Type Classes of Context Trees

Alvaro Martín, Gadiel Seroussi, Marcelo J. Weinberger

HP Laboratories
HPL-2012-10

Keyword(s):

context trees; method of types; enumeration; Markov chains; data compression

Abstract:

It is well known that a tree model does not always admit a finite-state machine (FSM) representation with the same (minimal) number of parameters. Therefore, known characterizations of type classes for FSMs do not apply, in general, to tree models. In this paper, the type class of a sequence with respect to a given context tree T is studied. An exact formula is derived for the size of the class, extending Whittle's formula for type classes with respect to FSMs. The derivation is more intricate than in the FSM case, since some basic properties of FSM types do not hold in general for tree types. The derivation also yields an efficient enumeration of the tree type class. A formula for the number of type classes with respect to T is also derived. The formula is asymptotically tight up to a multiplicative constant and also extends the corresponding result for FSMs. The asymptotic behavior of the number of type classes, and of the size of a class, are expressed in terms of the so-called minimal canonical extension of T, a tree that is generally larger than T but smaller than its FSM closure.

External Posting Date: January 21, 2012 [Fulltext] Approved for External Publication
Internal Posting Date: January 21, 2012 [Fulltext]

Type Classes of Context Trees*

Álvaro Martín†
Instituto de Computación, Universidad de la República
Montevideo, Uruguay
Email: almartin@fing.edu.uy

Gadiel Seroussi
Universidad de la República, Montevideo, Uruguay
and Hewlett-Packard Laboratories
Palo Alto, CA 94304, USA
Email: gseroussi@ieee.org

Marcelo J. Weinberger
Hewlett-Packard Laboratories
Palo Alto, CA 94304, USA
Email: marcelo.weinberger@hp.com

Abstract

It is well known that a tree model does not always admit a finite-state machine (FSM) representation with the same (minimal) number of parameters. Therefore, known characterizations of type classes for FSMs do not apply, in general, to tree models. In this paper, the type class of a sequence with respect to a given context tree T is studied. An exact formula is derived for the size of the class, extending Whittle's formula for type classes with respect to FSMs. The derivation is more intricate than in the FSM case, since some basic properties of FSM types do not hold in general for tree types. The derivation also yields an efficient enumeration of the tree type class. A formula for the number of type classes with respect to T is also derived. The formula is asymptotically tight up to a multiplicative constant and also extends the corresponding result for FSMs. The asymptotic behavior of the number of type classes, and of the size of a class, are expressed in terms of the so-called *minimal canonical extension* of T, a tree that is generally larger than T but smaller than its FSM closure.

*This paper was presented in part at the 2007 International Symposium on Information Theory (ISIT'07), Nice, France, 2007.

†Work done in part at HP Labs, Palo Alto, CA, and was supported by grant PDT - S/C/IF/63/147.

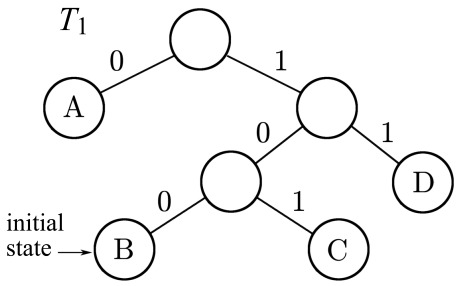

Fig. 1. A tree T_1 over $\mathcal{A} = \{0, 1\}$. For conciseness, we use the labels A,B,C,D for the leaves $0, 100, 101, 11$, respectively

I. Introduction

In the *method of types* [1] the set of sequences of a given length n over a finite alphabet $\mathcal{A}$ is partitioned into *type classes*, where two sequences belong to the same class if and only if every probability distribution in a certain family $\mathcal{P}$ assigns both sequences the same probability.[1] For a parametric family $\mathcal{P}$, a type class comprises all the sequences that are equiprobable under any value of the model parameter, e.g., all the sequences that yield the same vector of state-conditioned empirical distributions for a given *finite-state machine* (FSM) [3]. Applications of the method in hypothesis testing, channel coding, source coding, rate-distortion theory, and other areas are surveyed in [2]. Although the seminal reference [1] focuses on memoryless models, type classes and their applications have been studied for a variety of statistical models, such as finite memory (Markov) models (cf. [4], [5], [2], [6]) and FSM models (cf. [7]).[2] More recently, applications of the method of types to universal simulation for finite parametric models were presented in [8], and generalizations of the notion of type that preserve statistics, in an asymptotic sense, simultaneously for every Markov order k were presented in [9], [10].

Tree models [11], [12], [13] have also been extensively studied, as valuable tools in data compression and other applications in information theory and statistics (cf. [11], [12], [13], [14], [15], [16]). Roughly speaking, a tree model $\langle T, p_T \rangle$ consists of a full α-ary (context) tree T,[3] where α is the size of the finite

[1]Type classes were defined in terms of empirical distributions for memoryless models in [1]. The more general definition of type class used here was introduced in [2, Sec. VII], where extensions of the method of types to wider model families are considered.

[2]We make the distinction between *finite memory* Markov models of a given order k and the more general FSM models, whose state sequences are *Markov chains*, but which do not necessarily have the finite-memory property with respect to the original sequence alphabet [3].

[3]We say that an α-ary tree T is *full* if every internal node of T has exactly α children.

alphabet $\mathcal{A}$, and a set p_T of conditional probability distributions on $\mathcal{A}$, one associated with each leaf of the tree. Each edge of the tree is labeled with a symbol from $\mathcal{A}$. Every sufficiently long string, x^i, determines a unique leaf in the tree by descending from the root, matching the labels of the edges with the symbols in the string, starting from the last symbol and progressing in reverse order, until a leaf is reached. The probability of the next symbol of the string, x_{i+1}, given all the past, x^i, is determined by the conditional probability distribution associated to that leaf. The set of leaves of T, denoted S_T, is referred to as the set of *states* of the model. In the tree T_1 of Figure 1, all strings ending with the symbol 0 select the state A, while strings ending with 001 select the state B. By the model definition, the type class of a sequence x^n with respect to a tree is the set of sequences that have the same state-conditioned symbol occurrence counts as x^n (with an appropriate convention for the initial states).

In this paper, we study type classes for context trees. We derive a precise formula for the cardinality of a given class, and an asymptotic estimate (tight up to a multiplicative constant) of the number of classes. The derivation also yields an efficient enumeration scheme for type classes with respect to a tree T. This enumeration enables applications of tree models in enumerative source coding [17] and universal simulation [10]. In the former application, a sequence is encoded by first describing the type class it belongs to and then the index of the sequence within its type class according to an enumeration scheme. Thus, the number of type classes is related to the first part of the code, while the size of a class is related to the second part. In the case of universal simulation, as proposed in [10], a simulated sequence is generated by selecting at random from the type class of a training sequence, for which, again, an efficient enumeration scheme of type classes is instrumental. The logarithm of the size of a type class determines, in this case, the conditional entropy of the simulator output given the training sequence.

We say that a tree defines a *next-state function* if for every sufficiently long sequence x^i, the state selected by x^{i-1}, together with the symbol x_i, uniquely determine the state selected by x^i. We refer to such a tree also as an *FSM tree*, since it defines a (deterministic) finite-state machine [3]. For example, a tree whose leaves are all at the same depth k (corresponding to a Markov model of order k), is FSM. For FSM trees (and, in fact, for the significantly broader class of Markov chains defined on an arbitrary FSM), the size of a type class is given precisely by Whittle's formula [18]. Given the set of states $\mathcal{S}$ of a Markov chain, and a *sequence of states* $\mathbf{s} = s_0, s_1, \ldots, s_n$, let $(N_{\mathcal{S}})_{s,s'}$ denote the number of times there is a transition from s to s' in $\mathbf{s}$, and let $(N_{\mathcal{S}})_{s*} = \sum_{s' \in \mathcal{S}} (N_{\mathcal{S}})_{s,s'}$. Also, let $\hat{N}_{\mathcal{S}}$ denote the matrix obtained by normalizing each non-zero row of $N_{\mathcal{S}}$ (regarded as an $|\mathcal{S}| \times |\mathcal{S}|$ matrix) so that the row's entries sum to one. Consider the set $\mathcal{T}_{\mathcal{S}}(\mathbf{s})$ of state sequences that yield the same transition counts as $\mathbf{s}$.

Whittle's formula expresses the cardinality of $\mathcal{T}_{\mathcal{S}}(\mathbf{s})$ as

$$|\mathcal{T}_{\mathcal{S}}(\mathbf{s})| = M_{\mathcal{S}} \frac{\prod_{s\in\mathcal{S}}(N_{\mathcal{S}})_{s*}!}{\prod_{s,s'\in\mathcal{S}}(N_{\mathcal{S}})_{s,s'}!}\ , \tag{1}$$

where $M_{\mathcal{S}}$ denotes a well defined cofactor of $I-\hat{N}_{\mathcal{S}}$, with I denoting the $|\mathcal{S}|\times|\mathcal{S}|$ identity matrix. Clearly, when $\mathcal{S}$ corresponds to the set of states of an FSM tree T, and $\mathbf{s}$ is the state sequence obtained as the string of symbols x^n drives transitions in T, the set $\mathcal{T}_{\mathcal{S}}(\mathbf{s})$ is in one-to-one correspondence with the type class of x^n with respect to T. Formula (1) was also derived, using different methodologies, in [19] and [20].

It is well known, however, that arbitrary trees do not always define a next-state function, and Whittle's formula may not be directly applicable for general trees. For example, in the tree T_1 of Figure 1, the occurrence of symbol 1 in state A does not determine whether the next state will be B or C; we say that there is *loss of context* in state A. A characterization of FSM trees, and of the smallest FSM extension of a tree (called its *FSM closure*), is presented in [14], [15]. Generally, the type class of a sequence with respect to the FSM closure of T (whose size can be computed directly by means of Whittle's formula), will be smaller than the type class with respect to T, so the FSM closure does not provide a direct solution to the problem at hand.

Loss of context is of crucial importance when determining the size of a tree type class, as it limits the freedom to "rearrange" state transitions while remaining in the same type class. To illustrate this effect, we first observe (through application of Stirling's approximation to Whittle's formula) that the size of the type class of x^n relative to an FSM model behaves asymptotically as $2^{n\hat{\mathcal{H}}_{\mathcal{F}}(x^n)}$ for every sequence x^n,[4] where $\hat{\mathcal{H}}_{\mathcal{F}}(x^n)$ is the normalized empirical entropy of x^n with respect to the FSM $\mathcal{F}$ underlying the model.[5] For a tree T, $2^{n\hat{\mathcal{H}}_T(x^n)}$ is still an upper bound on the size of the type class (since the total probability of the type class with respect to its ML distribution is upper-bounded by 1), but the bound is not always asymptotically tight, as shown in Example 1 below. On the other hand, as mentioned above, the size of the type class with respect to T is lower-bounded by the size of the type class with respect to the FSM closure, T_F, of T, which does behave asymptotically as $2^{n\hat{\mathcal{H}}_{T_F}(x^n)}$.

[4] All explicit and implicit logarithms are taken to base 2.

