On the Decoding Delay of Encoders for Input-Constrained Channels*

Jonathan J. Ashley[†] Brian H. Marcus[†] Ron M. Roth[‡]

Abstract

Finite-state encoders that encode n-ary data into a constrained system S are considered. The anticipation, or decoding delay, of such an (S, n)-encoder is the number of symbols that a state-dependent decoder needs to look ahead in order to recover the current input symbol. Upper bounds are obtained on the smallest attainable number of states of any (S, n)-encoder with anticipation t. Those bounds can be explicitly computed from t and S, which implies that the problem of checking whether there is an (S, n)-encoder with anticipation t is decidable. It is also shown that if there is an (S, n)-encoder with anticipation t, then a version of the state-splitting algorithm can be applied to produce an (S, n) encoder with anticipation at most $2t-1$. We also observe that the problem of checking whether there is an (S, n)-encoder having a sliding-block decoder with a given memory and anticipation is decidable.

Keywords: anticipation, constrained systems, decoding delay, finite-state encoders, input-constrained channels, sliding-block decodability.

---

*This work was supported by grant No. 92-00210 from the United-States–Israel Binational Science Foundation (BSF), Jerusalem, Israel. This work was presented in part at the IEEE International Symposium on Information Theory, Trondheim, Norway, June 1994.

[†]IBM Research Division, Almaden Research Center, 650 Harry Road, San Jose, CA 95120.

[‡]Computer Science Department, Technion — Israel Institute of Technology, Haifa 32000, Israel.

1 Introduction

Input-constrained channels, also known as *constrained systems,* are widely-used models for describing the read-write requirements of secondary storage systems, such as magnetic disks or optical memory devices. A constrained system S is defined as the set of *constrained sequences* obtained from reading the labels of paths in a labeled finite directed graph G. We say that G is a *presentation* of S.

One goal in the study of constrained systems is designing encoders that map unconstrained n-ary sequences, referred to as *source sequences,* into constrained sequences of a given constrained system S. A *rate* $p : q$ *finite-state* (S, n)*-encoder* accepts an input p-block of n-ary source symbols and generates an output q-block that depends on the input block and the current internal state of the encoder. The input block and the current state determine the next state; the sequences obtained by concatenating the generated q-blocks are required to be in S.

An encoder has *finite local anticipation* if there is an integer N such that the encoder state at each time slot r, together with the q-blocks generated at times $r, r+1, \ldots, r+N$, determine uniquely the source p-block that was input at time slot r. The *(local) anticipation* of the encoder is the smallest N for which this holds. The anticipation can be viewed as the *decoding delay* of the encoder. By considering source p-blocks and constrained q-blocks to be symbols in bigger alphabets we can assume $p = q = 1$.

In Section 3, we show that if there exists an (S, n)-encoder with anticipation t, then there exists an (S, n)-encoder with anticipation at most t and with at most M states, where M is an integer that is explicitly computed from t and S. This result implies, in turn, that the problem of verifying whether there is an (S, n)-encoder with anticipation t is a decidable problem. While the upper bound, M, is linear in the number, $|V_G|$, of states of the smallest deterministic presentation G of S, it is, unfortunately, doubly exponential in t.

In Section 5, we show that if there is an (S, n)-encoder with anticipation t, then we can obtain an (S, n)-encoder with anticipation at most $2t-1$ using a variation, contained in [4], of the state-splitting algorithm [1]. The advantage of the resulting encoder is that its number of states is bounded above by an integer that is (singly) exponential in t and linear in $|V_G|$. It follows from [4],[7] that the smallest anticipation of any (S, n)-encoder is at most linear in $|V_G|$ (for other bounds on this smallest anticipation see Section 6). So, for given S and n, there is an (S, n)-encoder with anticipation at most twice the smallest possible anticipation and whose number of states is at most (singly) exponential in $|V_G|$.

As a by-product of the previous results, we obtain in Section 6 a new lower bound on the anticipation of an (S, n)-encoder for any given S and n. We then compare this lower bound with an earlier lower bound given in [20].

An encoder is (m, a)-*sliding-block decodable* if the source sequence which was input to the encoder at time r can be reconstructed by applying a decoding function on the 'window' of q-blocks at times $r-\mathsf{m}, r-\mathsf{m}+1, \ldots, r+\mathsf{a}$ in the constrained sequence. The parameter a must be nonnegative whereas the parameter m can be negative as long as $\mathsf{m}+\mathsf{a} \geq 0$. Any encoder which is (m, a)-sliding-block decodable must also have finite local anticipation — in fact, with anticipation at most a. Sliding-block decodability guarantees limited error propagation.

Using a result in [6], we show in Section 7 that checking whether there is an (S, n)-encoder that is (m, a)-sliding-block decodable is a decidable problem; given S, n, m, and a, the upper bound on the number of states in the smallest (m, a)-sliding-block decodable (S, n)-encoder is (singly) exponential in $|\mathsf{m}| + \mathsf{a}$ and linear in $|V_G|$.

The definitions and background that are used in this paper are summarized in Sections 2 and 4.

2 Definitions and background

In this section, we provide definitions and summarize some of the background that will be used throughout the paper.

2.1 Labeled graphs and constrained systems

A *labeled graph* (or a *labeled finite directed graph*) $G = (V, E, L)$ consists of —

- a finite set of states $V = V_G$;
- a finite set of edges $E = E_G$ where each edge e has an *initial state* $\sigma_G(e)$ and a *terminal state* $\tau_G(e)$, both in V;
- edge labels $L = L_G : E \to \Sigma$ drawn from a finite alphabet Σ.

We will also use the notation $u \xrightarrow{a} v$ to denote an edge labeled a from state u to state v in G. We sometimes refer to a labeled graph as simply a *graph*.

A path γ in a graph G is a finite sequence of edges $e_1 e_2 \ldots e_\ell$ such that $\sigma_G(e_{i+1}) = \tau_G(e_i)$ for $i = 1, 2, \ldots, \ell-1$. The length of a path γ is the number of edges along the path. A *cycle* in a graph is a path $e_1 e_2 \ldots e_\ell$ where $\tau_G(e_\ell) = \sigma_G(e_1)$.

A labeled graph can be used to generate finite symbol sequences, by reading off the labels along paths in the graph, thereby producing a *word* or a *string* or a *block*. The length of a word $\mathbf{w}$ will be denoted by $\ell(\mathbf{w})$. A word of length ℓ generated by G will be called an *ℓ-block*. The empty word, denoted $\boldsymbol{\epsilon}$, is generated from any state and is the only word of length 0.

We regard two labeled graphs as the same if there is a *labeled graph isomorphism* from one to the other — i.e., a mapping from states to states and edges to edges which preserves initial states, terminal states, and labels.

The fundamental object considered in the theory of constrained coding is the set of words generated by a labeled graph. A *constrained system* (or *constraint*), denoted S, is the set of all words (i.e., finite sequences) obtained from reading the labels of paths in a labeled graph G. We say that G *presents* S or is a *presentation* of S, and we write $S = S(G)$. If a path γ is labeled by a word $\mathbf{w}$, we say that γ generates $\mathbf{w}$ and write $L_G(\gamma) = \mathbf{w}$. The *alphabet* of S is the set of symbols that actually occur in words of S and is denoted $\Sigma = \Sigma(S)$.

2.2 Special presentations

A labeled graph is *deterministic* if at each state the outgoing edges are labeled distinctly. In other words, at each state, any label generated from that state uniquely determines an outgoing edge from that state. In the next paragraph, we indicate how any constrained system can be presented in this way.

Let G be a presentation of a constrained system S. We define the *determinizing graph* H of G in the following manner. For any word $\mathbf{w}$ and state $v \in V_G$, let $T_G(\mathbf{w}, v)$ denote the subset of states in G which are accessible from v by paths in G that generate $\mathbf{w}$. When $\mathbf{w}$ is the empty word $\boldsymbol{\epsilon}$, define $T_G(\boldsymbol{\epsilon}, v) = \{v\}$. The states of H are the distinct nonempty subsets $\{T_G(\mathbf{w}, v)\}_{\mathbf{w},v}$ of V_G. As for the edges of H, for any two states $Z, Z' \in V_H$ we draw an edge $Z \xrightarrow{b} Z'$ in H if and only if there exists a state $v \in V_G$ and a word $\mathbf{w}$ such that $Z = T_G(\mathbf{w}, v)$ and $Z' = T_G(\mathbf{w}b, v)$. In other words, each state of G in Z' is accessible in G from some state in Z by an edge labeled b. By construction, the determinizing graph H is deterministic, and it is not hard to show that $S(H) = S(G)$ (see [20]), and so H is a deterministic presentation of S (this construction is very closely related to the 'subset construction' of automata theory).

