_{\rm det}(S, q).$$

Finally, we shall show that $M_{\rm det}(S, q) \geq M_{\rm det}(\tilde{S}, q)$. Let x_1 and x_2 be 0–1 vectors of equal length such that

$$\tilde{A}^q \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \geq M_{\rm det}(\tilde{S}, q) \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}.$$

Define $y = \max\{x_1, x_2\}$ componentwise. Then

$$A^q y = B_q y + C_q y \geq B_q x_1 + C_q x_2 \geq M_{\rm det}(\tilde{S}, q) x_1$$

and

$$A^q y = C_q y + B_q y \geq C_q x_1 + B_q x_2 \geq M_{\rm det}(\tilde{S}, q) x_2.$$

Thus, $A^q y \geq M_{\rm det}(\tilde{S}, q) y$. This implies that

$$M_{\rm det}(S, q) \geq M_{\rm det}(\tilde{S}, q). \qquad \square$$

The following corollary follows immediately from the proof of the above proposition.

Corollary 3: If x is a 0–1 approximate eigenvector of A^q, then $[x^T \; x^T]^T$ is a 0–1 approximate eigenvector of $\tilde{A}^q$ with the same approximate eigenvalue.

Compared with Proposition 10, we do not have such a nice result for block and block-decodable encoders. Here are some examples showing that the code rate may be smaller or larger when designed in the NRZ domain.

- S = MTR$(2, 5)$, $q = 6$. $M_{\rm blkdec}(S, q) = 28$, and $M_{\rm blkdec}(\tilde{S}, q) < 28$. (This is an example where we have strict inequality in Corollary 4 below.)
- $S = S(G_1)$, $q = 2$ (see Fig. 9). $M_{\rm blkdec}(S, q) = 1$, and $M_{\rm blkdec}(\tilde{S}, q) = 2$.
- S=RLL$(1, 2)$, q=6. $M_{\rm blk}(S, q)$=2, and $M_{\rm blk}(\tilde{S}, q)$=1.
- $S = S(G_2)$, $q = 2$ (see Fig. 10). $M_{\rm blk}(S, q) = 2$, and $M_{\rm blk}(\tilde{S}, q) = 3$.

However, if we restrict to constraints S that satisfy either the Franaszek condition or the straight-line condition, then we have the following result on block-decodable encoder.

Corollary 4: If a constraint S satisfies either the Franaszek condition or the straight-line condition, then

$$M_{\rm blkdec}(S, q) \geq M_{\rm blkdec}(\tilde{S}, q).$$

Proof:

$$\begin{aligned} &M_{\rm blkdec}(S, q) \\ &\quad = M_{\rm det}(S, q) \quad \text{(Franaszek or straight-line condition)} \\ &\quad = M_{\rm det}(\tilde{S}, q) \quad \text{(Proposition 10)} \\ &\quad \geq M_{\rm blkdec}(\tilde{S}, q). \end{aligned} \qquad \square$$

Note that this is in contrast to Immink's result [7], [8] in which he shows that the code rate may be improved by designing the code in NRZ domain. This is because he considers a more general type of encoder which involves a nondeterministic input tag assignment and a positive-delay encoding).

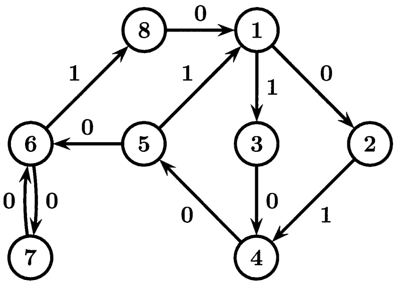

Fig. 9. G_1.

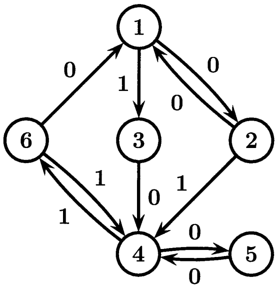

Fig. 10. G_2.

Corollary 5: The following are optimal sets of principal states for MTR(k, k)-block-decodable encoders and all block lengths:

$$\left\{\bar{1}, \ldots, \overline{\left\lceil \frac{k}{2} \right\rceil}, 1, \ldots, \left\lceil \frac{k}{2} \right\rceil\right\}$$
$$\left\{\bar{1}, \ldots, \overline{\left\lfloor \frac{k}{2} \right\rfloor + 1}, 1, \ldots, \left\lfloor \frac{k}{2} \right\rfloor + 1\right\}.$$

Proof: MTR(k, k) is the precoding of RLL$(0, k-1)$. Gu and Fuja [5] assert that optimal sets of principal states for RLL$(0, k-1)$ are

$$\left\{0, 1, \ldots, \left\lceil \frac{k-1}{2} \right\rceil\right\} \quad \text{and} \quad \left\{0, 1, \ldots, \left\lfloor \frac{k-1}{2} \right\rfloor\right\}.$$