[5] The normalized empirical entropy of x^n with respect to an FSM $\mathcal{F}$ with set of states $S_{\mathcal{F}}$ is $\hat{\mathcal{H}}_{\mathcal{F}}(x^n) = \sum_{s\in S_{\mathcal{F}}} \hat{p}(s)H(\hat{p}(\cdot|s))$, where $\hat{p}(\cdot|s)$ is the probability distribution over $\mathcal{A}$ defined as $\hat{p}(a|s) = (n_{\mathcal{F}})_s^{(a)}/(N_{\mathcal{F}})_{s*}$, with $(n_{\mathcal{F}})_s^{(a)}$ denoting number of emissions of symbol a in state s and $(N_{\mathcal{F}})_{s*}$ denoting the total number of symbols emitted in state s, and where $\hat{p}(s) = (N_{\mathcal{F}})_{s*}/n$. The normalized empirical entropy $\hat{\mathcal{H}}_T(x^n)$ with respect to a tree T is similarly defined, relative to the state set S_T. In either case, the normalized empirical entropy is equal to $-\frac{1}{n}\log\hat{P}_{\mathrm{ML}}(x^n)$, where $\hat{P}_{\mathrm{ML}}(x^n)$ is the *maximum likelihood* probability of x^n with respect to the structure, $\mathcal{F}$ or T, of interest.

Example 1: Consider the tree T_1 in Figure 1. The normalized empirical entropy of the sequence $x^n = 001\,001\ldots001$ with respect to T_1 is $\hat{\mathcal{H}}_{T_1}(x^n) = \frac{2}{3}h(\frac{1}{2})$ where h is the binary entropy function. The factor $h(\frac{1}{2})$ in $\hat{\mathcal{H}}_{T_1}(x^n)$ arises from state A, where half of the occurring symbols are 0 and half are 1 (we write $h(\frac{1}{2})$ unevaluated to emphasize this fact). On the other hand, in order to reach state $s_i = \mathrm{B}$, we must have $x_{i-2} = x_{i-1} = 0$ and, hence, A→A→B is a state sequence that *must* be followed to reach B. Therefore, to preserve conditional counts, a sequence in the same type class as x^n must follow the fixed state transition cycle A→A→B→A→A→B→…, and, thus, the type class of x^n is just $\{x^n\}$. The FSM closure, T_F, of T, is obtained by extending state A to states 00 and 01. It is readily verified that, in this case, we have $\hat{\mathcal{H}}_{T_F}(x^n) = 0$.

Example 1 can be seen as an extreme case of restrictions on state transitions that may, in general, rule out many of the state sequences that could be obtained by picking the next symbol at each state freely, according to the prescribed counts of the type class. Such freedom is already limited to some extent in the FSM case, as expressed by the cofactor $M_{\mathcal{S}}$ in (1). This cofactor, however, is polynomial in n, and thus negligible with respect to the main factor of the formula. As shown in the example, the reduction may be far more significant in the case of non-FSM trees, where the restrictions are more intricate. Thus, when studying the size of type classes with respect to arbitrary trees, instead of dealing only with (and counting) state transitions, we define, in Section II-B, a set of *pseudo states*, $\tilde{S}_T$, which includes and generally extends the original set of states S_T. Pseudo-states preserve some of the context information that is lost in the state transitions. Just as x^n uniquely determines a state sequence with respect to T, it will also uniquely determine a pseudo-state sequence, which, however, may be of length greater than n (as there may be pseudo-state transitions that are not associated with the emission of symbols in x^n).

With these tools on hand, we present, in Section II-C, our main result on size and enumeration of context tree type classes. We state this result (Theorem 1) for a restricted notion of type classes, which we call *close-ended*. We establish an explicit bijection between the close-ended type class of x^n and the set of sequences over $\tilde{S}_T$ that yield the same transition counts as the pseudo-state sequence of x^n, regarded as a realization of a Markov chain over $\tilde{S}_T$. The size of the type class in terms of Whittle's formula, and an efficient enumeration, then follow by application of the classical results on this Markov chain. When T is FSM, the pseudo-state sequence is the same as the state sequence, and our result reduces to the classical case. The proof of Theorem 1 is given in Section III. Later on, in Section IV, we show that the result is readily generalized from close-ended to regular type classes.

It follows from the main result in Section III that the asymptotic behavior of the size of the type class of x^n with respect to T is governed by the first order empirical entropy of the corresponding

pseudo-state sequence, rather than the empirical entropy of x^n itself with respect to T. For the sequence x^n of Example 1, the empirical entropy of the pseudo-state sequence will be zero, consistent with our finding that the type class consists of a single sequence, independently of n. In Section V, we study this asymptotic behavior in more detail, and express the second order term of the logarithm of the type class size in terms of the so-called *minimal canonical extension* (*MCE*) of a tree T. This extension of T, denoted $T_{\rm c}$, is (in general, properly) included in the FSM closure of T. An asymptotic result that holds *in expectation*, also relying on the MCE, was derived in [17].

Finally, in Section VI, we study the number of type classes induced by a tree T on sequences of length n. We present an estimate, also based on the MCE of T, showing that the number of type classes is proportional to $n^{|E_{\rm c}|-|S_{\rm c}|}$, where $S_{\rm c}$ is the set of states of $T_{\rm c}$, and $E_{\rm c}$ is the set of pairs $(u,v) \in S_{\rm c}^2$ such that some sequence over $\mathcal{A}$ causes a direct transition from u to v in $T_{\rm c}$. This estimate is tight up to a multiplicative constant. Once again, when T is FSM, we have $T = T_{\rm c}$, $|E_{\rm c}|-|S_{\rm c}| = (\alpha-1)|S_T|$, and the result reduces to a known result for FSMs (a lower bound is shown in [7], attributed to N. Alon, while an upper bound follows from a linear space dimensionality argument showing the existence of a set of $(\alpha-1)|S_{\mathcal{F}}|$ counters that suffice to determine a type class for an FSM with a set of states $S_{\mathcal{F}}$).

II. Tree type classes and close-ended type classes

A. Notation, background, and preliminaries

We denote by $\mathbb{Z}$, $\mathbb{N}$, and $\mathbb{N}_+$ the integers, nonnegative integers, and positive integers, respectively.

All strings (or sequences) in this paper are over a finite alphabet, $\mathcal{A}$, of size $\alpha \geq 2$. As usual, λ denotes the null string (the null element for the concatenation operation), and $\mathcal{A}^*$, $\mathcal{A}^+$, and $\mathcal{A}^m$, denote, respectively, the set of finite strings, positive-length strings, and strings of length m over $\mathcal{A}$. We use the notation $|z|$ for set cardinality or string length. For a string z, z_i denotes the i-th symbol of z, $1 \leq i \leq |z|$, and z_j^k denotes the substring $z_j z_{j+1} \ldots z_k \in \mathcal{A}^{k-j+1}$, $1 \leq j,k \leq |u|$. We omit the subscript when $j=1$, and we let $z_j^k = \lambda$ whenever $j > k$. For $z = z^k$, $\overleftarrow{z} = z_k z_{k-1} \ldots z_1$ denotes the reverse string of z, $\mathsf{head}(z) = z_1$ (or λ if $k=0$), and $\mathsf{tail}(z) = z_2^k$. We sometimes write sequences of symbols in reverse order, e.g., $z_j z_{j-1} \ldots z_k$. In such cases, we interpret the sequence as the null string λ whenever $j < k$. Concatenation of z and y is denoted zy, and $z \preceq y$ (resp. $z \prec y$) denotes the prefix (resp. proper prefix) relation.

A (*context*) *tree* T is a directed full α-ary tree, where each of the α edges departing from each internal node is labeled with a different symbol from $\mathcal{A}$, and each node is labeled with the string formed by concatenating the edge labels on the path from the root (labeled by λ) to the node. We identify a node

with its label, and a tree with its set of nodes, e.g., $u \in T$ indicates that there is a node of T labeled u. When $T \subseteq T'$ we say that the tree T' is an *extension* of T. A leaf of T is called a *state*, and we denote the set of states by S_T. If $s \in S_T$ and $u \in \mathcal{A}^+$, we say that su *extends* s. We denote the set of internal nodes of T by $\mathcal{I}(T)$, and the depth of T by $\mathsf{depth}(T) = \max\{|u| : u \in T\}$. For a sufficiently long string x, a *state selection function*, denoted $\sigma_T(x)$, assigns to x the (unique) prefix of $\overleftarrow{x}$ in S_T. We refer to $\sigma_T(x)$ as the state *selected* by x.

In the sequel, except when explicitly stated otherwise, we consider a fixed tree T, and we often omit the dependence on T to lighten notation. We will also assume that T is nontrivial, i.e., $|T| > 1$. When $|T| = 1$, the characterization of type classes reduces to a memoryless setting, which is well understood [1]. For the purpose of selecting states, we regard a sequence x^n as preceded by a fixed string x_{-d}^0 that selects an *initial state*, $s_0 = x_0 x_{-1} \dots x_{-d}$, of length $d + 1 = \mathsf{depth}(T)$. Thus, x^n determines a *state sequence*, denoted $\mathbf{s}(x^n)$, defined as $\mathbf{s}(x^n) = s_0(x^n), s_1(x^n), \dots, s_n(x^n)$, where $s_i(x^n) = \sigma(x_{-d}^i)$, which is well defined for all $0 \le i \le n$.[6] We omit the argument x^n of s_i and of other objects of the form $f(x^n)$ when clear from the context. We say that x_{i+1} *is emitted in state* s_i, $1 \le i < n$, and we refer to s_n as the *final state* of x^n. More generally, we say that x_{i+1} is emitted *in context* u in x^n if $\overleftarrow{u}$ is a suffix of x_{-d}^i. For $s \in S_T$ and $a \in \mathcal{A}$, we denote the number of times a symbol a is emitted in state s as $n_s^{(a)}(x^n)$.

The *state transition matrix* of x^n, denoted $N(x^n)$, is an $|S_T| \times |S_T|$ matrix, with rows and columns indexed by S_T, and entries

$$N_{s,t} = |\{i : 1 \le i \le n, s_{i-1} = s, s_i = t\}|, \quad s, t \in S_T,$$

namely, $N_{s,t}$ is the number of transitions from s to t in the state sequence of x^n. For a matrix L we define $L_{i*} = \sum_j L_{i,j}$ and $L_{*j} = \sum_i L_{i,j}$. Thus, N_{s*} is the number of symbols in x^n that are emitted in state s, and N_{*s} is the total number of incoming transitions into s in the state sequence of x^n. Clearly, the number of incoming transitions must equal the number of outgoing transitions for every state s, except possibly for the initial and final states. Therefore, N satisfies the *flow conservation equations*

$$N_{*s}(x^n) + \delta_{s,s_0} = N_{s*}(x^n) + \delta_{s,s_n}, \quad s \in S_T, \tag{2}$$

where $\delta_{u,v} = 1$ when $u = v$, and 0 otherwise. Notice that both the left-hand side and the right-hand side of (2) can be interpreted as the number of positions i in the state sequence of x^n, $0 \le i \le n$, such that $s_i = s$. The following lemma characterizes the support of $N(x^n)$.

[6] This convention simplifies the expression and the derivation of an explicit formula for the type class size, but is not essential. The results presented here can be adapted to other conventions for the selection of the first states, as, for example, the use of transient states in [15].

