Universal Polarization for Processes with Memory

Boaz Shuval, Ido Tal
Department of Electrical Engineering,
Technion, Haifa 32000, Israel.
Email: {bshuval@, idotal@ee.}technion.ac.il

***Abstract*—A transform that is universally polarizing over a set of channels with memory is presented. Memory may be present in both the channel and its input. Both the encoder and the decoder are aware of the input distribution, which is fixed. Only the decoder is aware of the actual channel being used. The transform is used to design a universal code for this scenario. The code is to have vanishing error probability when used over any channel in the set, and achieve the infimal information rate over the set. Universal polarization is established under two key properties: memory in the form of an underlying hidden Markov state sequence that is aperiodic and irreducible and a new property: forgetfulness.**

I. Introduction

We present polarization-based codes with vanishing error probability universally over a set of channels *with memory*. The input distribution to all channels in the set is fixed and known at the encoder and decoder. The encoder only knows that the channel belongs to the set, while the decoder is aware of the actual channel. Examples of channels with memory are finite-state channels, input-constrained channels, and intersymbol-interference channels. Our codes approach the infimal information rate among the set under successive-cancellation (SC) decoding, provided that every input-output process in the set satisfies mild technical constraints. The error probability of our construction has the same exponent as Arıkan's polar codes [1]. To keep the paper focused, we concentrate on channel-coding; however, our results apply both to channel and source coding.

Polar coding for a *class* of *memoryless* channels with decoder-side channel knowledge was first considered in [2]; the paper showed that Arıkan's polar codes [1] under SC decoding cannot achieve the compound capacity of a set of binary-input, memoryless, and symmetric (BMS) channels. It was shown in [3, Prop. 7.1] that this is due to SC decoding. Nevertheless, polarization-based coding methods have been shown to yield universal codes. Two polariztion-based designs that achieve universality over a set of BMS channels were presented in [4]. Another design was presented in [5]. The construction of this paper is a generalization of that of [5].

We present our universal construction in Section III. It consists of two stages, a slow stage followed by a fast stage. Both are recursive and use Arıkan transforms as building blocks. The fast stage consists of multiple applications of Arıkan transforms as in [1]. The slow stage uses Arıkan transforms differently. When used over a set of BMS channels and specialized appropriately, this universal construction is functionally equivalent to the one presented in [5]. Our goal, however, is to use it over a set of processes with memory.

Polar codes were shown to achieve vanishing error probability for processes with memory in [6] and [7]. Combined, the results of [6] and [7] enable information-rate-achieving polar codes for processes with memory that have an underlying hidden Markov structure. A practical, low-complexity, decoding algorithm for such processes with memory was described in [8]. The decoding of our universal code is based on this algorithm.

In our universal setting, the encoder has partial information: it knows that the process belongs to some set of processes with memory. The exact process is known only to the decoder, at the time of decoding. The encoder must employ a code that achieves vanishing error probability for any process in the set. Additionally, the code is to have the highest possible rate over the entire set. Thus, the code is to approach the infimal information rate over the entire set. This is indeed what we achieve here. We show that our polarization-based construction is universal over sets of processes with memory. We prove universality when the sets contain processes with memory that satisfy two technical constraints, presented in Section IV. Briefly, the processes have an underlying hidden finite-state Markov structure that is regular (aperiodic and irreducible). Additionally, the processes must have a property we call *forgetfulness*, which was not needed in [6], [7].

Due to length constraints, proofs and other results are omitted. These can be found in the full version of our paper [9].

II. Basic Definitions

The definitions below capture channel and source coding jointly.

Definition 1 (s/o-pair)**.** A *symbol-observation pair* (*s/o-pair*), is a pair of dependent random variables X and Y; X is the *symbol* and Y is the *observation*. An s/o-pair with symbol X and observation Y is denoted $X \rightarrowtail Y$.

An s/o-pair is specified using the *joint* distribution $P_{X,Y}(x, y) = P_X(x)P_{Y|X}(y|x)$. This is in contrast to a channel that is specified using only $P_{Y|X}(y|x)$. A channel becomes an s/o-pair once the input distribution is specified.

Definition 2 (s/o-process)**.** A sequence of s/o-pairs $X_i \rightarrowtail Y_i$, $i = 1, 2, \ldots$ is called a *symbol-observation process* (*s/o-process*). We use the notation $X_\star \rightarrowtail Y_\star$. We assume throughout that s/o-processes are stationary. The *conditional entropy rate* of an s/o-process $X_\star \rightarrowtail Y_\star$ is $\mathcal{H}(X_\star|Y_\star) \triangleq \lim_{N\to\infty} \frac{1}{N} H(X_1^N|Y_1^N)$.

Definition 3 (s/o-block)**.** A sequence of N consecutive s/o-pairs of an s/o-process is called an *s/o-block*. We use the notation $X_1^N \rightarrowtail Y_1^N$. An s/o-block has a natural indexing: $X_j \rightarrowtail Y_j$ is s/o-pair j of s/o-block $X_1^N \rightarrowtail Y_1^N$.

Generally, s/o-pairs in an s/o-block are dependent, due to memory. By stationarity all s/o-pairs of an s/o-block are identically distributed. For simplicity, we assume that s/o-pairs have binary symbols, and observations over a finite alphabet.