Since both RLL(S) and MTR$(\tilde{S})$ satisfy the Franaszek condition, Corollary 3 can be used to find the set of principal states for block-decodable encoder for $\tilde{S}$ as well. The corresponding sets of principals states for MTR(k, k) are

$$\left\{\bar{1}, \ldots, \overline{\left\lceil \frac{k+1}{2} \right\rceil}, 1, \ldots, \left\lceil \frac{k+1}{2} \right\rceil\right\}$$

and

$$\left\{\bar{1}, \ldots, \overline{\left\lfloor \frac{k+1}{2} \right\rfloor}, 1, \ldots, \left\lfloor \frac{k+1}{2} \right\rfloor\right\}. \qquad \square$$

Another precoding of interest is that of *signed NRZI* in which the output alphabet is $\mathcal{A} = \{-1, 0, 1\}$; the signed NRZI version $\hat{S}$ of a binary constraint S is the set of all words $w = w_1 \cdots w_\ell$ over the alphabet $\mathcal{A}$ such that

1) $|w_1| \cdots |w_\ell| \in S$, and
2) successive nonzero symbols in w alternate in sign.

If A is the adjacency matrix of G, a deterministic presentation of S, one can show that the matrix $\tilde{A}$ in (18) also supports a presentation $\hat{G}$ of $\hat{S}$. The edges corresponding to the top right A_1 have label 1, the edges corresponding to the bottom left A_1 have label -1, and the others have label 0. The concept of $+$ and $-$ states applies here as well. Now u^+ and u^- are the counterparts of u where the last nonzero symbol of the codeword ending at u^+ must be 1 and the last nonzero symbol of the codeword ending at u^- must be -1.

We can show the following.

- $M_{\mathrm{det}}(\hat{S}, q) = M_{\mathrm{det}}(S, q)$. This is because the matrix $\tilde{A}$ in (18) is also the adjacency matrix of a deterministic presentation of $\hat{S}$. Thus, Proposition 10 also holds for signed NRZI precoding.

- $M_{\mathrm{blkdec}}(\hat{S}, q) \geq M_{\mathrm{blkdec}}(S, q)$. Suppose $\{u_1, \ldots, u_k\}$ is the set of principal states for an NRZI block-decodable encoder. Then one can show that $\{u_1^-, \ldots, u_k^-, u_1^+, \ldots, u_k^+\}$ gives a signed NRZI block-decodable encoder with the same block length and rate. This is because we can assign the input tag of a codeword $w = w_1 \cdots w_\ell$ to be the input tag of $|w_1| \cdots |w_\ell|$ in the NRZI encoder. There can be no contradiction in the assignments because there is only one codeword, namely, the all-zeros word, that can be generated from a state in U^- and a state in U^+. An example where we have strict inequality is $S = S(G_1)$ in Fig. 9 with $q = 2$. In this case, $M_{\mathrm{blkdec}}(S, q) = 2$ and $M_{\mathrm{blkdec}}(\hat{S}, q) = 3$.

- $M_{\mathrm{blk}}(\hat{S}, q) \leq M_{\mathrm{blk}}(S, q)$. All principal states for a signed NRZI block encoder must be in either U^- or U^+ again because the only common codeword which can be generated from a state in U^- and U^+ is the all-zeros codeword. Then suppose that $\{u_1^+, \ldots, u_k^+\}$ is the set of principal states for a signed NRZI block encoder; one can show that $\{u_1, \ldots, u_k\}$ gives an NRZI block encoder with at least the same rate. An example where strict inequality holds is RLL$(1, 3)$, $q = 6$. In this case, $M_{\mathrm{blk}}(S, q) = 5$ and $M_{\mathrm{blk}}(\hat{S}, q) = 4$.

Analogous to Corollary 4, we can show that if S satisfies the Franaszek or the straight-line condition, then $M_{\mathrm{blkdec}}(\hat{S}, q) = M_{\mathrm{blkdec}}(S, q)$. This follows from the first and second bullets above. To see this, we need only show that the reverse inequality of the second bullet holds. Indeed, it does

$$M_{\mathrm{blkdec}}(\hat{S}, q) \leq M_{\mathrm{det}}(\hat{S}, q) = M_{\mathrm{det}}(S, q) = M_{\mathrm{blkdec}}(S, q) \tag{19}$$

(in (19), the inequality holds for all constraints, the first equality follows from the first bullet, and the second equality follows from the Franaszek or straight-line condition).