Encoder construction algorithms usually begin with a deterministic presentation and transform it into a presentation which satisfies the following weaker version of the deterministic property.

A labeled graph G has *finite local anticipation* (or, in short, *finite anticipation*), if there is an integer N such that any two paths of length $N+1$ with the same initial state and labeling must have the same initial edge. The *(local) anticipation* $\mathcal{A}(G)$ of G is the smallest N for which this holds. Hence, knowledge of the initial state of a path and the first $\mathcal{A}(G)+1$ symbols that it generates is sufficient information to determine the initial edge of the path. In case G does not have finite anticipation, we define $\mathcal{A}(G) = \infty$. Note that to say that a labeled graph is deterministic is to say that it has zero anticipation. Sometimes in the literature, finite local anticipation is called 'lossless of finite order.'

A labeled graph is *lossless* if any two distinct paths with the same initial state and terminal state have different labelings. A graph is called *decent* if every state has at least one outgoing edge. It is straightforward to see that any decent labeled graph with finite local anticipation is lossless. Thus, for decent graphs we have

$$\text{Deterministic} \quad\Longrightarrow\quad \text{Finite local anticipation} \quad\Longrightarrow\quad \text{Lossless.}$$

2.3 Irreducibility

A graph G is *irreducible* (or *strongly connected*), if for any ordered pair of states u, v, there is a path from u to v in G. Note that irreducibility does not depend on the labeling of the graph.

It will be useful later to know that any graph can, in some sense, be broken down into "maximal" irreducible pieces. To make this more precise we introduce the concept of an irreducible component. An *irreducible component* of a graph G is a maximal (with respect to inclusion) irreducible subgraph of G. The irreducible components of a graph are simply the subgraphs consisting of all edges whose initial and terminal states both belong to an equivalence class of the following relation: $u \sim v$ if there is a path from u to v and a path from v to u (we allow paths to be empty so that $u \sim u$).

An *irreducible sink* is an irreducible component H such that any edge which originates in H must also terminate in H. Any graph can be broken down into irreducible components with 'transient' connections between the components, and every graph has at least one irreducible sink [23].

A constrained system S is *irreducible* if for every pair of words $\mathbf{w}, \mathbf{w}'$ in S, there is a word $\mathbf{z}$ such that $\mathbf{wzw}'$ is in S. A constrained system that is not irreducible is called *reducible*.

We will make use of the following result from [20]:

Lemma 2.1 *Let S be an irreducible constrained system and let G be a labeled graph such that $S \subseteq S(G)$. Then for some irreducible component G' of G, $S \subseteq S(G')$.*

It follows from this result that a constrained system S is irreducible if and only if it can be presented by some irreducible (in fact, deterministic) labeled graph.

2.4 Shannon cover

A *Shannon cover* of a constrained system S is a deterministic presentation of S with a smallest number of states.

In general, the Shannon cover is not unique. However, it is unique, up to labeled graph isomorphism, for irreducible constrained systems. This is Theorem 2.2 below.

Let u be a state 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. Two states u and u' in a labeled graph G are said to be *follower-set equivalent*, denoted $u \simeq u'$, if they have the same follower set. It is easy to verify that follower-set equivalence satisfies the properties of an equivalence relation. A labeled graph G is called *reduced* if no two states in G are follower-set equivalent.

The following result summarizes the main properties of the Shannon cover of irreducible constrained systems. See [8],[9],[17],[18].

Theorem 2.2 *Let S be an irreducible constrained system. The Shannon cover of S is unique, up to labeled graph isomorphism. In fact, the Shannon cover is the unique presentation of S which is irreducible, deterministic, and reduced.*

If a labeled graph G presents a constrained system S, we can construct a reduced labeled graph H (called the *reduction* of G) that presents the same constrained system S by merging states in G which are follower-set equivalent. More precisely, each equivalence class C of follower-set equivalent states becomes a state in H, and we draw an edge $C \stackrel{a}{\to} C'$ in H if and only if there exists an edge $u \stackrel{a}{\to} u'$ in G for some states $u \in C$ and $u' \in C'$. It is easy to verify that, indeed, $S(H) = S(G)$, and that the reduction of an irreducible, deterministic graph is irreducible, deterministic, and reduced. Thus, the Shannon cover of an irreducible constrained system can be obtained from any deterministic presentation by first applying Lemma 2.1 to find an irreducible, deterministic presentation and then finding the reduction of this presentation.

The following lemma shows how follower sets of a deterministic presentation of a subsystem relate to follower sets of a deterministic presentation of the entire system.

Lemma 2.3 *[20] Let G and H be two irreducible deterministic graphs. Then $S(H) \subseteq S(G)$ if and only if for every $v \in V_H$ there exists $u \in V_G$ such that $\mathcal{F}_H(v) \subseteq \mathcal{F}_G(u)$.*

2.5 Some graph constructions

Let G be a labeled graph. 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.

The *Moore form* of a labeled graph G is a labeled graph H where $V_H = E_G$ and $e_1 \stackrel{a}{\to} e_2$ is an edge of H if and only if $\tau_G(e_1) = \sigma_G(e_2)$ and $L_G(e_2) = a$. The *Moore co-form* of G is defined in the same way except that the edge $e_1 \to e_2$ of the co-form inherits the labeling of e_1 rather than e_2. The Moore form and Moore co-form present the same constrained system as G.

2.6 Capacity

Let S be a constrained system and let $N(\ell; S)$ denote the number of words of length ℓ in S. The *base-α Shannon capacity*, or simply *base-α capacity* of S, is defined by

$$\mathsf{cap}_\alpha(S) \;=\; \lim_{\ell\to\infty} \frac{1}{\ell} \log_\alpha N(\ell; S) \;.$$

We omit the base notation unless we want to emphasize the specific base used.

The capacity of a constrained system is computed using linear algebra. In order to describe how to do this, we must introduce the notion of adjacency matrix of a graph as well as some material from Perron-Frobenius theory.

Let G be a graph. The *adjacency matrix* $A = A_G = [(A_G)_{u,v}]_{u,v\in V_G}$ is the $|V_G| \times |V_G|$ matrix where the entry $(A_G)_{u,v}$ is the number of edges from state u to state v in G. Note that A_G does not depend on the labeling of G.

A nonnegative real square matrix A is *irreducible* if for every row index u and column index v there exists a nonnegative integer ℓ such that $A^\ell_{u,v} > 0$. Note that a graph is irreducible if and only if its adjacency matrix is irreducible.

For a square real matrix A, we denote by $\lambda(A)$ the spectral radius of A — i.e., the largest of the absolute values of the eigenvalues of A.

The following result highlights the key aspects of Perron-Frobenius theory for irreducible matrices.

Theorem 2.4 (Perron-Frobenius Theorem) [11, Ch. XIII], [22, Ch. 1], [23, Ch. 1], [24, Ch. 2]) *Let $A = [(A)_{u,v}]_{u,v}$ be an irreducible matrix. Then $\lambda(A)$ is a simple eigenvalue of A and A has a positive (i.e., all components are positive) right and left eigenvector associated with the eigenvalue $\lambda(A)$.*

Since $\lambda(A)$ is actually an eigenvalue of A, we will refer to $\lambda(A)$ as the *largest eigenvalue of A*.

The following well-known result shows how to compute capacity.

Theorem 2.5 *Let S be an irreducible constrained system and let G be an irreducible lossless (for instance, deterministic) presentation of S. Then,*

$$\mathsf{cap}(S) \;=\; \log \lambda(A_G) \;.$$

2.7 Finite-state encoders

Let S be a constrained system and n be a positive integer. An (S, n)-*encoder* is a labeled graph $\mathcal{E}$ such that —

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

A *tagged* (S, n)-*encoder* is an (S, n)-encoder $\mathcal{E}$ where the outgoing edges from each state in $\mathcal{E}$ are assigned distinct *input tags* from an alphabet of size n. The notation $u \stackrel{s/a}{\to} v$ stands for an edge in $\mathcal{E}$ from state u to state v which is labeled a and tagged by s. A tagged encoder can be used to encode user sequences into constrained sequences, one symbol at a time, as follows: for each state u and source symbol s, there is a unique outgoing edge of the form $u \stackrel{s/a}{\to} v$; at state u, the symbol s is encoded to a and the state is changed to v.