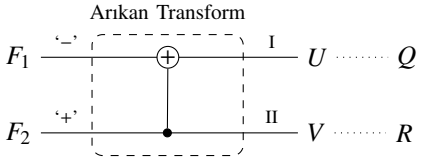

Fig. 1. An Arıkan transform transforms two input symbols, U (input-I) and V (input-II) to two output symbols, F_1 (output '−') and F_2 (output '+').

III. Universal Polar Transform

A. Overview of the Transform

The universal polar transform is a type of H-transform, a concept that we now define.

Definition 4 (H-transform)**.** An *H-transform* is a one-to-one and onto mapping f between two symbol vectors of length N. Moreover, when we say that s/o-block $X_1^N \rightarrowtail Y_1^N$ is transformed to s/o-block $F_1^N \rightarrowtail G_1^N$ by H-transform f, we mean that: (1) $F_1^N = f(X_1^N)$; (2) $G_i = (F_1^{i-1}, Y_1^N)$, for any i.

H-transforms are recursively defined. The recursive construction begins with an initial H-transform f_0 of length N_0. At step $n+1$, a step-$(n+1)$ H-transform is formed from two step-n H-transforms of consecutive symbol vectors. This generates a step-$(n+1)$ H-transform of a single, larger, symbol vector. A basic building block is the Arıkan transform [1], illustrated in Figure 1. It operates on two input symbols: input-I: U (with observation Q) and input-II: V (with observation R) and transforms them to two new symbols: a '−' symbol F_1 (with observation G_1) and a '+' symbol F_2 (with observation G_2), where $F_1 = U + V$, $G_1 = (Q, R)$ and $F_2 = V$, $G_2 = (F_1, Q, R)$.

Example 1. Arıkan's polar codes [1] are based on H-transforms. In this case, the mapping f is $F_1^N = f(X_1^N) = \mathsf{B}_N \mathsf{G}_2^{\otimes n} X_1^N$, where $N = 2^n$, B_N is the $N \times N$ bit-reversal matrix, $\mathsf{G}_2 = \left[\begin{smallmatrix} 1 & 0 \\ 1 & 1 \end{smallmatrix}\right]$, and $\otimes$ denotes a Kronecker product.

Consider an s/o-block $X_1^N \rightarrowtail Y_1^N$, with H-transform $F_1^N \rightarrowtail G_1^N$. We wish to recover the symbols X_1^N from Y_1^N. We denote the recovered symbols with a hat, ($\hat{\cdot}$). That is, $\hat{X}_1^N = \Phi(Y_1^N)$, where $\Phi(\cdot)$ is the algorithm for recovery. Rather than computing $\hat{X}_1^N$ from Y_1^N directly, we may compute $\hat{F}_1^N$ from Y_1^N. By the properties of H-transforms, there exists a mapping f^{-1}, such that $X_1^N = f^{-1}(F_1^N)$. Any algorithm for recovering F_1^N from Y_1^N is equivalent to an algorithm for recovering X_1^N from Y_1^N. For, if $\hat{F}_1^N = \Phi(Y_1^N)$ we can define $\hat{X}_1^N = f^{-1}(\hat{F}_1^N) = f^{-1}(\Phi(Y_1^N))$ and vice versa; clearly, $\mathbb{P}(\hat{F}_1^N \neq F_1^N) = \mathbb{P}(\hat{X}_1^N \neq X_1^N)$. We compute $\hat{F}_1^N$ sequentially. Let Φ_i be a maximum-likelihood decoder of F_i from G_i. At step i, we form $\hat{G}_i = (\hat{F}_1^{i-1}, Y_1^N)$ and decode $\hat{F}_i = \Phi_i(\hat{G}_i)$. This is tantamount to the successive-cancellation decoding described in [1].

Recall that the universal transform consists of a slow stage followed by a fast stage. Theorem 1, our main result, shows that the slow stage (referred to as BST and presented in Section III-B) is *monopolarizing*, a concept we now define.

Definition 5 (Monopolarizing H-transform)**.** Let $\eta > 0$ and let $\mathcal{L}, \mathcal{H} \subseteq \{1, 2, \ldots, N\}$ be two index sets. An H-transform f is $(\eta, \mathcal{L}, \mathcal{H})$-*monopolarizing* for a family of s/o-processes if for any s/o-block $X_1^N \rightarrowtail Y_1^N$ in the family, either $H(F_i|G_i) \leq \eta$ for all $i \in \mathcal{L}$ or $H(F_i|G_i) \geq 1-\eta$ for all $i \in \mathcal{H}$, where s/o-block $F_1^N \rightarrowtail G_1^N$ denotes the transformed s/o-block.