APPENDIX

Proof of Lemma 4:

$$\begin{aligned}
&\omega\left(\left\lceil \frac{j}{2} \right\rceil, k - \left\lfloor \frac{j}{2} \right\rfloor - 1\right)\\
&= \sum_{a=0}^{\lfloor \frac{j}{2} \rfloor} \sum_{b=1}^{\lceil \frac{j}{2} \rceil} \sum_{c=1}^{k} A_{1,1}^{q-a-b-c} + \sum_{a=0}^{\lfloor \frac{j}{2} \rfloor} \sum_{b=1}^{k-\lfloor \frac{j}{2} \rfloor -1} A_{1,1}^{q-a-b}\\
&= \sum_{b=1}^{\lceil \frac{j}{2} \rceil} \sum_{c=1}^{k} A_{1,1}^{q-b-c} + \sum_{a=1}^{\lfloor \frac{j}{2} \rfloor} \sum_{b=1}^{\lceil \frac{j}{2} \rceil} \sum_{c=1}^{k} A_{1,1}^{q-a-b-c}\\
&\quad + \sum_{b=1}^{k-\lfloor \frac{j}{2} \rfloor -1} A_{1,1}^{q-b} + \sum_{a=1}^{\lfloor \frac{j}{2} \rfloor} \sum_{b=1}^{k-\lfloor \frac{j}{2} \rfloor -1} A_{1,1}^{q-a-b}\\
&\overset{\text{(a)}}{=} \sum_{b=1}^{\lceil \frac{j}{2} \rceil} \sum_{c=1}^{k} A_{1,1}^{q-b-c} + \sum_{a=1}^{\lfloor \frac{j}{2} \rfloor} \sum_{b=1}^{\lceil \frac{j}{2} \rceil} \sum_{c=1}^{k} A_{1,1}^{q-a-b-c}\\
&\quad + \sum_{a=1}^{k-\lfloor \frac{j}{2} \rfloor -1} \sum_{b=1}^{j} \sum_{c=1}^{k} A_{1,1}^{q-a-b-c} + \sum_{a=1}^{\lfloor \frac{j}{2} \rfloor} \sum_{b=1}^{k-\lfloor \frac{j}{2} \rfloor -1} A_{1,1}^{q-a-b}.
\end{aligned} \tag{20}$$

(a) Use Lemma 2 with the third term (true for $q \geq k - \lfloor \frac{j}{2} \rfloor$).

$$\begin{aligned}
&\psi\left(\left\lceil \frac{j}{2} \right\rceil, k - \left\lfloor \frac{j}{2} \right\rfloor - 1\right)\\
&= \sum_{a=0}^{\lfloor \frac{j}{2} \rfloor +1} \sum_{b=1}^{k-\lfloor \frac{j}{2} \rfloor -1} \sum_{c=1}^{j} A_{1,1}^{q-a-b-c} + \sum_{a=0}^{\lfloor \frac{j}{2} \rfloor +1} \sum_{b=1}^{\lceil \frac{j}{2} \rceil} A_{1,1}^{q-a-b}\\
&= \sum_{b=1}^{k-\lfloor \frac{j}{2} \rfloor -1} \sum_{c=1}^{j} A_{1,1}^{q-b-c} + \sum_{a=1}^{\lfloor \frac{j}{2} \rfloor +1} \sum_{b=1}^{k-\lfloor \frac{j}{2} \rfloor -1} \sum_{c=1}^{j} A_{1,1}^{q-a-b-c}\\
&\quad \cdot \sum_{b=1}^{\lceil \frac{j}{2} \rceil} A_{1,1}^{q-b} + \sum_{a=1}^{\lfloor \frac{j}{2} \rfloor +1} \sum_{b=1}^{\lceil \frac{j}{2} \rceil} A_{1,1}^{q-a-b}\\
&\overset{\text{(a)}}{=} \sum_{b=1}^{k-\lfloor \frac{j}{2} \rfloor -1} \sum_{c=1}^{j} A_{1,1}^{q-b-c} + \sum_{a=1}^{\lfloor \frac{j}{2} \rfloor +1} \sum_{b=1}^{k-\lfloor \frac{j}{2} \rfloor -1} \sum_{c=1}^{j} A_{1,1}^{q-a-b-c}\\
&\quad \cdot \sum_{a=1}^{\lceil \frac{j}{2} \rceil} \sum_{b=1}^{j} \sum_{c=1}^{k} A_{1,1}^{q-a-b-c} + \sum_{a=1}^{\lfloor \frac{j}{2} \rfloor +1} \sum_{b=1}^{\lceil \frac{j}{2} \rceil} A_{1,1}^{q-a-b}.
\end{aligned} \tag{21}$$