A *rate* $p : q$ *finite-state* (S, α)-*encoder* is a tagged (S^q, α^p)-encoder, where we assume that the input tags are the α-ary p-blocks. A tagged (S, n)-encoder (respectively, a rate $p : q$ finite-state (S, α)-encoder) is deterministic or has finite anticipation according to whether the (S, n)-encoder (respectively, the (S^q, α^p)-encoder) does. In particular, only the (output) labels (and not the input tags) play a role in these properties. The anticipation of an encoder measures its *decoding delay*, namely, the number of output labels we need to look-ahead in order to decode the current input tag.

The following result, essentially due to Adler, Coppersmith and Hassner [1], shows when finite-state encoders exist.

Theorem 2.6 *Let S be a constrained system. Then there exists a rate $p : q$ finite-state (S, α)-encoder if and only if $p/q \le \mathsf{cap}_\alpha(S)$. Moreover, when $p/q \le \mathsf{cap}_\alpha(S)$ there exists such an encoder with finite anticipation.*

This result makes essential use of state splitting, as described in Section 4.

3 First upper bound on number of states

The main result of this section is Theorem 3.1 below, which provides an upper bound on the smallest number of states in any (S, n)-encoder with anticipation t, whenever there exists such an encoder. This result implies, in turn, that the problem of verifying whether there is an (S,n)-encoder with anticipation t is decidable. We assume that S is an irreducible constrained system, making a remark on reducible systems at the end of this section.

In the statement of our result, we use the notation $\mathcal{F}_G^t(u)$ to stand for the set of words of length t in $\mathcal{F}_G(u)$, i.e., the set of words of length t that can be generated from a state u in a labeled graph G.

Theorem 3.1 *Let S be an irreducible constrained system with Shannon cover G, let n and t be positive integers, and, for every state u in G, let $N(u,t) = |\mathcal{F}_G^t(u)|$. If there exists an (S,n)-encoder with anticipation t, then there exists an (S,n)-encoder with anticipation $\le t$ and at most $\sum_{u \in V_G} (2^{N(u,t)} - 1)$ states.*

By Lemma 2.1, we may assume that there is an *irreducible* (S,n)-encoder $\mathcal{E}$ with anticipation at most t. The proof of Theorem 3.1 is carried out by effectively constructing from $\mathcal{E}$ an (S,n)-encoder $\mathcal{E}'$ with anticipation $\le t$ and with a number of states which is at most the bound stated in the theorem. We describe the construction of $\mathcal{E}'$ below, and the theorem will follow from the next two lemmas.

For a state $u \in V_G$ and a nonempty subset $\mathcal{F}$ of $\mathcal{F}_G^t(u)$, let $\Gamma(u,\mathcal{F})$ denote the set of all states v in $\mathcal{E}$ for which $\mathcal{F}_\mathcal{E}(v) \subseteq \mathcal{F}_G(u)$ and $\mathcal{F}_\mathcal{E}^t(v) = \mathcal{F}$. Whenever $\Gamma(u,\mathcal{F})$ is nonempty we designate a specific such state $v \in \Gamma(u,\mathcal{F})$ and call it $v(u,\mathcal{F})$. By Lemma 2.3, at least one $\Gamma(u,\mathcal{F})$ is nonempty.

We now define the labeled graph $\mathcal{E}'$ as follows. The states of $\mathcal{E}'$ are the pairs $(u,\mathcal{F})$ such that $\Gamma(u,\mathcal{F})$ is nonempty. We draw an edge $(u,\mathcal{F}) \xrightarrow{a} (\widehat{u},\widehat{\mathcal{F}})$ in $\mathcal{E}'$ if and only if there is an edge $u \xrightarrow{a} \widehat{u}$ in G and an edge $v(u,\mathcal{F}) \xrightarrow{a} \widehat{v}$ in $\mathcal{E}$ for some $\widehat{v} \in \Gamma(\widehat{u},\widehat{\mathcal{F}})$.

Lemma 3.2 *For every $\ell \le t{+}1$,*

$$\mathcal{F}_{\mathcal{E}'}^\ell\Big((u,\mathcal{F})\Big) \;=\; \mathcal{F}_\mathcal{E}^\ell\Big(v(u,\mathcal{F})\Big)\,.$$

Proof. We prove that $\mathcal{F}_{\mathcal{E}'}^\ell\Big((u,\mathcal{F})\Big) \subseteq \mathcal{F}_\mathcal{E}^\ell\Big(v(u,\mathcal{F})\Big)$ by induction on ℓ. We leave the reverse inclusion (which is not used in this paper) to the reader.

The result is immediate for $\ell = 0$. Assume now that the result is true for some fixed $\ell \le t$. Let $w_0 w_1 \ldots w_\ell \in \mathcal{F}_{\mathcal{E}'}^{\ell+1}\Big((u,\mathcal{F})\Big)$, which implies that there is in $\mathcal{E}'$ a path of the form $(u,\mathcal{F}) \xrightarrow{w_0} (u_1,\mathcal{F}_1) \xrightarrow{w_1} (u_2,\mathcal{F}_2) \to \ldots \to (u_\ell,\mathcal{F}_\ell) \xrightarrow{w_\ell} (u_{\ell+1},\mathcal{F}_{\ell+1})$. By the inductive hypothesis, there is a path $v(u_1,\mathcal{F}_1) \xrightarrow{w_1} v_2 \xrightarrow{w_2} v_3 \to \ldots \to v_\ell \xrightarrow{w_\ell} v_{\ell+1}$ in $\mathcal{E}$. Therefore, the word $\mathbf{w} = w_1 w_2 \ldots w_\ell$ belongs to $\mathcal{F}_\mathcal{E}\Big(v(u_1,\mathcal{F}_1)\Big)$ and, since $\ell \le t$, we can extend $\mathbf{w}$ to form a word $\mathbf{w}\mathbf{w}'$ of length t that belongs to $\mathcal{F}_1$. Now, by definition of the edges in $\mathcal{E}'$, there is an edge $v(u,\mathcal{F}) \xrightarrow{w_0} \widehat{v}$ in $\mathcal{E}$ for some $\widehat{v} \in \Gamma(u_1,\mathcal{F}_1)$. Since $\mathbf{w}\mathbf{w}' \in \mathcal{F}_1$, there is a path labeled $\mathbf{w}$

outgoing from $\widehat{v}$ in $\mathcal{E}$ and, so, there is a path labeled $w_0 w_1 \ldots w_\ell$ outgoing from $v(u, \mathcal{F})$ in $\mathcal{E}$. Hence, $\mathcal{F}_{\mathcal{E}'}^{\ell+1}\Big((u, \mathcal{F})\Big) \subseteq \mathcal{F}_{\mathcal{E}}^{\ell+1}\Big(v(u, \mathcal{F})\Big)$, as desired. $\square$

The next lemma shows that $\mathcal{E}'$ is an (S, n)-encoder with anticipation $\leq t$.

Lemma 3.3 *The following three conditions hold:*

(a) The out-degree of each state in $\mathcal{E}'$ is n;

(b) $S(\mathcal{E}') \subseteq S$; and —

(c) $\mathcal{E}'$ has anticipation $\leq t$.