Theorem 1. *Let $X_\star \rightarrowtail Y_\star$ be a forgetful FAIM-derived s/o-process. For every $\eta > 0$ there exist L_0, M_0, and n_{th} such that if $n \geq n_{\text{th}}$ then a level-n BST initialized with parameters L_0 and M_0 is $(\eta, [\text{med}_+(n)], [\text{med}_-(n)])$-monopolarizing.*

Specifically, let $F_1^{N_n} \rightarrowtail G_1^{N_n}$ be a transformed s/o-block of a level-n BST initialized with L_0 and M_0 as above. Then:

- *if $\mathcal{H}(X_\star|Y_\star) \leq 1/2$ then $H(F_i|G_i) < \eta$, $\quad \forall i \in [\text{med}_+(n)]$;*
- *if $\mathcal{H}(X_\star|Y_\star) \geq 1/2$ then $H(F_i|G_i) > 1-\eta$, $\forall i \in [\text{med}_-(n)]$.*

The term 'forgetful FAIM-derived' and the parameters L_0, M_0, n_{th} will be made clear by Section IV. The sets $[\text{med}_+(n)]$ and $[\text{med}_-(n)]$, defined by the slow stage (see (1d), (1e), below), are of equal size. We now explain the theorem's importance.

Suppose we wish to design a universal code for a set of channels with memory, all with entropy rate less than $1/2$, and assume that the input distribution is uniform. A universal code is to have rate approaching $1/2$. We use Theorem 1 verbatim, and utilize the set $[\text{med}_+(n)]$ by appending to it the fast stage. We show in Lemma 3 that the fast stage polarizes fast. Thus, we achieve vanishing error probability for almost all indices in $[\text{med}_+(n)]$, resulting in a universal code approaching rate $1/2$.

If the universal code is to have a different rate, both sets $[\text{med}_+(n)]$ and $[\text{med}_-(n)]$ are utilized. E.g., if all channels in the set have entropy rate less than $1/4$, the desired universal code rate is $3/4$.[1] The set $[\text{med}_+(n)]$ already yields rate $1/2$; to increase the rate to $3/4$ we utilize $[\text{med}_-(n)]$. By applying a slow stage transform to $[\text{med}_-(n)]$, we generate two new (sub)sets of indices, half of which will have low entropy, which are added to the low entropy indices in $[\text{med}_+(n)]$ to obtain a code of rate $3/4$. This operation may be repeated multiple times, or in different combinations, to yield any desired rate.

B. Slow Polarization Stage

The slow stage transform is called a *basic slow transform* (BST). It is a generalization of the transform of [5, Section II].

The basic slow transform is constructed recursively. We call each construction step a *level*. Each level is an H-transform of length $N_n = 2L_n + M_n$, where L_n and M_n are specified in (2) below. The transformed s/o-block is a *level-n block*. We define the following index sets for a level-n block, $n \geq 0$.

$$[\text{lat}_1(n)] \triangleq \{i \mid \quad 1 \leq i \leq L_n\}, \tag{1a}$$

$$[\text{lat}_2(n)] \triangleq \{i \mid L_n + M_n + 1 \leq i \leq N_n\}, \tag{1b}$$

$$[\text{lat}(n)] \triangleq [\text{lat}_1(n)] \cup [\text{lat}_2(n)], \tag{1c}$$

$$[\text{med}_-(n)] \triangleq \{i \mid i = L_n + 2k - 1,\ 1 \leq k \leq M_n/2\}, \tag{1d}$$

$$[\text{med}_+(n)] \triangleq \{i \mid i = L_n + 2k, \quad 1 \leq k \leq M_n/2\}, \tag{1e}$$

$$[\text{med}(n)] \triangleq [\text{med}_-(n)] \cup [\text{med}_+(n)]. \tag{1f}$$

Symbol (or s/o-pair) i is lateral if $i \in [\text{lat}(n)]$; similarly, it is medial if $i \in [\text{med}(n)]$.

The construction is initialized with integer parameters L_0 and M_0. We assume that M_0 is even. The initial step f_0, which generates a level-0 block, is an H-transform of length $N_0 = 2L_0 + M_0$. We set f_0 as the identity mapping. Thus, the initial step transforms an s/o-block $X_1^{N_0} \rightarrowtail Y_1^{N_0}$ into an s/o-block $F_1^{N_0} \rightarrowtail G_1^{N_0}$, where, for $1 \leq i \leq N_0$,

$$F_i = X_i, \quad G_i = (F_1^{i-1}, Y_1^{N_0}).$$

[1] Another example is when the input distribution is not uniform, in which case a Honda-Yamamoto [10] scheme is used; see our full paper [9] for details.

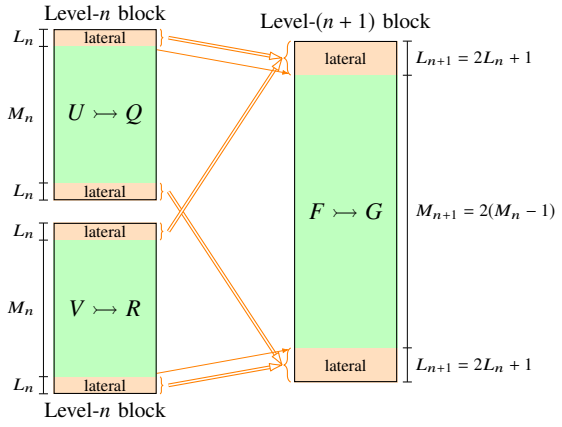

Fig. 2. A schematic description of forming lateral s/o-pairs of a level-$(n+1)$ block from two level-n blocks.

A level-$(n+1)$ BST is constructed from two level-n BSTs. Let f_n be a level-n BST. We define f_{n+1} in (3) and (4) below.