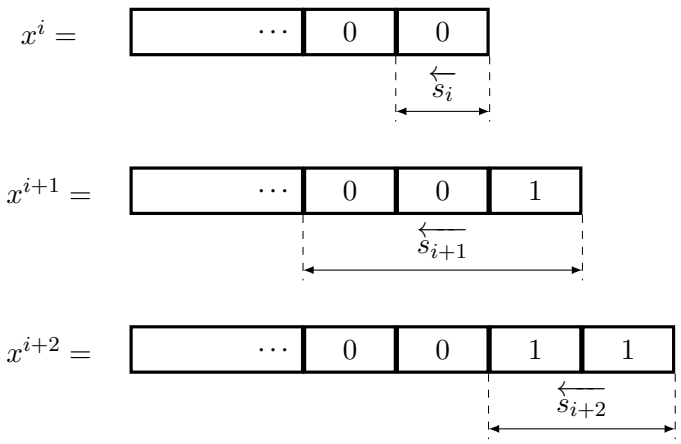

Fig. 2. State transitions in the tree T_1 of Figure 1. Transition $s_i \to s_{i+1}$, with $s_i = 0$ and $s_{i+1} = 100$, satisfies $s_i \preceq \mathsf{tail}(s_{i+1})$. On the other hand, the transition $s_{i+1} \to s_{i+2}$, with $s_{i+2} = 11$, satisfies $\mathsf{tail}(s_{i+2}) \prec s_{i+1}$.

Lemma 1: Let s, t be arbitrary states of T. There exists a string x^n such that $N_{s,t}(x^n) > 0$ if and only if $s \preceq \mathsf{tail}(t)$ or $\mathsf{tail}(t) \prec s$.

Proof: By the definition of state selection in trees, a symbol x_{i+1}, $1 \le i < n$, causes a transition from s to t if and only if $\mathsf{head}(t) = x_{i+1}$ and the reverse of both s and $\mathsf{tail}(t)$ are suffixes of x_{-d}^i, which implies that either $s \prec \mathsf{tail}(t)$, $\mathsf{tail}(t) \prec s$, or $s = \mathsf{tail}(t)$ (see Figure 2). ■

A *model parameter* for a tree T, denoted p_T, is a set of $|S_T|$ *conditional probability mass functions* over $\mathcal{A}$, one associated to each state of T. The tree T and the model parameter p_T define a *tree model*, which we denote by $\langle T, p_T \rangle$. The tree model $\langle T, p_T \rangle$, in turn, defines a *probability assignment* [21] $\mathrm{P}_{\langle T,p_T \rangle}(\cdot)$ given by

$$\mathrm{P}_{\langle T,p_T \rangle}(\lambda) = 1; \quad \mathrm{P}_{\langle T,p_T \rangle}(x^n) = \prod_{i=1}^{n} p_T(x_i | s_{i-1}), \quad n \ge 1\,. \tag{3}$$

For each $n \ge 0$, the assignment (3) defines a probability distribution on $\mathcal{A}^n$.

It follows from (3) that the type class of x^n with respect to T, formally defined as

$$\mathcal{T}_T(x^n) = \left\{ y^n \in \mathcal{A}^n : \mathrm{P}_{\langle T,p_T \rangle}(y^n) = \mathrm{P}_{\langle T,p_T \rangle}(x^n) \text{ for all admissible } p_T \right\},$$

takes the form

$$\mathcal{T}_T(x^n) \;=\; \left\{ y^n \in \mathcal{A}^n : n_s^{(a)}(y^n) \;=\; n_s^{(a)}(x^n)\ \forall s \in S_T, a \in \mathcal{A} \right\}.$$

We refer to $\mathcal{T}_T(x^n)$ also as the T-*class* of x^n, as a shorthand for the type class of x^n with respect to T, and we write simply $\mathcal{T}(x^n)$ when T is clear from the context. To derive some of our main results, it

will be convenient to employ a more restricted notion of type class, which we call the *close-ended type class* of x^n with respect to T (or, in short, the T-*class** of x^n), defined as

$$\mathcal{T}^*(x^n) = \{\, y^n \in \mathcal{T}(x^n) : s_n(x^n) = s_n(y^n) \} \ .$$

For FSM trees, by the flow conservation equations and the existence of a next-state function, equality of the final states is always satisfied for sequences of the same type. Therefore, in the FSM case, T-classes and T-classes* are equivalent. This may not be the case for non-FSM trees.

An *enumeration* of a finite set $\mathcal{S}$ is a one-to-one mapping $f : \mathcal{S} \to \{\, 0, 1, \ldots, |\mathcal{S}|-1 \,\}$. We define an *enumeration scheme for type classes* as an invertible function, $g : \mathcal{A}^* \to \mathbb{N}^{|S_T|\times|\mathcal{A}|+1}$, that assigns to each string x^n the $|S_T| \times |\mathcal{A}|$ counts $n_s^{(a)}$, $s \in S_T, a \in \mathcal{A}$, and the index $f(x^n)$ assigned to x^n by an enumeration f of the set $\mathcal{T}(x^n)$. If for every string x^n, both g and g^{-1} are computable in time polynomial in n, we say that g is an *efficient enumeration scheme for type classes*. Analogous definitions extend to close-ended type classes, where in this case the function g assigns to x^n the counts $n_s^{(a)}$, $s \in S_T, a \in \mathcal{A}$, a description of the final state, s_n, and the index assigned to x^n by an enumeration of $\mathcal{T}^*(x^n)$. We will show that efficient enumeration schemes for type classes are readily derived from efficient enumeration schemes for close-ended type classes.

B. Pseudo-states sequences

In this section we define the set of *pseudo-states* of a tree T, denoted $\tilde{S}_T$, and the *pseudo-state sequence* of a string x^n, denoted $\tilde{\mathbf{s}}(x^n)$. The pseudo-states will provide enough of the context lost by the states to make counting sequences in $\mathcal{T}^*(x^n)$ equivalent to counting pseudo-state sequences with the same transition counts as $\tilde{\mathbf{s}}(x^n)$.

Definition 1: The *extended context sequence* of x^n is $\mathbf{u}(x^n) = u_0, u_1, \ldots, u_n$, where $\overleftarrow{u_i}$, $0 \le i \le n$, is the shortest suffix of x_{-d}^i such that $x_j x_{j-1} \ldots x_{i+1} u_i \notin \mathcal{I}(T)$ for all j, $i \le j \le n$ (in other words, $\overleftarrow{u_i}$ provides enough initial context to determine states for all $j \ge i$).

Notice that, in particular, the case $j = i$ above implies $u_i \notin \mathcal{I}(T)$, $0 \le i \le n$. Also, we have $u_0 = s_0$, $u_n = s_n$, and $u_{i+1} \preceq x_{i+1} u_i$, and we note that only $\mathsf{depth}(T) - |s_i|$ symbols beyond position i need to be checked to determine u_i.

Example 2: Consider the tree T_1 of Figure 1. The sequence $x^n = 001101$, with initial state B, defines the state sequence illustrated in Figure 3(a),

$$\mathbf{s} = \text{B}{\to}\text{A}{\to}\text{A}{\to}\text{B}{\to}\text{D}{\to}\text{A}{\to}\text{C}\,,$$

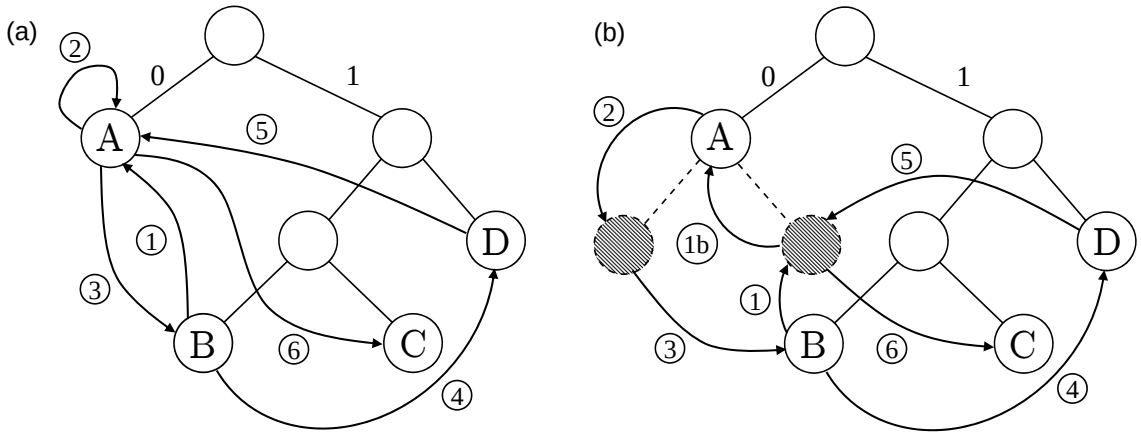

Fig. 3. State sequence (a) and pseudo-state sequence (b) of $x^n = 001101$ with respect to the tree of Figure 1. The pseudo-states 00 and 01 are represented in (b) as dark nodes on an extension of the tree. Transitions are labeled with circled numbers, indicating their order (Transition 1b comes after Transition 1 in (b)).

and we have

$$\mathbf{u} = \mathrm{B}{\to}\mathrm{A}{\to}00{\to}\mathrm{B}{\to}\mathrm{D}{\to}01{\to}\mathrm{C}\,, \tag{4}$$

where, as in the figure, we write $\mathrm{A}, \mathrm{B}, \mathrm{C}, \mathrm{D}$ as shorthands for 0, 100, 101, and 11, respectively. For $i = 1$, we have $u_1 = 0$, since $x_1^1 = 0$ determines $s_1 = \mathrm{A}$, $x_1^2 = 00$ determines $s_2 = \mathrm{A}$, and $x_1^3 = 001$ determines $s_3 = \mathrm{B}$. For $i = 2$, the suffix $x_2^2 = 0$ determines the state $s_2 = \mathrm{A}$, but the suffix $x_2^3 = 01$ does not suffice to determine a state. Thus, u_2 must be longer. The suffix $x_1^2 = 00$ determines $s_2 = \mathrm{A}$, and $x_1^3 = 001$ determines $s_3 = \mathrm{B}$. Thus, we have $u_2 = 00$. Similarly, for $i = 5$, the suffix $x_5^5 = 0$ determines the state $s_5 = \mathrm{A}$, but the suffix $x_5^6 = 01$ does not suffice to determine a state. Instead, the string $x_4^5 = 10$ determines $s_5 = \mathrm{A}$, and $x_4^6 = 101$ determines $s_6 = \mathrm{C}$. Thus, we have $u_5 = 01$.

Definition 2: The set of *pseudo-states* of T is defined as

$$\tilde{S}_T = \{\, s_h^{|s|} : s \in S_T,\, s_k^{|s|} \notin \mathcal{I}(T),\, 1 \le k \le h \,\}. \tag{5}$$

In words, $\tilde{S}_T$ is the set of strings u such that u is a suffix of some state $s \in S_T$ and neither u nor any of the longer suffixes of s is an internal node of T. Since every state is a suffix of itself, and, by definition, not an internal node of T, we have $S_T \subseteq \tilde{S}_T$. Notice that all extended contexts in $\mathbf{u}(x^n)$ belong to $\tilde{S}_T$. Indeed, u_i must be a suffix of s_k for some k, $i \le k \le n$, for otherwise it would not be the shortest suffix of x_{-d}^i that determines a state for all j, $i \le j \le n$. Also, by the definition of u_i, we have $x_j x_{j-1} \dots x_{i+1} u_i \notin \mathcal{I}(T)$ for all j, so u_i satisfies the two conditions for membership in $\tilde{S}_T$.

When T is FSM, all suffixes of a state s are nodes of T [14], [15], implying that $\tilde{S}_T = S_T$ and that the extended context sequence, $\mathbf{u}(x^n)$, is the same as the state sequence, $\mathbf{s}(x^n)$. Indeed, in this case, given $u_i = s_i$, the symbols x_{i+1}^n unambiguously drive state transitions to determine the states $s_{i+1}, s_{i+2}, \ldots, s_n$, by means of the next-state function. In general, however, S_T may be a proper subset of $\tilde{S}_T$, in which case the latter is not prefix-free. Besides the states of T, $\tilde{S}_T$ includes all the states and, possibly, some of the internal nodes of the FSM closure of T.