Proof. *Part (a):* It suffices to show that there is a one-to-one correspondence between the outgoing edges of $(u, \mathcal{F})$ in $\mathcal{E}'$ and those of $v(u, \mathcal{F})$ in $\mathcal{E}$. Consider the mapping Φ from outgoing edges of $(u, \mathcal{F})$ to outgoing edges of $v(u, \mathcal{F})$ defined by

$$\Phi\Big((u, \mathcal{F}) \stackrel{a}{\to} (\widehat{u}, \widehat{\mathcal{F}})\Big) \;=\; \Big(v(u, \mathcal{F}) \stackrel{a}{\to} \widehat{v}\Big)$$

where $\widehat{v} \in \Gamma(\widehat{u}, \widehat{\mathcal{F}})$. To see that Φ is well-defined, observe that since $\mathcal{E}$ has anticipation at most t, there cannot be two distinct edges $v(u, \mathcal{F}) \stackrel{a}{\to} \widehat{v}$ and $v(u, \mathcal{F}) \stackrel{a}{\to} \widehat{v}'$ with $\widehat{v}$ and $\widehat{v}'$ both belonging to the same $\Gamma(\widehat{u}, \widehat{\mathcal{F}})$. To see that Φ is onto, first consider an outgoing edge $v(u, \mathcal{F}) \stackrel{a}{\to} \widehat{v}$ from $v(u, \mathcal{F})$, and note that since $\mathcal{F} \subseteq \mathcal{F}_G(u)$, there is in G an outgoing edge $u \stackrel{a}{\to} \widehat{u}$ for some $\widehat{u}$. Let $\widehat{\mathcal{F}} = \mathcal{F}_{\mathcal{E}}^t(\widehat{v})$. We claim that $\widehat{v} \in \Gamma(\widehat{u}, \widehat{\mathcal{F}})$. Of course $\mathcal{F}_{\mathcal{E}}^t(\widehat{v}) = \widehat{\mathcal{F}}$; and since $\mathcal{F}_{\mathcal{E}}(v(u, \mathcal{F})) \subseteq \mathcal{F}_G(u)$ and G is deterministic, $\mathcal{F}_{\mathcal{E}}(\widehat{v}) \subseteq \mathcal{F}_G(\widehat{u})$. Thus, by definition of $\mathcal{E}'$ there is an edge $(u, \mathcal{F}) \stackrel{a}{\to} (\widehat{u}, \widehat{\mathcal{F}})$. We thus conclude that Φ is onto. Since u and a determine $\widehat{u}$ and since $\widehat{v}$ determines $\widehat{\mathcal{F}}$, it follows that Φ is 1–1. This completes the proof of (a).

Part (b): By definition of $\mathcal{E}'$, we see that whenever there is a path $(u_0, \mathcal{F}) \stackrel{w_0}{\to} (u_1, \mathcal{F}) \stackrel{w_1}{\to} (u_2, \mathcal{F}) \to \ldots \to (u_{\ell-1}, \mathcal{F}) \stackrel{w_{\ell-1}}{\to} (u_\ell, \mathcal{F})$ in $\mathcal{E}'$, there is also a path $u_0 \stackrel{w_0}{\to} u_1 \stackrel{w_1}{\to} \ldots \to u_{\ell-1} \stackrel{w_{\ell-1}}{\to} u_\ell$ in G. Thus $S(\mathcal{E}') \subseteq S(G) = S$, as desired.

Part (c): We must show that the initial edge of any path γ of length $t+1$ in $\mathcal{E}'$ is determined by its label $w_0 w_1 \ldots w_t$ and its initial state $(u, \mathcal{F})$. Write the initial edge of γ as: $(u, \mathcal{F}) \stackrel{w_0}{\to} (\widehat{u}, \widehat{\mathcal{F}})$. By Lemma 3.2, there is a path in $\mathcal{E}$ with label $w_0 w_1 \ldots w_t$ that begins at state $v(u, \mathcal{F})$. Since $\mathcal{E}$ has anticipation at most t, the label sequence $w_0 w_1 \ldots w_t$ and $v(u, \mathcal{F})$ determine the initial edge $v(u, \mathcal{F}) \stackrel{w_0}{\to} \widehat{v}$ of this path. So, it suffices to show that u, w_0, and $\widehat{v}$ determine $\widehat{u}$ and $\widehat{\mathcal{F}}$; for then $(u, \mathcal{F})$ and $w_0 w_1 \ldots w_t$ will determine the initial edge of γ.

Indeed, by definition of $\mathcal{E}'$, there must be an edge $u \stackrel{w_0}{\to} \widehat{u}$ in G such that $\widehat{v} \in \Gamma(\widehat{u}, \widehat{\mathcal{F}})$. Since G is deterministic, u and w_0 determine $\widehat{u}$. Furthermore, for any fixed $\widehat{u}$, the sets $\Gamma(\widehat{u}, \mathcal{G})$ are disjoint for distinct $\mathcal{G}$, and, so, $\widehat{v}$ determines $\widehat{\mathcal{F}}$. It follows that u, w_0, and $\widehat{v}$ determine $\widehat{u}$ and $\widehat{\mathcal{F}}$, as desired, thus proving (c). $\square$

Now, for every state $u \in V_G$, the number of distinct nonempty subsets $\Gamma(u, \mathcal{F})$ is bounded from above by $2^{N(u,t)}-1$. This yields the desired upper bound of Theorem 3.1 on the number of states of $\mathcal{E}'$.

Let S be an irreducible constrained system with Shannon cover G. It follows from Theorem 3.1, that in order to check whether there is an (S, n)-encoder with anticipation at most t, we can exhaustively check all irreducible graphs $\mathcal{E}$ with labeling over $\Sigma(S)$, with out-degree n, and with number of states $|V_\mathcal{E}| \leq \sum_{u \in V_G}(2^{N(u,t)} - 1)$. For such a graph $\mathcal{E}$, we must verify whether $\mathcal{E}$ has anticipation at most t and whether $S(\mathcal{E}) \subseteq S$. The former can in fact be accomplished by an efficient algorithm [19]. For the latter, proceed as follows. Construct the determinizing graph H of $\mathcal{E}$; since $\mathcal{E}$ is irreducible, every state of $\mathcal{E}$ must occur in a state Z of any irreducible sink H' of H, where Z is regarded as a subset of $V_\mathcal{E}$. Hence, we have

$$S(\mathcal{E}) \; = \; S(H') \; . \tag{1}$$

Now, consider the *fiber product* $G * H'$, which is defined to be the labeled graph consisting of states (u, v) and edges $(u, v) \xrightarrow{a} (\widehat{u}, \widehat{v})$ if and only if $u \xrightarrow{a} \widehat{u}$ in G and $v \xrightarrow{a} \widehat{v}$ in H'. It is easy to see that

$$S(G * H') \; = \; S(G) \cap S(H') \; \subseteq \; S(H') \; . \tag{2}$$

We see from (1) and (2) that $S(\mathcal{E}) \subseteq S$ if and only if

$$S(G * H') \; = \; S(H') \; . \tag{3}$$

Now, a necessary condition for this is that $S(G * H')$ be irreducible. So, to check (3), we enumerate over all irreducible components G_i of $G * H'$. All satisfy $S(G_i) \subseteq S(G * H')$, and equality must hold by Lemma 2.1 for at least one G_i. Hence, by (2), it suffices to check whether $S(H') = S(G_i)$ for at least one i. Now, H' and the G_i's are irreducible deterministic labeled graphs, and so to check whether $S(H') = S(G_i)$, we need only reduce H' and G_i to see if their reductions coincide. This can be done (efficiently) using the Moore algorithm [19].

For an irreducible graph which does not consist entirely of a single cycle, it is not hard to see that the Moore co-form increases anticipation by 1. Thus, except for a trivial case, the existence of an (S, n)-encoder with anticipation at most t and at most M states implies the existence of an (S, n)-encoder with anticipation exactly t and at most $n^t M$ states. It follows from this and the preceding discussion that the problem of checking whether an irreducible constrained system S has an (S, n)-encoder with anticipation (exactly) t is decidable.

So far in this section, we have dealt with irreducible constrained systems S, thus including most constrained systems of interest. For a general (possibly) reducible constrained system S, one can use the irreducible components of a presentation of S to find an explicit finite collection of irreducible subsystems $S_1, S_2, \ldots, S_k$, such that any irreducible (S, n)-encoder is an (S_i, n)-encoder for some i. Having done that, Theorem 3.1 can be applied to each of those subsystems.

4 State splitting

Let G be a deterministic presentation of S with $\mathsf{cap}(S) \geq \log n$. The state-splitting algorithm of [1] iteratively applies the following operation beginning with G in order to produce an (S, n)-encoder.

For every state u in G, denote by $E(u) = E_G(u)$ the set of outgoing edges from u in G. A (round of) *state splitting* on a graph G is specified by partitioning for each state $u \in V_G$ the set $E(u)$ into atoms $E(u, i) = E_G(u, i)$, $1 \leq i \leq x_u$. The states of the resulting graph, G', are defined to be these atoms $E(u, i)$, $u \in V_G$, $1 \leq i \leq x_u$. For each edge $u \stackrel{a}{\to} v$ in G belonging to $E(u, i)$, we endow G' with x_v edges $E(u, i) \stackrel{a}{\to} E(v, j)$, $1 \leq j \leq x_v$. Those edges are called the *descendants* in G' of the edge $u \stackrel{a}{\to} v$.