The BSTs of the two consecutive level-n s/o-blocks are

$$U_1^{N_n} = f_n(X_1^{N_n}), \quad Q_i = (U_1^{i-1}, Y_1^{N_n}), \quad 1 \le i \le N_n,$$
$$V_1^{N_n} = f_n(X_{N_n+1}^{2N_n}), \quad R_i = (V_1^{i-1}, Y_{N_n+1}^{2N_n}), \quad 1 \le i \le N_n.$$

The level-$(n+1)$ transformed s/o-block of length N_{n+1} is

$$F_1^{N_{n+1}} = f_{n+1}(X_1^{N_{n+1}}), \quad G_i = (F_1^{i-1}, Y_1^{N_{n+1}}), \quad 1 \le i \le N_{n+1}.$$

A level-$(n+1)$ block has length $N_{n+1} = 2L_{n+1} + M_{n+1}$, where

$$L_{n+1} = 2L_n + 1, \quad M_{n+1} = 2(M_n - 1). \tag{2}$$

Lateral symbols of a level-$(n+1)$ block are formed by renaming symbols of level-n s/o-pairs, as specified in (3). This is illustrated in Figure 2.

$$i \in [\mathrm{lat}(n+1)] \Rightarrow F_i = \begin{cases} U_j, & i = 2j-1, \\ V_j, & i = 2j. \end{cases} \tag{3}$$

All lateral symbols of the level-n blocks become lateral symbols of the level-$(n+1)$ block. Additionally, note that, by (1), (2), and (3), two medial symbols of the level-n blocks become lateral symbols of the level-$(n+1)$ block: $F_{L_{n+1}} = F_{2(L_n+1)-1} = U_{L_n+1}$ and $F_{L_{n+1}+M_{n+1}+1} = F_{2(L_n+M_n)} = V_{L_n+M_n}$.

Medial symbols of a level-$(n+1)$ block are formed using

$$i \in [\mathrm{med}(n+1)] \Rightarrow$$
$$F_i = \begin{cases} U_{j+1} + V_j, & i = 2j, \\ V_j, & i = 2j+1, \ j \in [\mathrm{med}_-(n)], \\ U_{j+1}, & i = 2j+1, \ j \in [\mathrm{med}_+(n)]. \end{cases} \tag{4}$$

As illustrated in Figure 3, medial symbols of a level-$(n+1)$ block are formed in pairs from medial symbols of level-n blocks using Arıkan transforms. Overall, $M_n - 1$ Arıkan transforms are performed. In each Arıkan transform, input-I is a symbol from $[\mathrm{med}_+(n)]$ of one level-n block and input-II is a symbol from $[\mathrm{med}_-(n)]$ of the other level-n block. The blocks *alternate* between successive Arıkan transforms: look at $F_{2L_n+2}, F_{2L_n+3}, F_{2L_n+4}$, and F_{2L_n+5} in Figure 3.

The fraction of medial symbols out of all symbols in a level-n block can be made arbitrary close to 1. Denoting this fraction by $\alpha_n = \frac{M_n}{2L_n+M_n}$, we have the following.

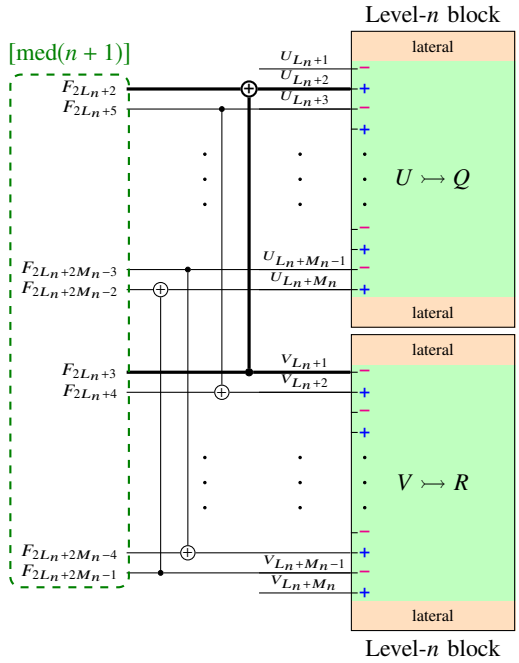

Fig. 3. Forming medial symbols of level $n+1$ of the BST. Arıkan transforms are used with a symbol from $[\mathrm{med}_+(n)]$ of one block as input-I and a symbol from $[\mathrm{med}_-(n)]$ of the other block as input-II. One Arıkan transform is highlighted using thicker edges.

Lemma 2. *Initialize a BST with parameters $L_0 \ge 0$ and M_0. Let $0 < \alpha < 1$. If $M_0 \ge \left\lceil \frac{2(1+\alpha L_0)}{1-\alpha} \right\rceil$, then $\alpha_n \ge \alpha$ for any $n \ge 0$.*

Discussion. The BST is a generalization of the Şaşoğlu-Wang transform (SWT) of [5]. In the memoryless case, it can be shown that the SWT and BST (with $L_0 = 0$) have the same performance. We show in Section IV that the BST can also be used for processes with memory, by taking $L_0 > 0$.