(a) Use Lemma 2 with the third term (true for $q \geq \lceil \frac{j}{2} \rceil + 1$). From (20) and (21), we have

$$\begin{aligned}
&\omega\left(\left\lceil \frac{j}{2} \right\rceil, k - \left\lfloor \frac{j}{2} \right\rfloor - 1\right) - \psi\left(\left\lceil \frac{j}{2} \right\rceil, k - \left\lfloor \frac{j}{2} \right\rfloor - 1\right)\\
&\overset{\text{(a)}}{=} \sum_{a=1}^{\lceil \frac{j}{2} \rceil} \sum_{b=\lfloor \frac{j}{2} \rfloor +2}^{k} A_{1,1}^{q-a-b} - \sum_{a=\lfloor \frac{j}{2} \rfloor +1}^{j} \sum_{b=1}^{\lceil \frac{j}{2} \rceil} \sum_{c=1}^{k} A_{1,1}^{q-a-b-c}\\
&\quad + \sum_{a=1}^{k-\lfloor \frac{j}{2} \rfloor -1} \sum_{b=1}^{j} \sum_{c=\lfloor \frac{j}{2} \rfloor +2}^{k} A_{1,1}^{q-a-b-c}\\
&\quad - \sum_{a=\lfloor \frac{j}{2} \rfloor +1}^{j} \sum_{b=1}^{k-\lfloor \frac{j}{2} \rfloor -1} A_{1,1}^{q-a-b}\\
&\overset{\text{(b)}}{\leq} \sum_{a=\lfloor \frac{j}{2} \rfloor +1}^{j} \sum_{b=2}^{k-\lfloor \frac{j}{2} \rfloor} A_{1,1}^{q-a-b} - \sum_{a=\lfloor \frac{j}{2} \rfloor +3}^{j+\lceil \frac{j}{2} \rceil +k} A_{1,1}^{q-a}\\
&\quad + \sum_{c=\lfloor \frac{j}{2} \rfloor +2}^{k} A_{1,1}^{q-c} - \sum_{a=\lfloor \frac{j}{2} \rfloor +1}^{j} \sum_{b=1}^{k-\lfloor \frac{j}{2} \rfloor -1} A_{1,1}^{q-a-b}
\end{aligned}$$

$$
\begin{aligned}
&\overset{(c)}{=} \sum_{a=k+1}^{k+\lceil \frac{j}{2}\rceil} A_{1,1}^{q-a} - \sum_{a=\lfloor \frac{j}{2}\rfloor+2}^{j+1} A_{1,1}^{q-a}\\
&\qquad - \sum_{a=\lfloor \frac{j}{2}\rfloor+3}^{j+\lceil \frac{j}{2}\rceil+k} A_{1,1}^{q-a} + \sum_{c=\lfloor \frac{j}{2}\rfloor+2}^{k} A_{1,1}^{q-c}\\
&\overset{(d)}{=} \sum_{a=j+2}^{k} A_{1,1}^{q-a} + \sum_{a=k+1}^{k+\lceil \frac{j}{2}\rceil} A_{1,1}^{q-a} - \sum_{a=\lfloor \frac{j}{2}\rfloor+3}^{j+\lceil \frac{j}{2}\rceil+k} A_{1,1}^{q-a}\\
&\overset{(e)}{=} \sum_{a=j+2}^{k+\lceil \frac{j}{2}\rceil} A_{1,1}^{q-a} - \sum_{a=\lfloor \frac{j}{2}\rfloor+3}^{j+\lceil \frac{j}{2}\rceil+k} A_{1,1}^{q-a}\\
&\le 0.
\end{aligned}
$$

(a) The first term is the difference between the first term in (20) and the fourth term in (21), the second term is the difference between the second term in (20) and the third term in (21) the third term is the difference between the third term in (20) and the second term in (21).

(b) Change the index of the first term; use the fact that the second term has the exponents from $q-\lfloor \frac{j}{2}\rfloor-3$ to $q-j-\lceil \frac{j}{2}\rceil+k$; use Lemma 2 with the third term (true for $q \ge k+1$).

(c) The first two terms are the difference between the first and the fourth terms of the previous equation.

(d) The first term comes from the difference between the second and the fourth terms of the previous equation.

(e) Combine the first two terms of the previous equation together. □

Proof of Lemma 5:

$$
\begin{aligned}
&\omega\left(\left\lceil \frac{j}{2}\right\rceil, k-\left\lfloor \frac{j}{2}\right\rfloor\right)\\
&\quad = \sum_{a=0}^{\lfloor \frac{j}{2}\rfloor}\sum_{b=1}^{\lceil \frac{j}{2}\rceil}\sum_{c=1}^{k} A_{1,1}^{q-a-b-c} + \sum_{a=0}^{\lfloor \frac{j}{2}\rfloor}\sum_{b=1}^{k-\lfloor \frac{j}{2}\rfloor} A_{1,1}^{q-a-b} \qquad (22)\\
&\psi\left(\left\lceil \frac{j}{2}\right\rceil, k-\left\lfloor \frac{j}{2}\right\rfloor\right)\\
&\quad = \sum_{a=0}^{\lfloor \frac{j}{2}\rfloor}\sum_{b=1}^{k-\lfloor \frac{j}{2}\rfloor}\sum_{c=1}^{j} A_{1,1}^{q-a-b-c} + \sum_{a=0}^{\lfloor \frac{j}{2}\rfloor}\sum_{b=1}^{\lceil \frac{j}{2}\rceil} A_{1,1}^{q-a-b} \qquad (23)\\
&\omega\left(\left\lceil \frac{j}{2}\right\rceil, k-\left\lfloor \frac{j}{2}\right\rfloor\right) - \psi\left(\left\lceil \frac{j}{2}\right\rceil, k-\left\lfloor \frac{j}{2}\right\rfloor\right)\\
&\quad \overset{(a)}{=} \sum_{a=0}^{\lfloor \frac{j}{2}\rfloor}\left[-\sum_{b=1}^{\lfloor \frac{j}{2}\rfloor}\sum_{c=\lceil \frac{j}{2}\rceil+1}^{k-\lfloor \frac{j}{2}\rfloor} A_{1,1}^{q-a-b-c} + \sum_{b=\lceil \frac{j}{2}\rceil+1}^{k-\lfloor \frac{j}{2}\rfloor} A_{1,1}^{q-a-b}\right]\\
&\quad = \sum_{a=0}^{\lfloor \frac{j}{2}\rfloor}\sum_{b=\lceil \frac{j}{2}\rceil+1}^{k-\lfloor \frac{j}{2}\rfloor}\left[-\sum_{c=1}^{\lfloor \frac{j}{2}\rfloor} A_{1,1}^{q-a-b-c} + A_{1,1}^{q-a-b}\right]\\
&\quad \overset{(b)}{\ge} 0.
\end{aligned}
$$

(a) The first term is the difference between the first terms in (22) and (23) using Lemma 1 ($d=\lceil \frac{j}{2}\rceil$, $e=j$, $f=k+\lceil \frac{j}{2}\rceil$); the second term is the difference between the second terms in (22) and (23).

(b) Use Lemma 3 (true for $q \ge k+2$). □

Proof of Lemma 6:

$$
\begin{aligned}
&\omega\left(\left\lceil \frac{j}{2}\right\rceil+1, k\right)\\
&\quad = \sum_{a=0}^{\lfloor \frac{j}{2}\rfloor-1}\sum_{b=1}^{\lceil \frac{j}{2}\rceil+1}\sum_{c=1}^{k} A_{1,1}^{q-a-b-c} + \sum_{a=0}^{\lfloor \frac{j}{2}\rfloor-1}\sum_{b=1}^{k} A_{1,1}^{q-a-b}\\
&\quad \overset{(a)}{=} \sum_{a=0}^{\lfloor \frac{j}{2}\rfloor-1}\sum_{b=1}^{\lceil \frac{j}{2}\rceil}\sum_{c=1}^{k} A_{1,1}^{q-a-b-c} + \sum_{a=0}^{\lfloor \frac{j}{2}\rfloor-1}\sum_{c=1}^{k} A_{1,1}^{q-a-\lceil \frac{j}{2}\rceil-1-c}\\
&\qquad + \sum_{a=0}^{\lfloor \frac{j}{2}\rfloor-1}\sum_{b=1}^{\lceil \frac{j}{2}\rceil} A_{1,1}^{q-a-b} + \sum_{a=0}^{\lfloor \frac{j}{2}\rfloor-1}\sum_{b=\lceil \frac{j}{2}\rceil+1}^{k} A_{1,1}^{q-a-b}. \qquad (24)
\end{aligned}
$$

(a) Break each term of the previous equation into two terms.