The following definition plays an essential role in the choices of which states to split and how to split them. Given a nonnegative integer square matrix A and a positive integer n, an (A, n)-*approximate eigenvector* is a nonnegative integer vector $\mathbf{x} \neq \mathbf{0}$ such that

$$A\mathbf{x} \ \geq \ n\mathbf{x} \ ,$$

where the (weak) inequality holds component-by-component. The set of all (A, n)-approximate eigenvectors is denoted $\mathcal{X}(A, n)$. Now it is a consequence of Perron-Frobenius theory that

$$\mathcal{X}(A, n) \neq \emptyset \quad \text{if and only if} \quad \lambda(A) \geq n \ ;$$

furthermore, if A is irreducible and $\lambda(A) = n$, then every (A, n)-approximate eigenvector is a true right eigenvector associated with eigenvalue n (see [21] for a proof of these statements). Since for a deterministic presentation G of S, $\mathsf{cap}(S) = \log \lambda(A_G)$, the preceding shows that approximate eigenvectors exist so long as the capacity is sufficiently large, i.e., $\mathsf{cap}(S) \geq \log n$.

The rough idea of the state-splitting algorithm is as follows. Let S be an irreducible constrained system with capacity $\mathsf{cap}(S) \geq \log n$ and let G denote the Shannon cover of S. Then $\lambda(A_G) \geq n$, and so there is an (A_G, n)-approximate eigenvector $\mathbf{x}$. Now, iteratively split states in a way that creates a sequence of graphs such that the number of states grows but the sizes of the approximate eigenvector entries decrease. Eventually, the all-ones vector is an approximate eigenvector; it is easy to see that this means that the resulting graph has minimum out-degree at least n. Then the resulting graph can be pruned to form an (S, n)-encoder; see [1] or [21] for details.

The following result shows what the approximate eigenvector looks like at the next-to-the-last round of state splitting.

Lemma 4.1 *Let G be a graph and, for each state $u \in V_G$, let $P_G(u) = \{E_G(u, i)\}_{i=1}^{x_u}$ be a partition of $E_G(u)$ into x_u atoms. Denote by G' the graph obtained from G by a round of* *state splitting according to the partitions $P_G(u)$. Then G' has out-degree at least n if and only if*

$$\sum_{e \in E_G(u,i)} x_{\tau(e)} \;\geq\; n \quad \textit{for each } (u,i)\ . \tag{4}$$

Proof. The lemma follows from the fact that each edge e in G has $x_{\tau(e)}$ descendant edges in G'. □

Note that (4) implies that $\mathbf{x} = [x_u]_u$ is an (A_G, n)-approximate eigenvector, since for each $u \in V_G$ we have

$$(A_G\mathbf{x})_u \;=\; \sum_{i=1}^{x_u} \sum_{e \in E_G(u,i)} x_{\tau(e)} \;\geq\; \sum_{i=1}^{x_u} n \;=\; n x_u\ . \tag{5}$$

5 Second upper bound on number of states

Usually, the state-splitting algorithm requires several rounds of splitting. But it is not hard to see that if a sequence of t rounds of splitting applied to a graph G yields a graph with minimum out-degree at least n, then a single round of splitting can be applied to the higher power graph G^t to obtain a graph with minimum out-degree at least n^t (see [4], [15]). The following result shows that this yields an encoder with anticipation at most $3t-1$.

Theorem 5.1 *Let S be a constrained system presented by a deterministic graph G and let n and t be positive integers. Suppose that G^t can be split in one round, yielding a graph with minimum out-degree at least n^t. Then there is an (S,n)-encoder with anticipation $\leq 3t-1$.*

Proof. Suppose that G^t can be split in one round, yielding a graph $\mathcal{E}_1$ with minimum out-degree $\geq n^t$. By deleting excess edges, $\mathcal{E}_1$ can be pruned into an (S^t, n^t)-encoder $\mathcal{E}_2$ with anticipation 1 over the alphabet $\Sigma(S^t)$. Let $\mathcal{E}_3$ be the Moore co-form of $\mathcal{E}_2$. Then $\mathcal{E}_3$ is an (S^t, n^t)-encoder with anticipation 2. If we replace the n^t outgoing edges from each state in $\mathcal{E}_3$ by an n-ary tree of depth t, we obtain an (S,n)-encoder $\mathcal{E}_4$ with anticipation $\leq 3t-1$. □

Our next result is, in a way, a converse to this.

Theorem 5.2 *Let S be an irreducible constrained system presented by an irreducible deterministic graph G and let n and t be positive integers. If there is an (S,n)-encoder with anticipation t, then G^t can be split in one round, yielding a graph with minimum out-degree at least n^t.*

Proof. Let $\mathcal{E}$ be an (S,n)-encoder with anticipation t, let $\Sigma = \Sigma(S)$, and let H be the determinizing graph constructed from $\mathcal{E}$ as in Section 2.2. Recall that each state $Z \in V_H$ is a subset $T_{\mathcal{E}}(\mathbf{w}, v)$ of states of $\mathcal{E}$ that can be reached in $\mathcal{E}$ from a given state $v \in V_{\mathcal{E}}$ by paths that generate a given word $\mathbf{w}$. Let H' be an irreducible sink of H and let $\mathbf{x} = [x_u]_{u \in V_G}$ be the nonnegative integer vector defined by

$$x_u = \max\{\, |Z| \,:\, Z \in V_{H'} \text{ and } \mathcal{F}_{H'}(Z) \subseteq \mathcal{F}_G(u) \,\} , \quad u \in V_G .$$

In case there is no state $Z \in V_{H'}$ such that $\mathcal{F}_{H'}(Z) \subseteq \mathcal{F}_G(u)$, define $x_u = 0$. Then $\mathbf{x}$ is an (A_G, n)-approximate eigenvector (see [20]).

Let $Z = T_{\mathcal{E}}(\mathbf{w}, v)$ be a state in H' and suppose that Z contains two distinct states, z and z', of $\mathcal{E}$. First, we claim that there is no word $\mathbf{w}'$ of length t that can be generated in $\mathcal{E}$ from both z and z'. Otherwise, we would have in $\mathcal{E}$ two paths of length $\ell(\mathbf{w}) + t$, starting at the same state v, with the same labeling $\mathbf{w}\mathbf{w}'$, that do not agree in at least one of their first $\ell(\mathbf{w})$ edges. This, however, contradicts the fact that $\mathcal{E}$ has anticipation t.

For $\mathbf{w}' \in \mathcal{F}^t_{H'}(Z)$, denote by $Z_{\mathbf{w}'}$ the terminal state in H' of a path labeled $\mathbf{w}'$ starting at Z. As we have just shown, a word $\mathbf{w}' \in \mathcal{F}^t_{H'}(Z)$ can be generated in $\mathcal{E}$ from exactly one state $z \in Z$. Therefore, the sets $\mathcal{F}^t_{\mathcal{E}}(z)$, $z \in Z$, form a partition of $\mathcal{F}^t_{H'}(Z)$. Furthermore, by the losslessness of $\mathcal{E}$, the number of paths in $\mathcal{E}$ that start at $z \in Z$ and generate $\mathbf{w}' \in \mathcal{F}^t_{\mathcal{E}}(u)$ equals $|T_{\mathcal{E}}(\mathbf{w}', z)| = |Z_{\mathbf{w}'}|$. Since $\mathcal{E}$ is an (S, n)-encoder, we conclude:

$$\sum_{\mathbf{w}' \in \mathcal{F}^t_{\mathcal{E}}(z)} |Z_{\mathbf{w}'}| \;=\; n^t \quad \text{for every } z \in Z . \tag{6}$$