C. Fast Polarization Stage

We will show in Section IV that the BST is $(\eta, \mathcal{L}, \mathcal{H})$-monopolarizing for a suitable family of s/o-processes with memory. However, even without memory [5], monopolarization is too slow to enable an SC decoder to succeed. Hence, as in [5], we append to the BST a fast polarization stage. This is illustrated in Figure 4 for a channel-coding setting. Namely, in the fast stage, we make $\hat{N} = 2^{\hat{n}}$ copies of a length-N BST, and apply to them $|\mathcal{L}|$ Arıkan transforms. The jth Arıkan transform operates on the jth medial s/o-pair from $\mathcal{L}$ from each of the BSTs.

In [9, Appendix A], we prove the following lemma.

Lemma 3. *Let $B_1, B_2, \ldots$ be independent and identically distributed random variables with $\mathbb{P}(B_i = 0) = \mathbb{P}(B_i = 1) = 1/2$. Let $Z_0, Z_1, \ldots$ be a $[0,1]$-valued random process that satisfies $Z_{n+1} \le \kappa Z_n^{2-B_i}$ for some $\kappa > 1$. Fix $0 < \beta < 1/2$. Then, for every $\delta > 0$ there exist $\eta > 0$ and n_0 such that if $Z_0 \le \eta$ then for every $0 < \beta < 1/2$, we have*

$$\mathbb{P}\left(Z_n \le 2^{-2^{n\beta}} \text{ for all } n \ge n_0\right) \ge 1 - \delta. \tag{5}$$

Crucially, η and n_0 depend on the Z_n only through κ. Note that we do *not* assume here that Z_n converges almost surely.

The Bhattacharyya parameter Z_n of a randomly-selected s/o-pair in an Arıkan transform satisfies an inequality precisely as in the lemma [1], even under memory [6]. Thus, with high

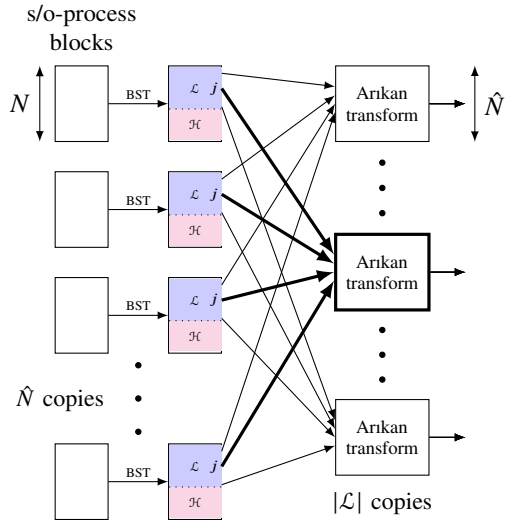

Fig. 4. The fast stage for a channel-coding application. First, perform $\hat{N}$ length-N BSTs. Then, apply $|\mathcal{L}|$ length-$\hat{N}$ Arıkan trasforms. The jth Arıkan transform (in bold) operates on the jth medial s/o-pair in $\mathcal{L}$ from each BST.

probability the Bhattacharyya parameter of any s/o-pair after the fast stage is less than $2^{-\hat{N}^\beta}$ for a fixed $\beta < 1/2$. This enables coding with probability of error upper bounded by $N\hat{N}2^{-\hat{N}^\beta}$, which vanishes as $\hat{N}$ increases, at negligible rate loss.

D. Decoding

The universal polar codes consist of a concatenation of the BST and Arıkan's polar codes. Thus, they consist of recursive applications of Arıkan transforms, which can be decoded efficiently using SC decoding, where both stages are decoded in lockstep. Due to memory in the s/o-process, the variation of SC decoding of [8] is used. The overall universal polar code length is $N \cdot \hat{N}$, so, by [8, Theorem 2], the decoding complexity is $O(|\mathcal{S}|^3 N\hat{N} \cdot \log(N\hat{N}))$. The paremeter $|\mathcal{S}|$ is defined below.

IV. The BST is Monopolarizing

We prove that the BST is monopolarizing for s/o-processes whose distribution depends on an underlying Markov sequence, $S_j, j \in \mathbb{Z}$. We assume throughout that, for any j, X_j is binary, $Y_j \in \mathcal{Y}$, and $S_j \in \mathcal{S}$, where $\mathcal{Y}, \mathcal{S}$ are finite alphabets.

Definition 6 (FAIM process)**.** A strictly stationary process (S_j, X_j, Y_j), $j \in \mathbb{Z}$ is called a *Finite-State, Aperiodic, Irreducible, Markov* (FAIM) process if, for any any j,

$$P_{S_j,X_j,Y_j|S_{-\infty}^{j-1},X_{-\infty}^{j-1},Y_{-\infty}^{j-1}} = P_{S_j,X_j,Y_j|S_{j-1}} = P_{S_j|S_{j-1}} \cdot P_{X_j,Y_j|S_j}, \quad (6)$$

and $S_j, j \in \mathbb{Z}$ is a finite-state, homogeneous, irreducible, and aperiodic stationary Markov chain.

A *FAIM-derived s/o-process* is an s/o-process whose joint distribution is derived from a FAIM process (S_j, X_j, Y_j).