Example 3: In T_1 of Figure 1, the suffixes $100, 00, 0$ of state B, and the suffixes $101, 01$ of state C, are not internal nodes of T_1. The only suffixes of states A and D that are not internal nodes are A and D themselves. We then have $\tilde{S}_T = S_T \cup \{00, 01\}$.

In analogy to the definition of the state selection function, $\sigma(x)$, we define a *pseudo-state selection function*, denoted $\tilde{\sigma}(x)$, which assigns to every sufficiently long string, x, the longest pseudo-state that is a prefix of $\overleftarrow{x}$. For $u \in \tilde{S}_T \setminus S_T$, we define the *parent* of u, denoted $\rho(u)$, as the longest proper prefix of u in $\tilde{S}_T$.

Now, notice that, since $u_{i+1} \in \tilde{S}_T$ and $u_{i+1} \preceq x_{i+1}u_i$, we have $u_{i+1} \preceq \tilde{\sigma}(\overleftarrow{u_i}x_{i+1})$.

Definition 3: We construct the pseudo-state sequence $\tilde{\mathbf{s}}(x^n)$ from $\mathbf{u}(x^n)$ by inserting, after each u_i such that $u_{i+1} \neq \tilde{\sigma}(\overleftarrow{u_i}x_{i+1})$, a *context-dropping sequence* of pseudo-states, $v_1, v_2, \ldots, v_k$, where $v_1 = \tilde{\sigma}(\overleftarrow{u_i}x_{i+1})$, $v_j = \rho(v_{j-1})$ for all j, $1 < j \leq k$, and $\rho(v_k) = u_{i+1}$. We refer to transitions of the form $u \rightarrow \rho(u)$ generated in this way as *context-dropping transitions*.

Notice that, in contrast to the defining property of the state sequence, $\tilde{\mathbf{s}}(x^n)$ may not be the same as $\{\tilde{\sigma}(x^i)\}_{i=0}^n$, and that $|\tilde{\mathbf{s}}(x^n)|$ will be larger than n if any context-dropping transitions are actually inserted. On the other hand, since each context symbol needed to determine future states in x_{-d}^{n-1} can be dropped by at most one context-dropping transition, the total number of such transitions is at most $n + d$, and we have $|\tilde{\mathbf{s}}(x^n)| \leq 2n + d$. As mentioned, for FSM trees, we have $u_i = s_i$, $u_{i+1} = s_{i+1}$, and $\tilde{S}_T = S_T$, which implies that $\tilde{\sigma}(\overleftarrow{u_i}x_{i+1}) = u_{i+1}$, for all i, yielding $\tilde{\mathbf{s}}(x^n) = \mathbf{u}(x^n) = \mathbf{s}(x^n)$. For general trees, we have $u_{i+1} \preceq \tilde{\sigma}(\overleftarrow{u_i}x_{i+1}) \preceq \tilde{\sigma}(x_{-d}^{i+1})$, where $\tilde{\sigma}(x_{-d}^{i+1})$ coincides with the state selected by x_{-d}^{i+1} in the FSM closure of T, which may be strictly longer than u_{i+1}.

Example 4: For the sequence $x^n = 001101$ of Example 2, we obtain the pseudo-state sequence

$$\tilde{\mathbf{s}} = \text{B}\rightarrow 01 \xrightarrow{\lambda} \text{A}\rightarrow 00\rightarrow \text{B}\rightarrow \text{D}\rightarrow 01\rightarrow \text{C}\,, \tag{6}$$

which is illustrated in Figure 3(b). In (4) we have $u_0 = \text{B} = 100$, $u_1 = \text{A} = 0$, and $x_1 = 0$. Since we have $\tilde{\sigma}(\overleftarrow{u_0}x_1) = 01 \neq u_1$, we insert a sequence of pseudo-states between u_0 and u_1, which in this case, since $\rho(01) = \text{A}$, is comprised of the single pseudo-state 01. This generates the context-dropping

transition labeled with 1b in Figure 3(b) and labeled with λ in (6).

Definition 4: The pseudo-state transition matrix of a sequence x^n, with pseudo-state sequence $\tilde{\mathrm{s}}(x^n) = \tilde{s}_0, \tilde{s}_1, \ldots, \tilde{s}_m$, is a $|\tilde{S}_T| \times |\tilde{S}_T|$ matrix $\tilde{N}(x^n)$, with rows and columns indexed by $\tilde{S}_T$, and entries

$$\tilde{N}_{v,w} = |\{i : 1 \le i \le m, \tilde{s}_{i-1} = v, \tilde{s}_i = w\}|, \quad v, w \in \tilde{S}_T.$$

We denote by $\hat{\mathcal{H}}(\tilde{\mathrm{s}})$ the normalized first-order empirical entropy of $\tilde{\mathrm{s}}$ over $\tilde{S}_T$, namely,

$$\hat{\mathcal{H}}(\tilde{\mathrm{s}}) = -\frac{1}{m} \sum_{v,w \in \tilde{S}_T, \tilde{N}_{v,w} > 0} \tilde{N}_{v,w} \log \frac{\tilde{N}_{v,w}}{\tilde{N}_{v*}} \,. \tag{7}$$

Example 5: The matrix $\tilde{N}(x^n)$ of the sequence $x^n = 001101$ of Example 4, whose pseudo-state sequence is given in (6), with rows and columns indexed by $\tilde{S}_T = \{\mathrm{A}, \mathrm{B}, \mathrm{C}, \mathrm{D}, 00, 01\}$ in this order, is given by

$$\tilde{N}(x^n) = \begin{pmatrix} 0&0&0&0&1&0 \\ 0&0&0&1&0&1 \\ 0&0&0&0&0&0 \\ 0&0&0&0&0&1 \\ 0&1&0&0&0&0 \\ 1&0&1&0&0&0 \end{pmatrix} . \tag{8}$$

C. Exact size and enumeration of close-ended type classes

With the tools developed so far, we are ready to state our main result on size and enumeration of type classes. We provide a formula for the calculation of $|\mathcal{T}^*(x^n)|$ given $\tilde{N}(x^n)$ and state the fact that $\mathcal{T}^*(x^n)$ can be efficiently enumerated. The proof is presented in Section III. Later, in Section IV, we show that these results are readily adapted also to the enumeration and calculation of the size of $\mathcal{T}(x^n)$.

Our construction of $\tilde{N}(x^n)$ so far has been based on the pseudo-state sequence $\tilde{\mathrm{s}}(x^n)$ constructed in Section II-B. As part of the proof in Section III, we show that $\tilde{N}(x^n)$ depends on x^n only through its close-ended type class, by deriving an explicit construction of $\tilde{N}(x^n)$ directly from the counts $n_s^{(a)}(x^n)$ and the final state $s_n(x^n)$.

We define the normalized $|\tilde{S}_T| \times |\tilde{S}_T|$ matrix $\hat{N}$ as $\hat{N}_{v,w} = \tilde{N}_{v,w}/\tilde{N}_{v*}$ if $\tilde{N}_{v*} > 0$ and $\hat{N}_{v,w} = 0$ otherwise.

Theorem 1: The size of a close-ended type class is given by

$$|\mathcal{T}^*(x^n)| = M \frac{\prod_v \tilde{N}_{v*}!}{\prod_{v,w} \tilde{N}_{v,w}!} \,, \tag{9}$$

where M denotes the cofactor of entry (s_n, s_0) in $I - \hat{N}$, which satisfies $M \le 1$. Furthermore, there exists an efficient enumeration scheme for $\mathcal{T}^*(x^n)$.

Example 6: For the sequence $x^n = 001101$ of Example 4, whose matrix $\tilde{N}$ is given in (8), we have

$$I - \hat{N} = \begin{pmatrix} 1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & -\frac{1}{2} & 0 & -\frac{1}{2} \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & -1 \\ 0 & -1 & 0 & 0 & 1 & 0 \\ -\frac{1}{2} & 0 & -\frac{1}{2} & 0 & 0 & 1 \end{pmatrix} . \tag{10}$$

The final state of x^n is $s_n = \mathrm{C}$ and the initial state is $s_0 = \mathrm{B}$. Thus, calculating the cofactor of (10) that corresponds to the third row and the second column, we obtain $M = \frac{1}{2}$. Since the multinomial factor $\frac{\prod_v \tilde{N}_{v*}!}{\prod_{v,w} \tilde{N}_{v,w}!}$ applied to (8) equals 4, (9) yields $|\mathcal{T}^*(x^n)| = 2$, and indeed, by direct counting, we obtain

$$\mathcal{T}^*(x^n) = \{001101, 100101\} .$$

Example 7: For the sequence x^n of Example 1, we obtain the pseudo-state sequence

$$\tilde{\mathbf{s}} = \mathrm{B} \to 01 \xrightarrow{\lambda} \mathrm{A} \to 00 \to \mathrm{B} \to \ldots \to \mathrm{B} \to 01 \xrightarrow{\lambda} \mathrm{A} \to 00 \to \mathrm{B} .$$

For this sequence, we have $\hat{\mathcal{H}}(\tilde{\mathbf{s}}) = 0$, which in this case coincides with $\hat{\mathcal{H}}_{T_F}(x^n)$ (this needs not be the case in general). We also obtain

$$\tilde{N} = \begin{pmatrix} 0 & 0 & 0 & 0 & \frac{n}{4} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{n}{4} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{n}{4} & 0 & 0 & 0 & 0 \\ \frac{n}{4} & 0 & 0 & 0 & 0 & 0 \end{pmatrix} , \quad I - \hat{N} = \begin{pmatrix} 1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} .$$

Thus, both the multinomial factor in (9) and the cofactor that corresponds to the second row and the second column of $I - \hat{N}$ evaluate to 1, in agreement with the observation in Section I that the type class is comprised of a single sequence.

As mentioned, when T is FSM, $\tilde{S}_T = S_T$, and the pseudo-state sequence $\tilde{\mathbf{s}}(x^n)$ coincides with the state sequence $\mathbf{s}(x^n)$. Thus, we have $\tilde{N} = N$, and in this case, (9) reduces to Whittle's formula on the original state sequence. In general, however, counting state sequences compatible with N, i.e., applying Whittle's formula directly to N, may result in overcounting sequences in $\mathcal{T}^*(x^n)$. Unlike in the FSM case, some such state sequences may not correspond to any symbol sequence, as shown in the following example.

Example 8: The sequence $x^n = 001101$ of Examples 2 and 4, defines the state sequence

$$\mathbf{s} = \mathrm{B} \to \mathrm{A} \to \mathrm{A} \to \mathrm{B} \to \mathrm{D} \to \mathrm{A} \to \mathrm{C} ,$$

which, taking A,B,C,D as the order for rows and columns, yields the state transition matrix

$$N(x^n) = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}. \tag{11}$$

The state sequence B→A→B→D→A→A→C is compatible with $N(x^n)$, but it does not correspond to any symbol sequence. If it were the state sequence of a string y, the first state transition, B→A, implies that $y_{-d}^{1} = 0010$ and, therefore, the next state, s_2, would not be B but C (if $y_2 = 1$) or A (if $y_2 = 0$). Transitions from A to B, however, may be valid elsewhere in the sequence. The extra context provided by the elements of the pseudo-state sequence accounts for these restrictions on state transition sequences.