For each state $u \in V_G$ such that $x_u \neq 0$, select some $Z = Z(u) \in V_{H'}$ such that $|Z| = x_u$ and $\mathcal{F}_{H'}(Z) \subseteq \mathcal{F}_G(u)$. Now, the partition $\{\mathcal{F}^t_{\mathcal{E}}(z) : z \in Z\}$ of $\mathcal{F}^t_{H'}(Z)$ may be regarded as a partition of $\mathcal{F}^t_G(u)$ by appending the complement $\mathcal{F}^t_G(u) \setminus \mathcal{F}^t_{H'}(Z)$ to one of the atoms $\mathcal{F}^t_{\mathcal{E}}(z)$, $z \in Z$. Since G^t is deterministic, this defines a partition $P_{G^t}(u) = \{E_{G^t}(u, z)\}_{z \in Z(u)}$ of the outgoing edges from u in G^t into $|Z(u)| = x_u$ atoms. For $\mathbf{w}' \in \mathcal{F}_{\mathcal{E}}(z)$, let u' denote the terminal state of the edge in G^t that begins at u and is labeled $\mathbf{w}'$. Now, if $\mathbf{w}'' \in \mathcal{F}_{H'}(Z_{\mathbf{w}'})$, then $\mathbf{w}'\mathbf{w}'' \in \mathcal{F}_{H'}(Z) \subseteq \mathcal{F}_G(u)$. Since G is deterministic, this implies that $\mathbf{w}'' \in \mathcal{F}_G(u')$. Thus $\mathcal{F}_{H'}(Z_{\mathbf{w}'}) \subseteq \mathcal{F}_G(u')$ and, so, $|Z_{\mathbf{w}'}| \leq x_{u'}$. This, together with equation (6), shows that the splitting of G^t defined by the partition $P_{G^t}(u)$ satisfies (4), namely,

$$\sum_{e \in E_{G^t}(u,z)} x_{\tau(e)} \;\geq\; n^t \quad \text{for every } u \in V_G \text{ and } z \in Z(u) .$$

Hence, by Lemma 4.1, the split graph has minimum out-degree at least n^t. $\square$

Theorems 5.1 and 5.2 show that if S is an irreducible constrained system with Shannon cover G, and t is the smallest possible anticipation of an (S, n)-encoder, then there is an (S, n)-encoder obtained by splitting G^t in one round such that this encoder has anticipation at most $3t-1$. Of course, this is within a constant factor of the smallest anticipation possible. In the following theorem we improve this to $2t-1$. This theorem trades a weaker bound on anticipation than the bound given by Theorem 3.1 for a stronger (*singly* exponential) bound on the number of states.

Theorem 5.3 *Let S be an irreducible constrained system with Shannon cover G, let n and t be positive integers, and, for every state u in G, let $N(u,t) = |\mathcal{F}_G^t(u)|$. If there exists an (S,n)-encoder with anticipation t, then there exists an (S,n)-encoder with anticipation $\leq 2t{-}1$ and at most $\sum_{u \in V_G} N(u,t)(n^t-1)/(n-1)$ states.*

Proof. Let $\mathcal{E}$ be an (S,n)-encoder with anticipation t and let H, H', and $Z(u)$, $u \in V_G$, be as in the proof of Theorem 5.2. Recall that each state in H' is a subset $Z = T_{\mathcal{E}}(\mathbf{w}, v)$ of states in $\mathcal{E}$ which are accessible from $v \in V_{\mathcal{E}}$ by paths labeled $\mathbf{w}$.

As in the proof of Theorem 5.2, for each $u \in V_G = V_{G^t}$, if $Z(u)$ is nonempty, the sets $\mathcal{F}_{\mathcal{E}}^t(z)$, $z \in Z(u)$, partition the set $\mathcal{F}_{H'}^t(Z(u)) \subseteq \mathcal{F}_G^t(u)$; and as in that proof, this partition may be regarded as a partition $P_{G^t}(u) = \{E_{G^t}(u,z)\}_{z \in Z(u)}$ of the outgoing edges from each state u in G^t. Split G^t according to the partitions $P_{G^t}(u)$, forming a graph G' with states $E(u,z) = E_{G^t}(u,z)$.

Henceforth, we regard the edges of G' as paths of length t (where we string $t{-}1$ dummy states between the symbols from Σ). From this point of view, it is easy to see that the anticipation of G' is at most $2t{-}1$.

For each $E(u,z) \in V_{G'}$, we partially merge some of the paths of length t emanating from $E(u,z)$ to form an n-ary tree. We do this as follows. Fix $\mathbf{w} \in \mathcal{F}_{\mathcal{E}}^t(z)$ and let u' be the terminal state of the path in G starting at state u that is labeled $\mathbf{w}$. The state u' (as a state of G^t) is split into $|Z(u')|$ states. Thus, there are $|Z(u')|$ paths emanating from state $E(u,z)$ in G' that are labeled $\mathbf{w}$. On the other hand, the number of paths emanating from state v in $\mathcal{E}$ that are labeled $\mathbf{w}$ is exactly $|T_{\mathcal{E}}(\mathbf{w}, v)|$. Now, $|Z(u')| \geq |T_{\mathcal{E}}(\mathbf{w}, v)|$ by the choice of $Z(u')$. Hence, we can form an injection

$$\iota : \left\{ \begin{array}{c} \text{paths of length } t \\ \text{emanating from } v \text{ in } \mathcal{E} \end{array} \right\} \rightarrow \left\{ \begin{array}{c} \text{paths of length } t \\ \text{emanating from } E(u,z) \text{ in } G' \end{array} \right\}$$

so that the label of the path $\iota(\gamma)$ in G' is the same as the label of the path γ in $\mathcal{E}$. We partially merge the paths in the image of ι as follows. If γ and γ' are paths of length t in $\mathcal{E}$ whose longest common prefix has length m, then the prefixes of length m of $\iota(\gamma)$ and $\iota(\gamma')$ are merged into a single path. Note that this only merges (prefixes of) paths having the same label, so that the resulting graph, G'', has a well-defined labeling. Moreover, this labeling has no more anticipation than the labeling of G' (at most $2t{-}1$). Furthermore, because $\mathcal{E}$ has outdegree n, G'' contains an irreducible subgraph having outdegree n. This subgraph has at most

$$\sum_{u \in V_G} N(u,t) \Big(\sum_{m=0}^{t-1} n^m \Big) \;=\; \sum_{u \in V_G} N(u,t)(n^t-1)/(n-1)$$

states. $\square$

6 Lower bound on the anticipation

The state-splitting algorithm provides an upper bound on the smallest anticipation of an (S, n)-encoder; namely, it produces encoders $\mathcal{E}$ such that

$$\mathcal{A}(\mathcal{E}) \leq \min_{\mathbf{x}=[x_u]_u \in \mathcal{X}(A_G, n)} \left\{ \textstyle\sum_u x_u \right\} ,$$

where $\mathsf{cap}(S) \geq \log n$ and G is a deterministic presentation of S (see [21]).

This bound is quite poor, since it may be exponential in $|V_G|$ (see [20]). However, in [4], Ashley showed that under the same hypotheses for $t = O(|V_G|)$, G^t can be split in one round yielding a graph with minimum out-degree at least n^t. Thus, by Theorem 5.1, there are always (S, n)-encoders with anticipation at most $O(|V_G|)$. For other upper bounds on the smallest anticipation of any (S, n)-encoder, see [7].

The following *lower* bound on the anticipation is taken from [20]. Hereafter, $\|\mathbf{x}\|_\infty$ stands for the largest component of $\mathbf{x}$.

Theorem 6.1 *Let S be a constrained system presented by a deterministic graph G and let $\mathcal{E}$ be an (S, n)-encoder. Then*

$$\mathcal{A}(\mathcal{E}) \geq \min_{\mathbf{x} \in \mathcal{X}(A_G, n)} \left\{ \log_n \|\mathbf{x}\|_\infty \right\} .$$

A special case of this lower bound appears in [10].

Now, Theorem 5.2 provides another lower bound on anticipation, namely:

Theorem 6.2 *Let S be a constrained system presented by a deterministic graph G and let $\mathcal{E}$ be an (S, n)-encoder. Then $\mathcal{A}(\mathcal{E})$ is at least the smallest t such that G^t can be split in one round, yielding a graph with minimum out-degree at least n^t. Moreover, $\mathcal{A}(\mathcal{E})$ is at least the smallest t for which there is an (A_G, n)-approximate eigenvector $\mathbf{x} = [x_u]_u$ and respective partitions $\{E_{G^t}(u, i)\}_{i=1}^{x_u}$ of the outgoing edges from each state u in G^t such that*

$$\sum_{e \in E_{G^t}(u,i)} x_{\tau(e)} \geq n^t \quad \textit{for each } (u, i) .$$

The first part of this result is simply a re-statement of Theorem 5.2. The second part is a consequence of the proof of Theorem 5.2. Note that if we had merely required $\mathbf{x}$ to be an (A_{G^t}, n^t)-approximate eigenvector, then the second part would follow from the statement of Theorem 5.2, the statement of Lemma 4.1 (made with respect to the graph G^t), and (5).