Definition 7 (Forgetful FAIM process)**.** A FAIM process (S_j, X_j, Y_j), $j \in \mathbb{Z}$ is said to be *forgetful* if for any $\epsilon > 0$ there exists a natural number λ such that if $k \geq \lambda$ then

$$I(S_1; S_k|X_1^k, Y_1^k) \leq \epsilon, \quad (7a)$$

$$I(S_1; S_k|Y_1^k) \leq \epsilon. \quad (7b)$$

A FAIM-derived s/o-process $X_\star \rightarrowtail Y_\star$ is *forgetful* if it is derived from a forgetful FAIM process.

Note. Both (7a) and (7b) are required: neither implies the other.

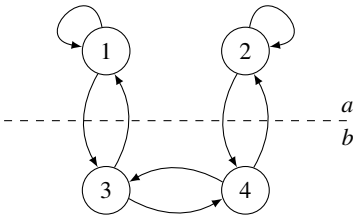

Fig. 5. The Markov chain S_j has four states. The possible transitions are depicted using arrows; the probability of choosing any transition is 1/2.

Example 2 shows that a FAIM process need not to be forgetful; see Section V for a sufficient condition for forgetfulness.

Example 2. This example is due to [11, Section 10]. In Figure 5 we illustrate the process (S_j, Y_j). Arrows depict the possible state transitions, all with probability 1/2. The observation is $Y_j = a$ if $S_j \in \{1, 2\}$ and $Y_j = b$ if $S_j \in \{3, 4\}$. In this example we will not be interested in X_j. This process is FAIM since S_j is a finite-state regular Markov chain.

Observe that given S_1 and the sequence Y_1^k, one can track the state and determine S_k precisely. E.g., if $S_1 = 1$ then $Y_1 = a$. If $Y_2 = b$ this implies that $S_2 = 3$, and so on. Thus, $I(S_1; S_k|Y_1^k)$ cannot vanish with k, so this process is not forgetful.

At this point, all required definitions for the statement of Theorem 1 have been presented.

Proof sketch for Theorem 1: We define a variation of the BST, the observation-truncated BST (OT-BST), in which the transformed observation G_i is replaced with a truncated version. We show that the OT-BST monopolarizes for s/o-processes that are independent in blocks of length $N_0 = 2L_0 + M_0$. Then, we show that due to forgetfulness and the FAIM property, one can set L_0 and M_0 such that the difference between the entropy of a transformed index of the BST applied to the original process and that of a transformed index of the OT-BST applied to the block-independent process is negligible. Thus, monopolarization is ensured. Bounds on the number of BST levels required to ensure a given η are also developed. ■

V. A Sufficient Condition for Forgetfulness

A hidden Markov model (HMM) is a process (A_n, B_n) such that $A_n \in \mathcal{A}$ is a homogeneous Markov chain and $B_n \in \mathcal{B}$ is an observation that is a function of A_n. We assume that $|\mathcal{A}|, |\mathcal{B}| < \infty$. Without loss of generality [9, Appendix E], B_n is a *deterministic* function of A_n. A FAIM process is equivalent to an HMM with $A_n = (S_n, X_n, Y_n)$ and $B_n = (X_n, Y_n)$.

The transition matrix of A_n is M, assumed aperiodic and irreducible. Thus, $(\mathsf{M})_{i,j} = \mathbb{P}(A_n = j|A_{n-1} = i)$. We denote by $\mathsf{M}(b)$, for $b \in \mathcal{B}$, the matrix whose elements are $(\mathsf{M}(b))_{i,j} = \mathbb{P}(A_n = j, B_n = b|A_{n-1} = i)$. Observe that $\mathsf{M} = \sum_{b\in\mathcal{B}} \mathsf{M}(b)$. For a sequence of observations b_r^s we denote $\mathsf{M}(b_r^s) = \mathsf{M}(b_r)\mathsf{M}(b_{r+1}) \cdots \mathsf{M}(b_s)$.

Definition 8. A nonnegative matrix M is called *subrectangular* if $(\mathsf{M})_{i,j} \neq 0$ and $(\mathsf{M})_{k,l} \neq 0$ imply $(\mathsf{M})_{i,l} \neq 0$ and $(\mathsf{M})_{k,j} \neq 0$.

The support $\sigma(\mathbf{x})$ of a vector $\mathbf{x}$ is its set of nonzero indices: $\sigma(\mathbf{x}) = \{i \mid x_i \neq 0\}$. If $\mathbf{x}$ and $\mathbf{y}$ are nonnegative vectors with $\sigma(\mathbf{x}) = \sigma(\mathbf{y})$, the projective distance between them is

$$d(\mathbf{x}, \mathbf{y}) = \max_{j,l\in\sigma(\mathbf{x})} \ln \frac{x_j y_l}{x_l y_j}.$$

It can be shown [9, Lemma 24] that if M is subrectangular then $\sigma(\mathbf{x}^T\mathsf{M}) = \sigma(\mathbf{y}^T\mathsf{M})$.

Definition 9. The *Birkhoff contraction coefficient* [12] $\tau(\mathsf{M})$ of a nonnegative matrix M is defined as

$$\tau(\mathsf{M}) = \begin{cases} 0, & \mathsf{M} = 0, \\ \sup\limits_{\mathbf{x}>0,\,\mathbf{y}>0} \dfrac{d(\mathbf{x}^T\mathsf{M}, \mathbf{y}^T\mathsf{M})}{d(\mathbf{x},\mathbf{y})}, & \mathsf{M} \text{ subrectangular, } \mathsf{M} \neq 0, \\ 1, & \text{otherwise,} \end{cases}$$

with the convention $0/0 = 0$. If M is subrectangular, $\tau(\mathsf{M}) < 1$. This fact will be crucial later.