$$
\begin{aligned}
&\psi\left(\left\lceil \frac{j}{2}\right\rceil, k-\left\lfloor \frac{j}{2}\right\rfloor\right)\\
&\quad = \sum_{a=0}^{\lfloor \frac{j}{2}\rfloor}\sum_{b=1}^{k-\lfloor \frac{j}{2}\rfloor}\sum_{c=1}^{j} A_{1,1}^{q-a-b-c} + \sum_{a=0}^{\lfloor \frac{j}{2}\rfloor}\sum_{b=1}^{\lceil \frac{j}{2}\rceil} A_{1,1}^{q-a-b}\\
&\quad \overset{(a)}{=} \sum_{a=0}^{\lfloor \frac{j}{2}\rfloor}\left[\sum_{b=1}^{\lceil \frac{j}{2}\rceil}\sum_{c=1}^{k} A_{1,1}^{q-a-b-c} + \sum_{b=1}^{\lfloor \frac{j}{2}\rfloor}\sum_{c=\lceil \frac{j}{2}\rceil+1}^{k-\lfloor \frac{j}{2}\rfloor} A_{1,1}^{q-a-b-c}\right]\\
&\qquad + \sum_{a=0}^{\lfloor \frac{j}{2}\rfloor}\sum_{b=1}^{\lceil \frac{j}{2}\rceil} A_{1,1}^{q-a-b}\\
&\quad \overset{(b)}{\ge} \sum_{a=0}^{\lfloor \frac{j}{2}\rfloor-1}\sum_{b=1}^{\lceil \frac{j}{2}\rceil}\sum_{c=1}^{k} A_{1,1}^{q-a-b-c} + \sum_{b=1}^{\lceil \frac{j}{2}\rceil}\sum_{c=1}^{k} A_{1,1}^{q-\lfloor \frac{j}{2}\rfloor-b-c}\\
&\qquad \cdot\sum_{a=0}^{\lfloor \frac{j}{2}\rfloor}\sum_{b=\lceil \frac{j}{2}\rceil+2}^{k} A_{1,1}^{q-a-b} + \sum_{a=0}^{\lfloor \frac{j}{2}\rfloor-1}\sum_{b=1}^{\lceil \frac{j}{2}\rceil} A_{1,1}^{q-a-b}\\
&\qquad + \sum_{b=1}^{\lceil \frac{j}{2}\rceil} A_{1,1}^{q-\lfloor \frac{j}{2}\rfloor-b}. \qquad (25)
\end{aligned}
$$

(a) Use Lemma 1 with the first term in the previous equation ($d=\lceil \frac{j}{2}\rceil$, $e=j$, $f=k+\lceil \frac{j}{2}\rceil$).

(b) Break the first and the third terms of the previous equation into two terms each; the third term comes from the fact that the exponents of the second term in the bracket in the previous equation range from $q-\lceil \frac{j}{2}\rceil-2$ to $q-k$.

$$
\begin{aligned}
&\omega\left(\left\lceil \frac{j}{2}\right\rceil+1, k\right) - \psi\left(\left\lceil \frac{j}{2}\right\rceil, k-\left\lfloor \frac{j}{2}\right\rfloor\right)\\
&\quad \overset{(a)}{\le} \sum_{c=1}^{k}\left[\sum_{a=\lceil \frac{j}{2}\rceil+1}^{j} A_{1,1}^{q-a-c} - \sum_{b=\lfloor \frac{j}{2}\rfloor+1}^{j} A_{1,1}^{q-b-c}\right]\\
&\qquad - \sum_{a=0}^{\lfloor \frac{j}{2}\rfloor}\sum_{b=\lceil \frac{j}{2}\rceil+2}^{k} A_{1,1}^{q-a-b}
\end{aligned}
$$

$$\begin{aligned}
&+\left[\sum_{a=0}^{\lfloor\frac{j}{2}\rfloor-1}\sum_{b=\lceil\frac{j}{2}\rceil+1}^{k} A_{1,1}^{q-a-b}-\sum_{b=1}^{\lceil\frac{j}{2}\rceil} A_{1,1}^{q-\lfloor\frac{j}{2}\rfloor-b}\right]\\
&\overset{(b)}{\le} -\sum_{a=0}^{\lfloor\frac{j}{2}\rfloor}\sum_{b=\lceil\frac{j}{2}\rceil+2}^{k} A_{1,1}^{q-a-b}+\left[\sum_{a=0}^{\lfloor\frac{j}{2}\rfloor-1} A_{1,1}^{q-a-\lceil\frac{j}{2}\rceil-1}\right.\\
&\quad\left.+\sum_{a=0}^{\lfloor\frac{j}{2}\rfloor-1}\sum_{j=\lceil\frac{j}{2}\rceil+2}^{k} A_{1,1}^{q-a-b}-\sum_{b=1}^{\lceil\frac{j}{2}\rceil} A_{1,1}^{q-\lfloor\frac{j}{2}\rfloor-b}\right]\\
&\overset{(c)}{\le} \sum_{a=\lceil\frac{j}{2}\rceil+1}^{j} A_{1,1}^{q-a}-\sum_{b=\lfloor\frac{j}{2}\rfloor+1}^{j} A_{1,1}^{q-b}\\
&\le 0.
\end{aligned}$$

(a) The first terms of (24) and (25) cancel; the first bracket is the difference between the second terms in (24) and (25); the second bracket is the difference between the last terms in (24) and (25).

(b) The first bracket in the previous equation is clearly less than or equal to zero; break the first term in the second bracket of the previous equation into two terms.