III. Proof of Theorem 1

To prove Theorem 1 we will define a *tagging* function, ω, that maps sequences of pseudo-states to symbol sequences, and will show that ω determines, in fact, a bijection between pseudo-state sequences starting at s_0 and compatible with $\tilde{N}(x^n)$, and sequences in $\mathcal{T}^*(x^n)$. As Example 8 shows, such a one-to-one correspondence does not exist, in general, between state sequences compatible with $N(x^n)$ and sequences in $\mathcal{T}^*(x^n)$, since, for an arbitrary tree, some state sequences may not correspond to any symbol sequence. We first show, in Subsection III-A, that the above counterexample notwithstanding, two sequences x^n, y^n belong to the same close-ended type class if and only if $N(y^n) = N(x^n)$. Using this result, in Subsection III-B, we describe an explicit construction of the matrix $\tilde{N}(x^n)$ given the close-ended type $\mathcal{T}^*(x^n)$ (but not necessarily the sequence x^n itself). In Subsection III-C, we define the tagging function ω, and use the fact that $\tilde{N}(x^n)$ depends on x^n only through $\mathcal{T}^*(x^n)$ to establish that any sequence $y^n \in \mathcal{T}^*(x^n)$ can be obtained by tagging a pseudo-state sequence starting at s_0 and compatible with $\tilde{N}(x^n)$, namely, the pseudo-state sequence of y^n. Moreover, we prove that the tagging of any two different such sequences of pseudo-states yields different symbol sequences, establishing, therefore, the fact that ω is bijective. The proof of Theorem 1 will then follow straightforwardly, by application of Whittle's formula to $\tilde{N}(x^n)$.

A. State transition counts and close-ended type classes

The following lemma connects sequences in $\mathcal{T}^*(x^n)$ and state transition counts, as collected in the state transition matrix $N(x^n)$.

Lemma 2: For sequences $x^n, y^n \in \mathcal{A}^n$, and a tree T, we have $y^n \in \mathcal{T}^*(x^n)$ if and only if $N(y^n) = N(x^n)$.

Proof: Clearly, since a state transition $s_{i-1} \to s_i$ uniquely determines the symbol $x_i = \mathsf{head}(s_i)$ that is emitted in state s_{i-1}, the equality $N(y^n) = N(x^n)$ implies that $y^n \in \mathcal{T}(x^n)$. Moreover, by the flow conservation equations (2), the final state of a sequence is uniquely determined by the initial state s_0 and the state transition matrix. Thus, if $N(y^n) = N(x^n)$, then we must have $s_n(y^n) = s_n(x^n)$ and therefore $y^n \in \mathcal{T}^*(x^n)$.

To prove the "only if" part, we recall that, by Lemma 1, a state transition $s \to t$ is possible if and only if either $s \preceq \mathsf{tail}(t)$ or $\mathsf{tail}(t) \prec s$. In the former case, all the transitions into state t originate from s, while in the latter case, all the transitions from s to t are driven by an emission of the symbol $\mathsf{head}(t)$ in s. Thus, from the flow conservation equations (2), we have

$$N_{s,t}(x^n) = \begin{cases} \sum_{b\in\mathcal{A}} n_t^{(b)}(x^n) + \delta_{t,s_n(x^n)} - \delta_{t,s_0}, & \text{if } s \preceq \mathsf{tail}(t)\,, \\ n_s^{(a)}(x^n), \ \text{where } a = \mathsf{head}(t), & \text{if } \mathsf{tail}(t) \prec s\,, \end{cases} \tag{12}$$

which implies that $N(x^n)$ depends on x^n only through its close-ended type $\mathcal{T}^*(x^n)$. ■

B. General construction of $\tilde{N}(x^n)$

To calculate $\tilde{N}(x^n)$ from $\mathcal{T}^*(x^n)$ we first establish necessary conditions for entries of $\tilde{N}(x^n)$ to be positive. These conditions are analogous to (albeit, at this point, weaker than) the conditions established in Lemma 1 for entries of $N(x^n)$.

Lemma 3: If $\tilde{N}_{v,w}(x^n) > 0$ for some sequence x^n, then the pseudo-states v, w must satisfy one of the following conditions:

$$v \in \tilde{S}_T \setminus S_T\,, \quad v = \mathsf{tail}(w)\,, \tag{13}$$

$$v \in S_T\,, \quad w = \tilde{\sigma}(\overleftarrow{v}\,\mathsf{head}(w))\,, \tag{14}$$

$$w = \rho(v)\,, \quad \mathsf{tail}(v) \in T\,. \tag{15}$$

Proof: We show that any two consecutive pseudo-states in $\tilde{\mathbf{s}}(x^n)$ satisfy one of (13)–(15). Consider consecutive pseudo-states u_i and u_{i+1} in the extended context sequence, $\mathbf{u}(x^n)$, defined in Section II-B (Definition 1). We have $u_{i+1} \preceq x_{i+1}u_i$. If $u_{i+1} = x_{i+1}u_i$, since no context-dropping transitions are inserted between u_i and u_{i+1} when constructing $\tilde{\mathbf{s}}(x^n)$ in this case, u_i and u_{i+1} are consecutive in $\tilde{\mathbf{s}}(x^n)$ and satisfy (13) or (14). Otherwise, if $u_{i+1} \prec x_{i+1}u_i$, by the definition of the extended context sequence, s_i must depend on the last symbol of u_i, implying $u_i = s_i$. Thus, the pseudo-state following u_i in $\tilde{\mathbf{s}}(x^n)$ will be $\tilde{\sigma}(\overleftarrow{u_i}x_{i+1})$, as in (14), either because it coincides with u_{i+1} or because it is inserted as the first pseudo-state in a context-dropping sequence $v_1, v_2, \ldots, v_k$. If such an insertion takes place, since

$v_1 = \tilde{\sigma}(\overleftarrow{u_i} x_{i+1}) \preceq x_{i+1}u_i = x_{i+1}s_i$, then, for all $h = 1 \dots k$, $\mathsf{tail}(v_h) \preceq s_i$ (thus, it belongs to T), so that the context-dropping transitions are of the form (15). ■

The conditions (13)–(15) are, in general, loose, in the sense that there may exist pairs of pseudo-states (v, w) that satisfy one of (13)–(15), but such that $\tilde{N}_{v,w}(x^n) = 0$ for all sequences x^n. This gap does not affect our ongoing derivations, and will be closed in Section V, when we focus on a subset of context trees for which (13)–(15) are both necessary and sufficient.

In the proof of the following lemma we show, explicitly, how $\tilde{N}(x^n)$ can be calculated from $\mathcal{T}^*(x^n)$.

Lemma 4: For every sequence x^n, $\tilde{N}(x^n)$ depends on x^n only through its close-ended type $\mathcal{T}^*(x^n)$.

Proof: By Lemma 2, it suffices to show that $\tilde{N}(x^n)$ depends on x^n only through $N(x^n)$.

Given $w \in \tilde{S}_T$ such that $\mathsf{tail}(w) \in \tilde{S}_T \setminus S_T$, consider the set S_w of states s such that w is a suffix of s, none of the longer suffixes of s are in $\mathcal{I}(T)$, and s does not contain as proper suffix a state with those properties (by the definition of $\tilde{S}_T$ the set S_w is nonempty). Now, notice that by the definition of the extended context sequence $\mathbf{u}(x^n)$, for a pseudo-state $u_i \notin S_T$ we have $u_{i+1} = x_{i+1}u_i$. Therefore, there exists $k \geq i$ such that $s_k \in S_{u_i}$, implying that every transition of the form (13) occurs in conjunction with a state in S_w. Moreover, the exclusion from S_w of states which have a suffix in S_w guarantees that each such transition is counted only once. Thus,

$$\tilde{N}_{\mathsf{tail}(w),w} = \sum_{s \in S_w} N_{*s} - \sum_{wz \in \tilde{S}_T, z \in \mathcal{A}^+} \tilde{N}_{\mathsf{tail}(wz),wz}\,, \tag{16}$$

where the summation subtracted in (16) accounts for states s' having a state $s \in S_w$ as a proper prefix of one (or more) of its suffixes. When such a state s' occurs in a state sequence, say $s_j = s'$, we must have $s_i = s$ for some $i < j$, and this occurrence of s will not be associated to the occurrence of the pseudo-state w but to a longer pseudo-state wz (since a longer context is required to determine $s_j = s'$). Notice that (16) defines a recursion, since wz is strictly longer than w, and that N_{*s} can be derived from symbol counts by (12).

For v, w of the form (14), with $b = \mathsf{head}(w)$, we have

$$\tilde{N}_{v,w} = n_v^{(b)} - \sum_{bvz \in \tilde{S}_T, z \in \mathcal{A}^+} \tilde{N}_{vz,bvz}\,. \tag{17}$$

This formula gives the number of positions in $\mathbf{u}(x^n)$ in which the extended context u_i is $s_i = v$ and the next symbol is $x_{i+1} = b$. In all these cases, the next pseudo-state in $\tilde{\mathbf{s}}(x^n)$ will be $\tilde{\sigma}(\overleftarrow{v}\, b)$, either because it coincides with u_{i+1} or because it is the starting point for a context-dropping sequence inserted between u_i and u_{i+1}, as defined in Section II-B (Definition 3). Notice that all terms in the summation of (17) can be calculated from (16).

Finally, a transition of the form (15), i.e., $v \to \rho(v)$, occurs as a result of the insertion of a context-dropping sequence for each index i such that $v \preceq \tilde{\sigma}(\overleftarrow{u_i} x_{i+1})$ and $u_{i+1} \prec v$ (and only in those cases). The first condition occurs if and only if there is a transition from v' to w' of the form (13)-(14) with $v \preceq w'$, whereas the second condition occurs if and only if there is a transition of the same form with $v' \prec v$. It then follows that

$$\tilde{N}_{v,\rho(v)} = \sum_{\substack{v',w' \in \tilde{S}_T, \\ v \preceq w', w' \not\preceq v'}} \tilde{N}_{v',w'} - \sum_{\substack{v',w' \in \tilde{S}_T, \\ v' \prec v, w' \not\preceq v'}} \tilde{N}_{v',w'} \,. \qquad \blacksquare$$

C. *Pseudo-state transition counts and close-ended type classes*

In this subsection we establish a connection between symbol sequences and sequences of pseudo-states, by means of a tagging function ω. Specifically, given a sequence of pseudo-states, $\tilde{\mathbf{s}} = v_0, v_1, \ldots, v_m$, we tag each transition $v_i \to v_{i+1}$ with a string

$$\omega(v_i, v_{i+1}) = \begin{cases} \lambda, & v_{i+1} \prec v_i, \\ \mathsf{head}(v_{i+1}), & \text{otherwise}\,. \end{cases} \tag{18}$$

We then define $\omega(\tilde{\mathbf{s}})$ as the string obtained by concatenating the transition tags, in the order in which they appear. Notice that for the pseudo-state sequence of x^n, since $\mathsf{head}(u_{i+1}) = x_{i+1}$, we have $\omega(\tilde{\mathbf{s}}(x^n)) = x^n$. For completeness, we define $\omega(\tilde{\mathbf{s}}) = \lambda$ when $\tilde{\mathbf{s}}$ is comprised of a single pseudo-state v_0.