We prove the following lemma in [9]. Here, $\|\mathbf{x}\|_1 = \sum_i |x_i|$.

Lemma 4. *Let $\mathsf{M}_1, \mathsf{M}_2, \ldots, \mathsf{M}_m$ be a sequence of square nonzero subrectangular matrices, and let T, as well as $\mathsf{T}_1, \mathsf{T}_2, \ldots, \mathsf{T}_m$ be square nonnegative nonzero matrices. Denote $\mathsf{R} = \mathsf{T}_1\mathsf{M}_1\mathsf{T}_2\mathsf{M}_2\cdots\mathsf{T}_m\mathsf{M}_m$. Then, for any two nonnegative vectors $\mathbf{x}, \mathbf{y}$ such that $\left\|\mathbf{x}^T\mathsf{R}\mathsf{T}\right\|_1 > 0$ and $\left\|\mathbf{y}^T\mathsf{R}\mathsf{T}\right\|_1 > 0$ we have*

$$\log\left(\frac{\left\|\mathbf{x}^T\mathsf{R}\mathsf{T}\right\|_1}{\left\|\mathbf{y}^T\mathsf{R}\mathsf{T}\right\|_1} \cdot \frac{\left\|\mathbf{y}^T\mathsf{R}\right\|_1}{\left\|\mathbf{x}^T\mathsf{R}\right\|_1}\right) \leq 4\log\left(\frac{1+\tau(\mathsf{M}_1)}{1-\tau(\mathsf{M}_1)}\right) \cdot \prod_{\ell=2}^{m} \tau(\mathsf{M}_\ell).$$

This lemma is useful because $I(A_0; A_{n+1}|B_1^n)$ can be expressed as the expectation of an expression similar to the left-hand side of the inequality above. Namely [9, Eq. 93],

$$I(A_0; A_{n+1}|B_1^n) = \mathbb{E}\left[\log\left(\frac{\left\|\mathbf{e}_{A_0}^T\mathsf{M}(B_1^n)\mathsf{T}_{A_{n+1}}\right\|_1}{\left\|\boldsymbol{\pi}^T\mathsf{M}(B_1^n)\mathsf{T}_{A_{n+1}}\right\|_1} \cdot \frac{\left\|\boldsymbol{\pi}^T\mathsf{M}(B_1^n)\right\|_1}{\left\|\mathbf{e}_{A_0}^T\mathsf{M}(B_1^n)\right\|_1}\right)\right], \tag{8}$$

where $\mathbf{e}_a$ is a unit vector with 1 in position a and zeros otherwise, $\boldsymbol{\pi}$ is the stationary distribution of A_n, and T_a is an $|\mathcal{A}| \times |\mathcal{A}|$ matrix with $(\mathsf{T}_a)_{a,a} = (\mathsf{M})_{a,a}$ and zeros otherwise.

Our sufficient condition is based on the following condition, named in honor of Prof. Thomas Kaijser [11].

Condition K. The HMM (A_n, B_n) is characterized by matrices $\mathsf{M}(b)$, $b \in \mathcal{B}$ such that:

1) The matrix $\mathsf{M} = \sum_{b\in\mathcal{B}} \mathsf{M}(b)$ is aperiodic and irreducible.
2) There exists an ordered sequence $\beta_1, \beta_2, \ldots, \beta_l$ of elements of $\mathcal{B}$ such that the matrix $\mathsf{M}(\beta_1^l) = \mathsf{M}(\beta_1)\mathsf{M}(\beta_2)\cdots\mathsf{M}(\beta_l)$ is nonzero and subrectangular.

An important consequence of Condition K is the following.

Lemma 5. *If the HMM (A_n, B_n) satisfies Condition K then there exist a positive integer $n_\star$ and constants $\delta_\star < 1$ and $0 \leq \tau_\star < 1$ such that*

$$\mathbb{P}\left(\tau(\mathsf{M}(B_1^{n_\star})) \leq \tau_\star | A_0 = a_0\right) \geq 1 - \delta_\star, \quad \forall a_0 \in \mathcal{A}. \tag{9}$$

An HMM that satisfies (9) is called an $(n_\star, \delta_\star, \tau_\star)$-KHMM. The following proposition holds for any KHMM.

Proposition 6. *Let (A_n, B_n) be an $(n_\star, \delta_\star, \tau_\star)$-KHMM with $\delta_\star > 0$. Denote $\gamma = 1/\delta_\star$, $\alpha = \gamma \cdot \log|\mathcal{A}|$, $\rho = \delta_\star^{1/n_\star} < 1$. Then, for any $m \leq n$ we have*

$$I(A_0; A_{n+1}|B_1^n) \leq 4\log\left(\frac{1+\tau_\star}{1-\tau_\star}\right)\tau_\star^m + \alpha\frac{(\gamma n)^m}{m!}\rho^{n+1}. \tag{10}$$