How do these two bounds compare? We give examples to show that neither of these lower bounds implies the other.

Example 6.3 The following is an example where the bound in Theorem 6.1 is better than that of Theorem 6.2. Let G be the graph whose adjacency matrix is

$$A_G \;=\; \begin{bmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 5 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} ,$$

with the edges of G all labeled distinctly. Let $n = 2$ and $\mathbf{x} = [\,2\;3\;5\;1\,]^T$. Then a calculation shows that $\mathbf{x}$ is an (A_G, n)-approximate eigenvector. Moreover, the approximate eigenvector algorithm, due to P. Franaszek (see [21]), shows that $\mathbf{x}$ achieves the minimum $\|\mathbf{x}\|_\infty$. Thus, the bound of Theorem 6.1 is $\lceil \log_2 5 \rceil = 3$. But it is possible to split G^2 in one round, yielding a graph with minimum out-degree at least $n^2 = 4$: we describe this splitting by decomposing the rows of

$$A_G^2 = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 5 & 1 & 0 & 0 \\ 0 & 5 & 0 & 5 \\ 0 & 0 & 1 & 1 \end{bmatrix} ,$$

namely:

$$\begin{array}{rcl} [\,0\,1\,1\,1\,] & = & [\,0\,1\,0\,1\,] \;+\; [\,0\,0\,1\,0\,] \\ [\,5\,1\,0\,0\,] & = & [\,2\,0\,0\,0\,] \;+\; [\,2\,0\,0\,0\,] \;+\; [\,1\,1\,0\,0\,] \\ [\,0\,5\,0\,5\,] & = & [\,0\,1\,0\,1\,] \;+\; [\,0\,1\,0\,1\,] \;+\; [\,0\,1\,0\,1\,] \;+\; [\,0\,1\,1\,0\,] \;+\; [\,0\,1\,0\,1\,] \\ [\,0\,0\,1\,1\,] & = & [\,0\,0\,1\,1\,]\,; \end{array}$$

here, the row of A_G corresponding to state u has been written as a sum of x_u vectors $\mathbf{y}^{(u,i)}$, where $(\mathbf{y}^{(u,i)})_v$ is the number of edges in $E_{G^2}(u,i)$ that end at state v. It is straightforward to check that this splitting satisfies (4) with respect to the graph G^2 and, so, by Lemma 4.1 this splitting yields a graph with minimum out-degree at least $n^2 = 4$. Thus, the lower bound in Theorem 6.2 is at most 2 (and it is easy to check that it is exactly 2). So for this example, the bound in Theorem 6.1 is better than that of Theorem 6.2. We leave it as an exercise for the reader to show that in this case this better bound (namely anticipation equaling 3) can actually be achieved by splitting G in three rounds.

On the other hand, when S is irreducible, $\mathsf{cap}(S) = \log n$, and G is the Shannon cover of S, we claim that the lower bound of Theorem 6.2 implies the lower bound of Theorem 6.1 (and Example 6.4 below shows that in this case Theorem 6.2 is in fact strictly stronger). Indeed, let t denote the bound of Theorem 6.2. Then for each state $u \in V_G$ there is a partition $\{E_{G^t}(u,i)\}_{i=1}^{x_u}$ of the outgoing edges from u in G^t such that the vector $\mathbf{x} = [x_u]_u$ is an (A_G, n)-approximate eigenvector and

$$\sum_{e \in E_{G^t}(u,i)} x_{t(e)} \;\geq\; n^t \quad \text{for each } (u,i)\;. \tag{7}$$

We now claim that (7) holds in our case with equality for every (u,i). Otherwise, the corresponding splitting would yield a lossless presentation of S^t with minimum out-degree at least n^t and at least one state with out-degree greater than n^t — contradicting the equality $\mathsf{cap}(S) = \log n$.

Let $u_{\max}$ be a state in G for which $x_{u_{\max}} = \|\mathbf{x}\|_\infty$. Also, let v be a state with an outgoing edge, in G^t, to $u_{\max}$. Then any edge e from v to $u_{\max}$ in G^t belongs to some $E_{G^t}(v,i)$ and so the equality $\sum_{e \in E_{G^t}(v,i)} x_{t(e)} = n^t$ implies

$$x_{u_{\max}} \le n^t$$

— i.e., $t \ge \log_n \|\mathbf{x}\|_\infty \ge \min_{\mathbf{y} \in \mathcal{X}(A_G,n)} \Big\{ \log_n \|\mathbf{y}\|_\infty \Big\}$.

Example 6.4 The following is an example where the bound of Theorem 6.2 is better than that of Theorem 6.1. Let G be the graph whose adjacency matrix is

$$A_G = \begin{bmatrix} 4 & 3 \\ 1 & 2 \end{bmatrix} ,$$

again with the edges of G all labeled distinctly. Let $n = 5$ and $\mathbf{y} = [\,3\ 1\,]^T$. Then $\mathbf{y}$ is an (A_G,n)-approximate eigenvector. In fact, it is a true eigenvector, and so $\mathsf{cap}(S) = \log 5$. Thus, any (A_G,n)-approximate eigenvector must be a multiple of $\mathbf{y}$. So, the bound in Theorem 6.1 is $\lceil \log_5 3 \rceil = 1$. We claim that the bound in Theorem 6.2 is at least 2. Indeed, if t were 1, then we could find partitions $\{E(u,i)\}_{i=1}^{x_u}$ for each state u in G such that (7) holds with equality. Now, $\mathbf{x} = [x_u]_u$ must be a multiple of $\mathbf{y}$. However, since $\mathbf{y}$ contains the entry 3, every proper multiple of $\mathbf{y}$ contains an entry which is greater than $n = 5$, making it impossible to have equality in (7) for every (u,i). Therefore, we must have $\mathbf{x} = \mathbf{y}$, and an exhaustive check reveals that there are no partitions $\{E(u,i)\}_{i=1}^{x_u}$ so that (7) holds with equality. Thus, the bound in Theorem 6.2 is indeed at least 2. In fact, this bound is exactly 2 and in this case it can be achieved by splitting G in two rounds.

7 Sliding-block decodability

A tagged (S,n)-encoder is (m,a)-*sliding-block decodable*, if the following holds: for any two paths $e_{-\mathsf{m}} e_{-\mathsf{m}+1} \ldots e_0 \ldots e_{\mathsf{a}}$ and $e'_{-\mathsf{m}} e'_{-\mathsf{m}+1} \cdots e'_0 \cdots e'_{\mathsf{a}}$ that generate the same word, the edges e_0 and e'_0 have the same (input) tag. We allow m to be negative, but it only makes sense when $\mathsf{m} + \mathsf{a} \ge 0$.

A *sliding-block decoder* for a tagged (S,n)-encoder is a mapping $\mathcal{D}$ from the set $(\Sigma(S))^{\mathsf{m}+\mathsf{a}+1}$ to the set of input tags, such that, if $\mathbf{w} = w_1 w_2 \cdots$ is any symbol sequence generated by the encoder from the input tag sequence $\mathbf{s} = s_1 s_2 \cdots$, then, for $i > \mathsf{m}$,

$$s_i = \mathcal{D}(w_{i-\mathsf{m}}, \ldots, w_i, \ldots, w_{i+\mathsf{a}}) \ .$$

It is easy to verify that a tagged (S, n)-encoder has a sliding-block decoder if and only if it is sliding-block decodable.

Notions of sliding-block decodability naturally extend to rate $p : q$ finite-state (S, α)-encoders. When used to decode data received from a noisy channel, a single error at the input to a sliding-block decoder can only affect the decoding of q-blocks that fall in a "window" of length at most $\mathsf{m}+\mathsf{a}+1$, measured in q-blocks. Thus, error propagation is controlled by sliding-block decoders.