(c) The sum of the first and the third terms in the previous equation are obviously less than or equal to zero; change the index of the second and the fourth terms in the previous equation. □

Proof of Lemma 7:

$$\begin{aligned}
&\omega\left(\left\lceil\frac{j}{2}\right\rceil, k-\left\lfloor\frac{j}{2}\right\rfloor-1\right)\\
&\quad=\sum_{a=0}^{\lfloor\frac{j}{2}\rfloor}\sum_{b=1}^{\lceil\frac{j}{2}\rceil}\sum_{c=1}^{k} A_{1,1}^{q-a-b-c}+\sum_{a=0}^{\lfloor\frac{j}{2}\rfloor}\sum_{b=1}^{k-\lfloor\frac{j}{2}\rfloor-1} A_{1,1}^{q-a-b} \qquad (26)
\end{aligned}$$

$$\begin{aligned}
&\psi\left(\left\lceil\frac{j}{2}\right\rceil, k-\left\lfloor\frac{j}{2}\right\rfloor\right)\\
&\quad=\sum_{a=0}^{\lfloor\frac{j}{2}\rfloor}\sum_{b=1}^{k-\lfloor\frac{j}{2}\rfloor}\sum_{c=1}^{j} A_{1,1}^{q-a-b-c}+\sum_{a=0}^{\lfloor\frac{j}{2}\rfloor}\sum_{b=1}^{\lceil\frac{j}{2}\rceil} A_{1,1}^{q-a-b} \qquad (27)
\end{aligned}$$

$$\begin{aligned}
&\omega\left(\left\lceil\frac{j}{2}\right\rceil, k-\left\lfloor\frac{j}{2}\right\rfloor-1\right)-\psi\left(\left\lceil\frac{j}{2}\right\rceil, k-\left\lfloor\frac{j}{2}\right\rfloor\right)\\
&\quad\overset{(a)}{=}\sum_{a=0}^{\lfloor\frac{j}{2}\rfloor}\left[-\sum_{b=1}^{\lfloor\frac{j}{2}\rfloor}\sum_{c=\lceil\frac{j}{2}\rceil+1}^{k-\lfloor\frac{j}{2}\rfloor} A_{1,1}^{q-a-b-c}+\sum_{b=\lceil\frac{j}{2}\rceil+1}^{k-\lfloor\frac{j}{2}\rfloor-1} A_{1,1}^{q-a-b}\right]\\
&\quad=\sum_{a=\lceil\frac{j}{2}\rceil}^{j}\left[\sum_{b=1}^{k-j-1} A_{1,1}^{q-a-b}-\sum_{b=1}^{\lfloor\frac{j}{2}\rfloor}\sum_{c=1}^{k-j} A_{1,1}^{q-a-b-c}\right]. \qquad (28)
\end{aligned}$$

(a) The first term comes from using Lemma 1 with the first terms in (26) and (27) ($d=\lceil\frac{j}{2}\rceil$, $e=j$, $f=k+\lceil\frac{j}{2}\rceil$).

Next we fix j and define

$$h(k)=\sum_{b=1}^{k-j-1} A_{1,1}^{q-b}-\sum_{b=1}^{\lfloor\frac{j}{2}\rfloor}\sum_{c=1}^{k-j} A_{1,1}^{q-b-c}.$$

Consider

$$h(k+1)-h(k)=A_{1,1}^{q-k+j}-\sum_{b=1}^{\lfloor\frac{j}{2}\rfloor} A_{1,1}^{q-b-k+j-1}$$

which is greater than or equal to zero by Lemma 2. Therefore, h is a nondecreasing function of k. Hence, it is enough to prove $h(k)\ge 0$ for $k=j+\lfloor\frac{j}{2}\rfloor+1$ and $h(k)\le 0$ for $k=j+\lfloor\frac{j}{2}\rfloor$.

$$\begin{aligned}
h\left(j+\left\lfloor\frac{j}{2}\right\rfloor+1\right)&=\sum_{b=1}^{\lfloor\frac{j}{2}\rfloor} A_{1,1}^{q-b}-\sum_{b=1}^{\lfloor\frac{j}{2}\rfloor}\sum_{c=1}^{\lfloor\frac{j}{2}\rfloor+1} A_{1,1}^{q-b-c}\\
&\ge\sum_{b=1}^{\lfloor\frac{j}{2}\rfloor} A_{1,1}^{q-b}-\sum_{b=1}^{\lfloor\frac{j}{2}\rfloor}\sum_{c=1}^{j} A_{1,1}^{q-b-c}\\
&\overset{(a)}{\ge}\sum_{b=1}^{\lfloor\frac{j}{2}\rfloor} A_{1,1}^{q-b}-\sum_{b=1}^{\lfloor\frac{j}{2}\rfloor} A_{1,1}^{q-b}\\
&=0.
\end{aligned}$$

(a) Use Lemma 3 (true for $q\ge j+\lfloor\frac{j}{2}\rfloor+2=k+1$).