Proof sketch: By (8), there exists a random variable J such that $I(A_0; A_{n+1}|B_1^n) = \mathbb{E}[J]$. Consider the matrix product $\mathsf{M}(B_1^n)$. In this product, we denote by D_n the number of non-overlapping occurrences of contiguous sequences of matrices whose product has Birkhoff contraction coefficient at most $\tau_\star$. Clearly, D_n is uniquely defined by B_1^n. We thus have

$$\begin{aligned} &I(A_0; A_{n+1}|B_1^n) = \mathbb{E}[J] \\ &= \mathbb{E}[J|D_n \leq m]\,\mathbb{P}(D_n \leq m) + \mathbb{E}[J|D_n > m]\,\mathbb{P}(D_n > m). \end{aligned}$$

To obtain (10), we upper-bound each right-hand summand. Regularity of A_n and (9) yield $\mathbb{E}[J|D_n \leq m]\,\mathbb{P}(D_n \leq m) \leq \log|\mathcal{A}|\gamma\frac{(n\gamma)^m}{m!}\rho^{n+1}$. Next, using (8) and Lemma 4, we obtain $\mathbb{E}[J|D_n > m]\,\mathbb{P}(D_n > m) \leq 4\log\left(\frac{1+\tau_\star}{1-\tau_\star}\right)\cdot\tau_\star$. ∎

Our sufficient condition follows from the following theorem.

Theorem 7. *Suppose the HMM (A_n, B_n) satisfies Condition K. Then, for every $\epsilon > 0$ there exists an integer λ such that if $n \geq \lambda$ then $I(A_0; A_{n+1}|B_1^n) \leq \epsilon$.*

Proof sketch: By Lemma 5, (A_n, B_n) is an $(n_\star, \delta_\star, \tau_\star)$-KHMM for some $n_\star$, $\delta_\star$, $\tau_\star$. *Case 1:* If $\delta_\star > 0$, Proposition 6 holds. Set $n = \lambda$ such that each term on the right-hand side of (10) is upper-bounded by $\epsilon/2$; this is possible since $\rho, \tau_\star < 1$. *Case 2:* If $\delta_\star = 0$, an expression similar to (10) holds, with the right-hand side containing only the first term. Set $n = \lambda$ such that it is upper-bounded by ϵ; this is possible since $\tau_\star < 1$. ∎

Using the data processing inequality, one can obtain:

Corollary 8. *Suppose the HMM (A_n, B_n) satisfies Condition K. Then, for every $\epsilon > 0$ there exists an integer λ such that if $n \geq \lambda$ then $I(A_1; A_n|B_1^n) \leq \epsilon$.*

Sufficient Condition: A FAIM process is equivalent to an HMM with $A_n = (S_n, X_n, Y_n)$ and $B_n = (X_n, Y_n)$. Further denote $C_n = Y_n$. Both (A_n, B_n) and (A_n, C_n) are HMMs. If both satisfy Condition K then Corollary 8 holds for either, implying that that FAIM process (S_n, X_n, Y_n) is forgetful (see Definition 7). Thus, a sufficient condition for forgetfulness is that both HMMs (A_n, B_n) and (A_n, C_n) satsify Condition K.

References

[1] E. Arıkan, "Channel polarization: a method for constructing capacity-achieving codes for symmetric binary-input memoryless channels," *IEEE Trans. on Inf. Theory*, vol. 55, no. 7, pp. 3051–3073, July 2009.

[2] S. H. Hassani, S. B. Korada, and R. Urbanke, "The compound capacity of polar codes," in *2009 47th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, Sept 2009, pp. 16–21.

[3] E. Şaşoğlu, "Polar Coding Theorems for Discrete Systems," Ph.D. dissertation, School Comput. Commun. Sci., EPFL, Lausanne, Switzerland, 2011.

[4] S. H. Hassani and R. Urbanke, "Universal polar codes," in *2014 IEEE Int. Symp. on Inf. Theory*, June 2014, pp. 1451–1455.

[5] E. Şaşoğlu and L. Wang, "Universal polarization," *IEEE Trans. on Inf. Theory*, vol. 62, no. 6, pp. 2937–2946, June 2016.

[6] E. Şaşoğlu and I. Tal, "Polar coding for processes with memory," *IEEE Trans. on Inf. Theory*, vol. 65, no. 4, pp. 1993–2003, April 2019.

[7] B. Shuval and I. Tal, "Fast polarization for processes with memory," *IEEE Trans. on Inf. Theory*, vol. 65, no. 4, pp. 2004–2020, April 2019.

[8] R. Wang, J. Honda, H. Yamamoto, R. Liu, and Y. Hou, "Construction of polar codes for channels with memory," in *2015 IEEE Information Theory Workshop*, October 2015, pp. 187–191.

[9] B. Shuval and I. Tal, "Universal polarization for processes with memory," 2018. [Online]. Available: arXiv:1811.05727

[10] J. Honda and H. Yamamoto, "Polar coding without alphabet extension for asymmetric models," *IEEE Trans. on Inf. Theory*, vol. 59, no. 12, pp. 7829–7838, December 2013.

[11] T. Kaijser, "A limit theorem for partially observed Markov chains," *The Annals of Probability*, vol. 3, no. 4, pp. 677–696, 08 1975.

[12] E. Seneta, *Non-negative matrices and Markov chains*, ser. Springer series in statistics. New York, NY: Springer, 2006.