The state-splitting algorithm of [1] constructs encoders with sliding-block decoders for a broad class of constrained systems, namely the *finite-memory systems* (also called *shifts of finite type*) — see [21]. This was extended to more general constrained systems in [16]. The proof of Theorem 3.1 can be modified to show that the problem of checking whether for given S, n, m, a, there is an (m, a)-sliding-block decodable tagged (S, n)-encoder is decidable. But a more efficient scheme is actually a consequence of a result in [6]:

Theorem 7.1 *Let S be an irreducible constrained system with a Shannon cover G and let n, m, and a be integers such that $n > 0$, $\mathsf{a} \geq 0$, and $\mathsf{m}+\mathsf{a} \geq 0$. For every state u in G, let $N(u, t) = |\mathcal{F}_G^t(u)|$. If there exists an (m, a)-sliding-block decodable (S, n)-encoder, then there exists such an encoder with at most $\sum_{u \in V_G} N(u, |\mathsf{m}| + 2\mathsf{a})$ states.*

Proof. The proof is an easy corollary of Proposition 12.1 in [6], which asserts that if there exists an (m, a)-sliding-block decodable (S, n)-encoder, then one can construct such an encoder by starting with G and applying $|\mathsf{m}| + \mathsf{a}$ rounds of state splitting in the backward direction, followed by a rounds of state splitting in the forward direction. The states of the resulting graph can be identified with the atoms of a partition of the set of all paths of length $|\mathsf{m}| + 2\mathsf{a}$ in G. Any such partition has at most $\sum_{u \in V_G} N(u, |\mathsf{m}| + 2\mathsf{a})$ atoms. $\square$

The following bound is easily verified.

Proposition 7.2 *Let $\mathcal{E}$ be an irreducible (m, a)-sliding-block decodable encoder. Then $\mathsf{a} \geq \mathcal{A}(\mathcal{E})$.*

Hence, we can apply the lower bounds on the anticipation that were presented in Section 6, to obtain lower bounds on the parameter a in sliding-block decodable encoders. On the other hand, results in [7] provide upper bounds on the attainable value of a in encoders obtained by constructions that yield sliding-block decodable encoders for finite-type constrained systems.

Finally, it is natural to wonder if the problem of checking whether a given constrained system has a sliding-block decodable encoder with prescribed decoding window size is decidable. Precisely, given S, n, and w, how does one decide if there is a tagged (S, n)-encoder which is (m, a)-sliding-block decodable for some m and a such that $\mathsf{m} + \mathsf{a} + 1 = \mathsf{w}$? Such a procedure does not necessarily follow from Theorem 7.1 since m may be negative. Allowing the memory to be negative is very important since sometimes the minimum decoding window size cannot be achieved with $\mathsf{m} \geq 0$ (see [14], [12]). Nevertheless, Hollmann [13] has found such a procedure.

References

[1] R.L. Adler, D. Coppersmith, M. Hassner, *Algorithms for sliding block codes — an application of symbolic dynamics to information theory, IEEE Trans. Inform. Theory,* 29 (1983), 5–22.

[2] R.L. Adler, L.W. Goodwyn, B. Weiss, *Equivalence of topological Markov shifts, Israel J. Math.,* 27 (1977), 49–63.

[3] J.J. Ashley, *On the Perron-Frobenius eigenvector for non-negative integral matrices whose largest eigenvalue is integral, Linear Algebra Appl.,* 94 (1987), 103–108.

[4] J.J. Ashley, *A linear bound for sliding block decoder window size, IEEE Trans. Inform. Theory,* 34 (1988), 389–399.

[5] J.J. Ashley, *A linear bound for sliding block decoder window size, II,* preprint, 1995.

[6] J.J. Ashley, B.H. Marcus, *Canonical encoders for sliding block decoders, SIAM J. Discrete Math,* to appear.

[7] J.J. Ashley, B.H. Marcus, R.M. Roth, *Construction of encoders with small decoding look-ahead for input-constrained channels, IEEE Trans. Inform. Theory,* 41 (1995), 55-76.

[8] R. Fischer, *Sofic systems and graphs, Monats. fur Math.* 80 (1975), 179–186.

[9] R. Fischer, *Graphs and symbolic dynamics, Colloq. Math. Soc. János Bólyai, Topics in Information Theory,* 16 (1975), 229–243

[10] P.A. Franaszek, *Coding for constrained channels: a comparison of two approaches, IBM J. Res. Dev.,* 33 (1989), 602–607.

[11] F.R. Gantmacher, *Matrix Theory, Volume II,* Chelsea Publishing Company, New York, 1960.

[12] H.D.L. Hollmann, *On the construction of bounded-delay encodable codes for constrained systems,* preprint, 1993.

[13] H.D.L. Hollmann, Personal communication.

[14] K.A.S. Immink, *Block-decodable runlength-limited codes via look-ahead technique, Philips J. Research,* 46 (1992), 293-310.

[15] H. Kamabe, *Minimum scope for sliding block decoder mappings, IEEE Trans. Inform. Theory,* 35 (1989), 1335–1340.

[16] R. Karabed, B.H. Marcus, *Sliding-block coding for input-restricted channels, IEEE Trans. Inform. Theory,* 34 (1988), 2–26.

[17] Z.-A. Khayrallah, D. Neuhoff, *Subshift models and finite-state codes for input-constrained noiseless channels: a tutorial,* preprint, 1990.

[18] D. Lind and B. Marcus, *An Introduction to Symbolic Dynamics and Coding,* Cambridge University Press, 1995.

[19] Z. Kohavi, *Switching and Finite Automata Theory,* Second Edition, Tata McGraw-Hill, New Delhi, 1978.

[20] B.H. Marcus, R.M. Roth, *Bounds on the number of states in encoders graphs for input-constrained channels, IEEE Trans. Inform. Theory,* IT-37 (1991), 742–758.

[21] B.H. Marcus, P.H. Siegel, J.K. Wolf, *Finite-state modulation codes for data storage, IEEE J. Sel. Areas Comm.,* 10 (1992), 5–37.

[22] H. Minc, *Nonnegative Matrices,* Wiley, New York, 1988.

[23] E. Seneta, *Non-negative Matrices and Markov Chains,* Second Edition, Springer, New York, 1980.

[24] R.S. Varga, *Matrix Iterative Analysis,* Prentice-Hall, Englewood Cliffs, New Jersey, 1962.

JONATHAN J. ASHLEY was born in Fresno, California in 1954. He received the B.A. in mathematics from U.C. Berkeley, the M.A. in mathematics from California State University, Fresno, and the Ph.D. in computer science from U.C. Santa Cruz. Since 1990 he has worked at the IBM Almaden Research Center in San Jose, California.

BRIAN H. MARCUS attended Claremont Mens College and received his B.A. from Pomona College in 1971. He received his M.A. and Ph.D. in mathematics from the University of California at Berkeley in 1972 and 1975, respectively. From 1975–1984 he was Assistant Professor and then Associate Professor of Mathematics (with tenure) at the University of North Carolina - Chapel Hill. Since 1984 he has been a Research Staff Member at the IBM Almaden Research Center. His current research interests include symbolic dynamics and coding for storage devices.

Dr. Marcus held the IBM Watson Postdoctoral Fellowship in 1976–7 and shared the Leonard G. Abraham Prize Paper Award of the IEEE Communications Society in 1993. He gave an invited plenary lecture at the 1995 International Symposium on Information Theory. He has held visiting positions at Stanford University, UC-Berkeley, UC-Santa Cruz and the University of Maryland. He is the author of fifty papers in ergodic theory and information theory as well as a textbook, *An Introduction to Symbolic Dynamics* (Cambridge University Press, 1995).

RON M. ROTH was born in Ramat Gan, Israel, in 1958. He received the B.Sc. degree in computer engineering, the M.Sc. in electrical engineering and the D.Sc. in computer science from Technion — Israel Institute of Technology, Haifa, Israel, in 1980, 1984 and 1988, respectively. Since 1988 he has been with the Computer Science Department at Technion. During the academic years 1989–91 he was a Visiting Scientist at the IBM Research Division, Almaden Research Center, San Jose, California, and, since 1994, he has been a consultant for Hewlett-Packard Company — Israel Science Center, Haifa, Israel. He is currently on a sabbatical leave from Technion, visiting Hewlett-Packard Laboratories, Palo Alto, California.

His research interests include coding theory, information theory, and their application to the theory of complexity.