$$h\left(j+\left\lfloor\frac{j}{2}\right\rfloor\right)=\sum_{b=1}^{\lfloor\frac{j}{2}\rfloor-1} A_{1,1}^{q-b}-\sum_{b=1}^{\lfloor\frac{j}{2}\rfloor}\sum_{c=1}^{\lfloor\frac{j}{2}\rfloor} A_{1,1}^{q-b-c}. \qquad (29)$$

From Lemma 2, we have

$$A_{1,1}^{q-\lfloor\frac{j}{2}\rfloor}-\sum_{b=1}^{\lfloor\frac{j}{2}\rfloor}\sum_{c=1}^{k-\lfloor\frac{j}{2}\rfloor} A_{1,1}^{q-\lfloor\frac{j}{2}\rfloor-b-c}\ge 0. \qquad (30)$$

We sum (29) and (30) and we get

$$\begin{aligned}
h\left(j+\left\lfloor\frac{j}{2}\right\rfloor\right)&\le\sum_{b=1}^{\lfloor\frac{j}{2}\rfloor} A_{1,1}^{q-b}-\sum_{b=1}^{\lfloor\frac{j}{2}\rfloor}\sum_{c=1}^{k} A_{1,1}^{q-b-c}\\
&\overset{(a)}{\le}\sum_{b=1}^{\lfloor\frac{j}{2}\rfloor} A_{1,1}^{q-b}-\sum_{b=1}^{\lfloor\frac{j}{2}\rfloor} A_{1,1}^{q-b}\\
&=0.
\end{aligned}$$

(a) Use Lemma 3 (true for $q\ge j+\lfloor\frac{j}{2}\rfloor+2=k+2$). □

Acknowledgment

The authors would like to thank Mario Blaum, Jorge Campello, John Gill, Richard New, Lei Poo, and Bruce Wilson for their valuable comments and suggestions. The authors would also like to thank the IBM Almaden Research Center for hosting the seminar where parts of this work were first presented. They are also grateful to the anonymous referee for a thorough and detailed review that greatly improved the presentation of this paper.

References

[1] K. A. S. Abdel-Ghaffar and J. H. Weber, “Constrained block codes for class-iv partial-response channels with maximum likelihood sequence estimation,” *IEEE Trans. Inform. Theory,* vol. 42, pp. 1405–1424, Sept. 1996.

[2] P. A. Franaszek, “Sequence-state coding for digital transmission,” *Bell Sys. Tech. J.* , vol. 47, pp. 143–155, 1968.

[3] ——, "Sequence-state methods for run-length-limited coding," *IBM J. Res. Develop.*, vol. 14, pp. 376–383, 1970.

[4] C. Freiman and A. Wyner, "Optimum block codes for noiseless input restricted channels," *Inform. Contr.*, vol. 7, pp. 398–415, 1964.

[5] J. Gu and T. Fuja, "A new approach to constructing optimal block codes for runlength-limited channels," *IEEE Trans. Inform. Theory*, vol. 40, pp. 774–785, May 1994.

[6] H. D. L. Hollmann, "Bounded-delay-encodable, block-decodable codes for constrained systems," *IEEE Trans. Inform. Theory*, vol. 42, pp. 1957–1970, Nov. 1996.

[7] K. A. S. Immink, "Block-decodable runlength-limited codes via look-ahead technique," *Philips J. Res.*, vol. 46, pp. 293–310, 1992.

[8] ——, *Codes for Mass Data Storage Systems*. Geldrop, The Netherlands: Shannon Foundation Publishers, 1999.

[9] P. Lee, "Combined error-correcting/modulation recording codes," Ph.D. dissertation, Univ. California, San Diego, La Jolla, 1988.

[10] P. Lee and J. K. Wolf, "A general error-correcting code construction for runlength limited binary channels," *IEEE Trans. Inform. Theory*, vol. 35, pp. 1330–1335, Nov. 1989.

[11] B. H. Marcus, R. M. Roth, and P. H. Siegel, *Handbook of Coding Theory*. Amsterdam, The Netherlands: Elsevier, 1998, ch. 20.

[12] C. Menyennett and H. C. Ferreira, "Sequences and codes with asymmetrical runglength constraints," *IEEE Trans. Commun.*, vol. 40, pp. 1862–1865, May 1995.

[13] J. Moon and B. Brickner, "Maximum transition run codes for data storage systems," *IEEE Trans. Magn.*, vol. 32, pp. 3992–3994, Sept. 1996.

[14] E. Seneta, *Non-Negative Matrices and Markov Chains*. New York: Springer-Verlag, 1980.

[15] T. J. Tjalkens, "On the principal state method for run-length limited sequences," *IEEE Trans. Inform. Theory*, vol. 40, pp. 934–941, May 1994.

[16] S. Wang and A. Taratorin, *Magnetic Information Storage Technology*. San Diego, CA: Academic, 1999.