Lemma 5: Let $\tilde{\mathbf{s}} = v_0, v_1, \ldots, v_m$ and $\tilde{\mathbf{s}}' = v'_0, v'_1, \ldots, v'_{m'}$ be sequences of pseudo-states such that all pseudo-state transitions $v \to w$ satisfy one of (13)–(15). Then,

(i) $\overleftarrow{v_m}$ is a suffix of $\overleftarrow{v_0}\omega(\tilde{\mathbf{s}})$, and

(ii) if $v_0 = v'_0$, $v_m = v_{m'} = s \in S_T$, and $\omega(\tilde{\mathbf{s}}) = \omega(\tilde{\mathbf{s}}')$, then $\tilde{\mathbf{s}} = \tilde{\mathbf{s}}'$.

Proof: We prove (i) by induction on m. For $m = 0$, the claim is trivial. Assume it is true for $m-1$ and consider the last transition, $v_{m-1} \to v_m$. If $v_m = \rho(v_{m-1})$ (as in (15)), then the transition is tagged with λ. Thus, $\overleftarrow{v_m}$, which is a suffix of $\overleftarrow{v_{m-1}}$, is also a suffix of $\overleftarrow{v_0}\omega(\tilde{\mathbf{s}})$. If the transition is of type (13) or of type (14), then it is tagged with $\mathsf{head}(v_m)$ and $\mathsf{tail}(v_m) \preceq v_{m-1}$. Hence, by the induction hypothesis, $\overleftarrow{v_m}$ is a suffix of $\overleftarrow{v_0}\omega(\tilde{\mathbf{s}})$, which proves (i).

To prove (ii), let j be the largest index such that $v_i = v'_i$ for all $i \le j$ ($j \ge 0$). If $j = \min\{m, m'\}$ then, either $m = m'$, in which case the lemma is proved, or all remaining transitions in the longest sequence must be tagged with λ, i.e., must be of the form (15), contradicting the fact that $v_m = v'_m$. Thus, we can assume $j < \min\{m, m'\}$. Let $b_1^k = \omega(v_j, v_{j+1}, \ldots, v_m)$. By the definition of j, and since $\omega(\tilde{\mathbf{s}}) = \omega(\tilde{\mathbf{s}}')$, we must also have $b_1^k = \omega(v'_j, v'_{j+1}, \ldots, v'_m)$.

Letting $v_j = v'_j = v$, one of v_{j+1} and v'_{j+1} must be $\rho(v)$, for otherwise they would coincide, because the next pseudo-state in a transition of any the forms (13)-(14) must be $\tilde{\sigma}(\overleftarrow{v}\, b_1)$. Hence, $v \notin S_T$, which discards a transition of the form (14) from v, and we must have, with no loss of generality, $v_{j+1} = b_1 v$ and $v'_{j+1} = \rho(v)$.

Notice that, by an application of (i) to the subsequence of $\tilde{\mathbf{s}}'$ starting at pseudo-state v'_{j+1}, the remaining pseudo-states in $\tilde{\mathbf{s}}'$ are all prefixes of $b_i b_{i-1} \dots b_1 \rho(v)$, $1 \leq i \leq k$. As a result, $b_i b_{i-1} \dots b_1 v \notin S_T$ for all i, $0 \leq i \leq k$. We further claim that $v_{j+i} = b_i b_{i-1} \dots b_1 v$, $0 \leq i \leq m-j$, implying that $v_{j+i} \notin S_T$, in contradiction with $v_m = s$ (note also that this claim implies $k = m-j$). We prove the claim by induction on i. For $i = 1$, we already established that $v_{j+1} = b_1 v$ and $v_j = v$. Assume now that the claim holds for $i-1$ and $i-2$. We have $v_{j+i-1} = b_{i-1} b_{i-2} \dots b_1 v \notin S_T$, excluding a transition $v_{j+i-1} \to v_{j+i}$ of type (14). Moreover, $\mathsf{tail}(v_{j+i-1}) = v_{j+i-2} \notin T$, excluding also a transition of type (15). Therefore, the only possible transition is of type (13), implying $v_{j+i} = b_i b_{i-1} \dots b_1 v$. ■

Lemma 5(ii) states that two different pseudo-state sequences that start and end at the same pseudo-state yield two different symbol sequences through the tagging function ω. Lemma 6 below makes use of this fact to show that, in fact, the sequences in $\mathcal{T}^*(x^n)$ and the sequences of pseudo-states starting at s_0 and compatible with $\tilde{N}(x^n)$ are in one to one correspondence.

Lemma 6: Let x^n be a sequence in $\mathcal{A}^n$. The function ω defines a bijection between $\mathcal{T}^*(x^n)$ and the set of pseudo-state sequences starting at s_0 and compatible with $\tilde{N}(x^n)$.

Proof: By the definition of ω and of the pseudo-state sequence of a string, we have $\omega\left(\tilde{\mathbf{s}}(y^n)\right) = y^n$ for every sequence y^n and, if $y^n \in \mathcal{T}^*(x^n)$, by Lemma 4, we have $\tilde{N}(y^n) = \tilde{N}(x^n)$. Thus, every sequence in $\mathcal{T}^*(x^n)$ is the image, under the tagging function, of a sequence of pseudo-states that starts at s_0 and is compatible with $\tilde{N}(x^n)$.

Now, let $\tilde{\mathbf{v}} = v_0, v_1, \dots, v_m$ be an arbitrary sequence of pseudo-states, with $v_0 = s_0$, compatible with $\tilde{N}(x^n)$. Let also $\tilde{\mathbf{s}}(x^n) = \tilde{s}_0, \tilde{s}_1, \dots, \tilde{s}_m$. Since $\tilde{\mathbf{s}}(x^n)$ and $\tilde{\mathbf{v}}$ share the same pseudo-state transition counts, there exists a permutation $i \mapsto i'$ of the indexes i, $0 \leq i < m$, so that $v_i = \tilde{s}_{i'}$ and $v_{i+1} = \tilde{s}_{i'+1}$, for all i, $0 \leq i < m$. Thus, by Lemma 5(i), $\omega\left(v_0, v_1, \dots, v_i\right)$ selects the same state as $\omega\left(\tilde{s}_0, \tilde{s}_1, \dots, \tilde{s}_{i'}\right)$, and $\omega\left(v_0, v_1, \dots, v_{i+1}\right)$ selects the same state as $\omega\left(\tilde{s}_0, \tilde{s}_1, \dots, \tilde{s}_{i'+1}\right)$, for all i, $0 \leq i < m$. As a consequence, $\omega\left(\tilde{\mathbf{s}}(x^n)\right)$ and $\omega\left(\tilde{\mathbf{v}}\right)$ share the same state transition matrix, implying, by Lemma 2, that $\omega\left(\tilde{\mathbf{v}}\right)$ belongs to $\mathcal{T}^*(x^n)$. Moreover, if $\tilde{\mathbf{v}}' = v'_0, v'_1, \dots, v'_m$ is a different sequence of pseudo-states, with $v'_0 = s_0$, that is also compatible with $\tilde{N}(x^n)$, then, by the flow conservation equations (2) on $\tilde{N}(x^n)$, we must have $v_m = v'_m$, implying, by Lemma 5(ii), that $\omega\left(\tilde{\mathbf{v}}'\right) \neq \omega\left(\tilde{\mathbf{v}}\right)$. ■

Example 9: The pseudo-state sequence $\tilde{\mathbf{s}} = \text{B}\to 01 \xrightarrow{\lambda} \text{A}\to 00\to\text{B}\to\text{D}\to 01\to\text{C}$, from Example 4, is illus-

trated in Figure 3(b), following the transitions in order $1, 1\text{b}, 2, 3, 4, 5, 6$. Notice that in $\tilde{\mathbf{s}}$ the state B is preceded by 00, which in turn is preceded by A. This pseudo-state sequence, A→00→B, is associated to the emission of the string 01 starting from A, which causes the state sequence A→A→B. Thus, no permutation of the pseudo-state transitions of $\tilde{\mathbf{s}}$ generates the invalid sequence B→A→B mentioned in Example 8. Other permutations of pseudo-state transitions, however, give valid strings in the type class of x^n. The tagging of the sequence of pseudo-states obtained by following the transitions in the order $4, 5, 1\text{b}, 2, 3, 1, 6$ in Figure 3(b) gives the string $y^n = 100101$, which, as observed in Example 6, belongs to $\mathcal{T}^*(x^n)$. In fact, x^n and y^n are the only two strings in $\mathcal{T}^*(x^n)$.

Proof of Theorem 1: Whittle's formula (1) applied to $\tilde{N}(x^n)$ gives the number of sequences of pseudo-states starting at s_0 and compatible with $\tilde{N}(x^n)$, which, by Lemma 6, are in bijective correspondence with the elements of $\mathcal{T}^*(x^n)$, yielding (9). Moreover, since the multinomial factor in (9) is a trivial upper bound on the number of such sequences of pseudo-states, we must have $M \leq 1$ as claimed. Now, given x^n, one can compute $\tilde{\mathbf{s}}(x^n)$ in linear time (regarding T as fixed), as in Section II-B. Computing $\omega(\tilde{\mathbf{s}})$ given the pseudo-state sequence $\tilde{\mathbf{s}}$ of a string y^n can be done in time proportional to the length of $\tilde{\mathbf{s}}$, which we recall, from Section II-B, that is upper-bounded by $2n + d$. Hence, by Lemma 6, enumerating sequences in $\mathcal{T}^*(x^n)$ is equivalent to enumerating sequences of pseudo-states compatible with $\tilde{N}(x^n)$. The latter, in turn, can be done polynomially in n by recursive application of Whittle's formula, similarly to the enumeration of Markov type classes in [22]. ■

IV. Connections between close-ended type classes and type classes

We study connections between close-ended type classes and type classes, which show that the problem of enumerating sequences in $\mathcal{T}^*(x^n)$ is essentially equivalent to that of enumerating sequences in $\mathcal{T}(x^n)$. These connections also provide the means to apply (9) to calculate $|\mathcal{T}(x^n)|$.

We start by stating the following relationship between the final states of any two sequences in the same T-class.

Lemma 7: Let $y^n \in \mathcal{T}(x^n)$. If $s_{n-1}(x^n)$ and x_n determine $s_n(x^n)$, then we have $s_n(x^n) = s_n(y^n)$. Otherwise, we must have $s_{n-1}(x^n) = s_{n-1}(y^n)$, and $x_n = y_n$.

Notice that when T is FSM, in which case $s_{n-1}(x^n)$ and x_n always determine $s_n(x^n)$, Lemma 7 states that we must have $s_n(x^n) = s_n(y^n)$ whenever $y^n \in \mathcal{T}(x^n)$. Indeed, this fact is a trivial consequence of the flow conservation equations (2) and the existence of a next-state function. The proof of Lemma 7 for general trees is deferred to Appendix A. It follows from standard arguments based on the definition of the state selected by a sequence, and the flow conservation equations. The following theorem connects

$\mathcal{T}^*(x^n)$ with $\mathcal{T}(x^n)$, enabling the application of (9) to the calculation of $|\mathcal{T}(x^n)|$, and, moreover, it states that efficient enumeration schemes for type classes are readily derived from efficient enumeration schemes for close-ended type classes.

Theorem 2: (i) For a sequence x^n, we have $\mathcal{T}(x^n) = \mathcal{T}^*(x^n)$ if $s_{n-1}(x^n)$ and x_n determine $s_n(x^n)$, or $\mathcal{T}(x^n) = \{y^{n-1}x_n : y^{n-1} \in \mathcal{T}^*(x^{n-1})\}$ otherwise. Thus, we have $|\mathcal{T}(x^n)| = |\mathcal{T}^*(x^r)|$, with either $r = n$ or $r = n-1$.

(ii) Given an efficient enumeration scheme for close-ended type classes, g, we can construct an efficient enumeration scheme for type classes, g', such that the computations of g' and its inverse for sequences of length n take $O(n)$ additional operations with respect to the computations of g and g^{-1}, respectively.[7]

Proof: (i) If $s_{n-1}(x^n)$ and x_n determine $s_n(x^n)$, then, by Lemma 7 and the definition of $\mathcal{T}^*(x^n)$, we have $\mathcal{T}(x^n) = \mathcal{T}^*(x^n)$. Suppose now that $s_{n-1}(x^n)$ and x_n do not determine $s_n(x^n)$, and consider the subsequence x^{n-1}. For all states s and all symbols a, we have

$$n_s^{(a)}(x^{n-1}) = n_s^{(a)}(x^n) - \delta_{s,s_{n-1}(x^n)}\delta_{a,x_n}\,.$$

Similarly, for an arbitrary string $z^n \in \mathcal{T}(x^n)$, we have

$$n_s^{(a)}(z^{n-1}) = n_s^{(a)}(z^n) - \delta_{s,s_{n-1}(z^n)}\delta_{a,z_n}\,.$$

Since, by Lemma 7, we must have $s_{n-1}(z^n) = s_{n-1}(x^n)$ and $z_n = x_n$, we conclude that $z^{n-1} \in \mathcal{T}^*(x^{n-1})$. Thus, the string z^n belongs to the set $\{y^{n-1}x_n : y^{n-1} \in \mathcal{T}^*(x^{n-1})\}$. Conversely, if we append the symbol x_n to the end of a string $y^{n-1} \in \mathcal{T}^*(x^{n-1})$, we get

$$n_s^{(a)}(y^{n-1}x_n) = n_s^{(a)}(y^{n-1}) + \delta_{s,s_{n-1}(y^{n-1})}\delta_{a,x_n}, \quad \text{for all } s \in S_T, a \in \mathcal{A}\,.$$

Since $n_s^{(a)}(y^{n-1}) = n_s^{(a)}(x^{n-1})$ and $s_{n-1}(y^{n-1}) = s_{n-1}(x^{n-1})$ by the definition of $\mathcal{T}^*(x^{n-1})$, we conclude that $n_s^{(a)}(y^{n-1}x_n) = n_s^{(a)}(x^n)$ for all states s and all symbols a, and, therefore, $y^{n-1}x_n \in \mathcal{T}(x^n)$.

(ii) The result is straightforward for the computation of $g'(x^n)$, which amounts to computing the counts $n_s^{(a)}$ and, by (i), the index assigned by g to either x^n or x^{n-1}. We focus on the inverse computation, i.e., given the counts $n_s^{(a)}$ and the index assigned by g' to x^n, reconstruct x^n. For this case, the claim follows if we can efficiently perform the following computational tasks: decide whether s_{n-1} and x_n determine s_n (in which case we just need to apply g^{-1}) or not, and in the latter case, find the values of s_{n-1} and x_n (from which the counts $n_s^{(a)}(x^{n-1})$, needed to apply g^{-1}, are obtained readily from the

[7] Asymptotics are with respect to n, with $\mathcal{A}$ and T given.

counts $n_s^{(a)}(x^n)$). To this end, let $S'_T = \{s \in S_T \ : \ s = aw, a \in \mathcal{A}, w \in T\}$. Clearly, s_{n-1} and x_n determine s_n if and only if $s_n \in S'_T$. For states $s = aw \in S'_T$, we can compute the number of incoming transitions into s as $\sum_{v:wv \in S_T} n_{wv}^{(a)}$, and we can determine if s is the final state by checking whether $\sum_{v:wv \in S_T} n_{wv}^{(a)} + \delta_{s,s_0} > \sum_{b \in \mathcal{A}} n_s^{(b)}$. The test can be done for all $s \in S'_T$ and if s_n is not found in S'_T, then s_{n-1} and x_n do not determine s_n. In this case, we must have $s_n = auv$, with $a \in \mathcal{A}$, $u \in S_T$, and $v \in \mathcal{A}^+$. For each internal node au of T with $u \in S_T$, we can determine if the final state is of the form auv by comparing $n_u^{(a)} + \sum_{v:auv \in S_T} \delta_{auv,s_0}$ with $\sum_{v:auv \in S_T} \sum_{b \in \mathcal{A}} n_{auv}^{(b)}$. The two quantities will differ for one and only one such node au, yielding $x_n = a$, and $s_{n-1} = u$. ■

V. Canonical extensions and asymptotic behavior

Using Stirling's approximation, it follows from Theorem 1 that the size of $\mathcal{T}_T(x^n)$ is exponential in the first-order empirical entropy of $\tilde{\mathbf{s}}(x^n)$ over $\tilde{S}_T$. In this section, we study this asymptotic behavior in more detail, and show that its second-order term is related to a special extension of the tree T, which may generally be different from either T or its FSM closure, and which will also be crucial for the results of Section VI. We start by defining this extension and studying its properties.

Definition 5: We say that a state s of T is *forgetful* if $as \in \mathcal{I}(T)$ for *all* $a \in \mathcal{A}$, namely, s does not have sufficient context for *any* transition. A tree with no forgetful states is called *canonical*.

For a tree to be canonical we require that for every state s, there exist *at least one symbol* a, such that the emission of a in state s unambiguously determines the next state. On the other hand, in an FSM tree, for every state s and *every* symbol a, the emission of a in state s unambiguously determines the next state. Therefore, FSM trees are, a fortiori, canonical.

Example 10: In the binary tree T of Figure 4, the next state following the emission of symbol 0 from $s = 01$ can be any of $00100, 00101, 0011$, and the next state can be any of $1010, 1011$ after the emission of symbol 1. Since neither 0 nor 1 determine the next state from s, s is forgetful and, thus, T is not canonical.

A *single extension* of a state s of T consists of extending T with a full complement of children of s. Consider the following procedure applied to a tree T that is not canonical: select a forgetful state s, and singly extend it; continue until no forgetful states remain. Since states of length $\mathsf{depth}(T) - 1$ or above are never forgetful, the procedure must stop after a finite number of steps, resulting in a canonical tree. The next lemma shows that this extension is unique and independent of the order in which forgetful states are selected for extension. We call it the *minimal canonical extension* (MCE) of T, and denote it by T_{c}.

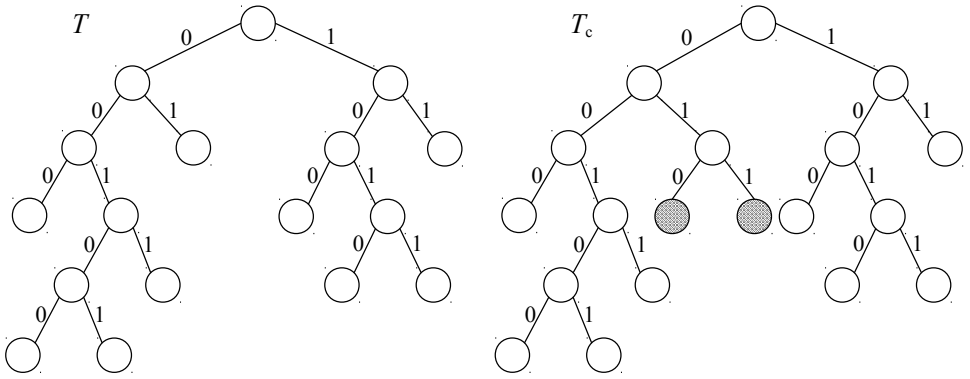

Fig. 4. A non-canonical tree T over $\mathcal{A} = \{0, 1\}$ and its MCE T_c. Nodes added in T_c with respect to T are highlighted.

Lemma 8: Let T' and T'' be two extensions of T, each obtained by a finite sequence of single extension steps on forgetful states. If T' and T'' are both canonical, then $T' = T''$.

Proof: Suppose $T' \neq T''$ and, without loss of generality, suppose $T' \backslash T''$ is not empty. Let $v_1, v_2, \ldots, v_m$ be the sequence of forgetful states extended in the process of constructing T' from T. There must be some node in this sequence that was not extended in T'' (otherwise we would have $T' \subseteq T''$). Let v_j be the first such node in the sequence. We must have $v_j \in T''$, either because v_j was originally in T, or because it was created by extension of some v_i with $i < j$, which was also extended in T''. Moreover, v_j must be a leaf of T'', since it was not extended in that tree, and it could not have been an internal node of T since it was selected for extension in T'. Let T'_j be the tree in the sequence of extensions leading to T' at the time v_j was chosen for extension. Clearly, $T'_j \subseteq T''$, and since v_j is forgetful in T'_j, it must also be forgetful in T'', contradicting the assumptions of the lemma. Hence, we must have $T' = T''$. ■

Example 11: In the tree T of Figure 4 the state $s = 01$ is forgetful and the tree T_c on the right, which is obtained by extending s, is its MCE. Indeed, no state in T_c is forgetful. Notice that the states 010 and 011 (highlighted nodes in T_c), which result from extending s, unambiguously determine the next state for symbol 1, although symbol 0 does not determine the next state from state 010, which can be either 00100 or 00101. Thus, T_c is not FSM.

Next, we show that a tree T induces essentially the same close-ended type classes as its MCE T_c, which is reflected in the fact that our asymptotic results here and in Section VI will depend on parameters of T_c rather than T. For succinctness, we denote the type class of x^n with respect to T_c as $\mathcal{T}_c(x^n)$.

We start by studying how extending a forgetful state of T affects type class partitions.

Lemma 9: Let s be a forgetful state of T, and let T' be the tree obtained from T by a single extension of s. If two sequences belong to the same T-class*, they belong to the same T'-class.

Proof: Consider arbitrary symbols $a, b \in \mathcal{A}$ and the symbol count $n_{sb}^{(a)}$, where the latter is with respect to the state sb of T'. Since s is forgetful, $as \in \mathcal{I}(T)$ and, therefore, $asb \in T$. Thus, the emission of a in state sb causes a transition to a state (of both $S_{T'}$ and S_T) of the form $asbu$, $u \in \mathcal{A}^*$, implying that $n_{sb}^{(a)}$ is the total number of occurrences of states $asbu$. The latter, in turn, by the flow conservation equations (2) applied to T, is determined by the final state with respect to T and the total number of symbols emitted in states $asbu$, which is constant among all sequences in the same T-class*. ■

Corollary 1: If x^n and y^n are sequences in the same T-class that share the same final state with respect to $T_{\rm c}$, then they belong to the same $T_{\rm c}$-class*.

Proof: The claim is an immediate consequence of Lemma 9 and the sequential construction of $T_{\rm c}$, noting that if x^n and y^n share the same final state with respect to $T_{\rm c}$, then they also share the same final state in T and all the intermediate trees constructed on the way to $T_{\rm c}$. ■

For a tree T, we define the *state transition support*, denoted E_T, as the set of pairs $(s,t) \in S_T^2$ such that there is a transition from s to t in the state sequence of some string x^n. In view of Lemma 1, we have

$$E_T = \{\, (s,t) \in S_T^2 \,:\, s \preceq \mathsf{tail}(t) \text{ or } \mathsf{tail}(t) \prec s \,\}\,. \tag{19}$$

The following theorem presents a pointwise asymptotic tight estimate of the logarithm of the size of a T-class, provided all normalized pseudo-state transition counts are bounded away from zero. The second order term of the estimate is expressed in terms of the MCE $T_{\rm c}$.

Theorem 3: Let x^n be a sequence with pseudo-state sequence $\tilde{\mathbf{s}}$, of length m, with respect to an arbitrary context tree T. Then,[8]

$$\log|\mathcal{T}(x^n)| \;=\; m\hat{\mathcal{H}}(\tilde{\mathbf{s}}) - \frac{|E_{T_{\rm c}}| - |S_{T_{\rm c}}|}{2}\log n + O(1)\,, \tag{20}$$

provided the pseudo-state sequence of x^n with respect to $T_{\rm c}$ satisfies $\tilde{N}_{v,w} > \epsilon n$ for all $v, w \in \tilde{S}_{T_{\rm c}}$ of one of the forms (13)–(15), for some fixed positive constant ϵ.

When T is FSM, it is also canonical, we have $|E_T| = \alpha|S_T|$, and the pseudo-state sequence $\tilde{\mathbf{s}}(x^n)$ coincides with the state sequence, which implies that $\hat{\mathcal{H}}(\tilde{\mathbf{s}})$ is equal to the normalized empirical entropy

[8]We use standard asymptotic notation: $f(n) = O(g(n))$ if and only if $|f(n)| \le \kappa|g(n)|$ for some positive number κ and sufficiently large n, $f(n) = \Omega(g(n))$ if and only if $g(n) = O(f(n))$, and $f(n) = \Theta(g(n))$ if and only if $f(n) = O(g(n))$ and $f(n) = \Omega(g(n))$.

of x^n with respect to T. Thus, in this case, (20) is a standard consequence of well known facts for FSMs. We present the proof of Theorem 3 at the end of this section, following a few auxiliary lemmas. First, we recall some notions from graph theory, which we will rely upon in the sequel.

A *(directed) graph* is a pair $G = (V, E)$ where V is a finite set of *vertices* (or *nodes*) and E is a subset of $V \times V$. For an edge $e = (u, v)$ we call u the *source* of e, and v its *destination*. Both u and v are called *endpoints* of e. A path is a sequence of vertices, $v_0, v_1, \ldots, v_m$, such that $(v_i, v_{i+1}) \in E$ for all i, $0 \leq i < m$. We allow an arbitrary single vertex to represent an empty path. A path is *closed* if it starts and ends at the same vertex and it is *simple* if no edge (v_i, v_{i+1}) appears twice. A *circuit* is a closed simple path. We say that a (directed) graph is *strongly connected* if for any two vertices, u, w, there exists a path from u to w.

Definition 6: The *state transition support graph* of T, denoted G_T, is the graph with set of vertices S_T and set of edges E_T, defined in (19).

Definition 7: The *pseudo-state transition support graph* of T is a graph $\tilde{G}_T = (\tilde{S}_T, \tilde{E}_T)$, where a pair of pseudo-states (v, w) belongs to $\tilde{E}_T$ if and only if there exists a sequence x^n that yields $\tilde{N}_{v,w}(x^n) > 0$.

Notice that the state sequence of a string x^n, $\mathbf{s}(x^n)$, defines a path over the state transition support graph of T, and, analogously, the pseudo-state sequence of x^n, $\tilde{\mathbf{s}}(x^n)$, defines a path over the pseudo-state transition support graph of T.

Lemma 10: If T is canonical and (v, w) is a pair of pseudo-states satisfying one of (13)–(15), then there exists a fixed string $y_{(v,w)}$ such that for every sequence x^n that contains $y_{(v,w)}$, i.e., $x^n = z y_{(v,w)} z'$ for some $z, z' \in \mathcal{A}^*$, we have $\tilde{N}_{v,w}(x^n) > 0$.

Proof: First, observe that given $s \in S_T$, for all $i \geq 0$, there exists a string $a^i \in \mathcal{A}^i$ such that $a_i a_{i-1} \ldots a_1 s \notin \mathcal{I}(T)$. Indeed, proceeding by induction on i, the claim for $i = 1$ follows from the fact that s is not forgetful, so a symbol a_1 such that $\overleftarrow{s}\, a_1$ selects a state can be found; similarly, for $i > 1$, the claim follows from the fact that the state selected by $\overleftarrow{s}\, a^{i-1}$ is well defined and not forgetful, so an appropriate a_i can be found.

Now, given (v, w) satisfying (13), let $s \in S_T$ denote a state satisfying (5) for the pseudo-state w (namely, in (5), $s_h^{|s|} = w$), choose $i = d + 1$, the depth of T, and consider the string $y_{(v,w)} = \overleftarrow{s}\, a^{d+1}$, so that $x^n = z \overleftarrow{s}\, a^{d+1} z'$. Assume $x^j = z \overleftarrow{s}$. Clearly, by the definition of an extended context and the selection of s and a^{d+1}, we have $u_{j-|s|+|v|} = v$ and $u_{j-|s|+|v|+1} = w$, proving the claim for transitions of type (13) since no context-dropping transitions are inserted between these two extended contexts.

Next, consider (v, w) satisfying (14). Since $v\mathsf{head}(w) \notin \mathcal{I}(T)$, for $s = v \in S_T$ we can choose $a_1 = \mathsf{head}(w)$. Let $y_{(v,w)} = \overleftarrow{v}\, \mathsf{head}(w) a_2^{d+1}$ and consider again $x^j = z \overleftarrow{v}$. Clearly, $u_j = v$ and either

$u_{j+1} = w$ (with no context-dropping sequence inserted), or a context-dropping sequence starting with w is inserted after u_j. In either case, the claim is proven for transitions of type (14).

Finally, consider (v, w) satisfying (15), let t denote a descendant of $\mathsf{tail}(v)$ in S_T, and let $s = \sigma(\overleftarrow{t}\,\mathsf{head}(v)) \in S_T$. Clearly, s is well defined, with $s \prec v$ (by the existence of $\rho(v)$). Now, consider $x^n = z\overleftarrow{t}\,\mathsf{head}(v)a_1^{d+1}z'$ with $x^j = z\overleftarrow{t}$. It is then easy to see that $u_{j+1} = s$ and $u_j = t$. Since $\tilde{\sigma}(\overleftarrow{u_j}x_{j+1}) = \tilde{\sigma}(\overleftarrow{t}\,\mathsf{head}(v))$, of which v is a prefix, and $u_{j+1} \prec v$, a context-dropping sequence including the transition $(v, \rho(v))$ is inserted. ■

We recall from Lemma 3 that a necessary condition for $\tilde{N}_{v,w}(x^n) > 0$ is that the pseudo-state transition $v \to w$ is of one of the forms (13)–(15). However, in general, the condition is not sufficient for the existence of x^n such that $\tilde{N}_{v,w}(x^n) > 0$. Lemma 10 states the sufficiency of the condition for canonical trees. It also follows from the lemma that if $\langle T, p_T\rangle$ is a tree model such that T is canonical and all conditional probabilities are positive, then all pseudo-state transitions $v \to w$ satisfying one of (13)–(15) occur with positive probability. In particular, in this case, the assumptions of Theorem 3 hold with high probability.

It is well known that the state transition graph of a tree T is strongly connected. Lemma 10 allows us to establish, through the following corollary, that the same is true for the pseudo-state transition graph.

Corollary 2: The pseudo-state transition graph, $\tilde{G}_T$, of a tree T, is strongly connected.

Proof: Since every pseudo-state is either a state or the tail of a pseudo-state, for every $v \in \tilde{S}_T$ there exists $w \in \tilde{S}_T$ such that there is a transition from v to w satisfying either (13) or (14). Thus, by Lemma 10, there exists a corresponding outgoing edge in $\tilde{G}_T$. Similarly, for every $w \in \tilde{S}_T$ there exists an incoming edge (v, w) in $\tilde{G}_T$ with $v = \mathsf{tail}(w)$ (satisfying either (13) or (14)), except if $\mathsf{tail}(w) \in \mathcal{I}(T)$, in which case there must exist $t \in S_T$ such that $\mathsf{tail}(w) \prec t$ and therefore an incoming edge (v, w) satisfying (14) (if $w = \tilde{\sigma}(\overleftarrow{t}\,\mathsf{head}(w))$) or (15) (a context-dropping sequence initiated in $\tilde{\sigma}(\overleftarrow{t}\,\mathsf{head}(w))$). Now, given $v, w \in \tilde{S}_T$, let e be an edge with source v, and e' be an edge with destination w. Clearly, the pseudo-state sequence $\tilde{\mathbf{s}}(y_e y_{e'})$, with y_e and $y_{e'}$ as in Lemma 10, defines a path from v to w in $\tilde{G}_T$. Therefore, $\tilde{G}_T$ is strongly connected. ■

Lemma 11: For a canonical tree T, we have $|\tilde{E}_T| - |\tilde{S}_T| = |E_T| - |S_T|$.

We defer the proof of Lemma 11 to Appendix B, and proceed directly to the proof of the theorem.

Proof of Theorem 3: By Corollary 1, we can equivalently bound the maximum of $|\mathcal{T}_c^*(y^n)|$ among all sequences $y^n \in \mathcal{T}(x^n)$. Now, let $\mathbf{u} = u_0, u_1, \ldots, u_n$ and $\mathbf{u}' = u'_0, u'_1, \ldots, u'_n$ be the extended context sequences of an arbitrary sequence y^n with respect to T and T_c, respectively. We claim that $\mathbf{u}'$ and $\mathbf{u}$ must coincide, except, if u_n is a forgetful state of T, for the last r extended contexts, where $r \leq d$ and

we recall that $d = \mathsf{depth}(T) - 1$. By the sequential construction of T_{c} from T, it suffices to show that the claim is true when $\mathbf{u}'$ is the extended context sequence of y^n with respect to a tree T' obtained from T by a single extension of a forgetful state s. Clearly, since T' is an extension of T, we must have $u_i \preceq u'_i$, for all i, $0 \leq i \leq n$. If $u_i \prec u'_i$, then, by the definition of extended context sequence (Definition 1), we must have $y_j y_{j-1} \dots y_{i+1} u_i \in \mathcal{I}(T') \setminus \mathcal{I}(T) = \{s\}$ for some $j \geq i$. By the Definitions 1 and 5, a forgetful state can only be an extended context u_j if $j = n$. Thus, we must have $y_n y_{n-1} \dots y_{i+1} u_i = s$, which, since $|s| \leq d$ because s is forgetful and $|u_i| \geq 1$, implies that $i > n - d$.

By the claim above, for all $v, w \in \tilde{S}_T \cup \tilde{S}_{T_{\mathrm{c}}}$, the number of occurrences of a transition $v \to w$ in the pseudo-state sequence of a string y^n with respect to T and T_{c} differ in $O(1)$. In addition, if $y^n \in \mathcal{T}(x^n)$, by Theorem 2(i), we have either $y^n \in \mathcal{T}^*(x^n)$ or $y^{n-1} \in \mathcal{T}^*(x^{n-1})$. Therefore, by Lemma 4 and the fact that all but at most d of the last extended contexts of $\mathbf{u}(x^n)$ are determined by x^{n-1}, the pseudo-state transition counts of y^n and x^n with respect to T also differ in $O(1)$. As a consequence, since all normalized pseudo-state transition counts of x^n with respect to T_{c} are bounded away from zero, the logarithm of the multinomial factors in (9) for the pseudo-state sequences of x^n with respect to T and y^n with respect to T_{c} differ by $O(1)$. Furthermore, the cofactor M in (9) satisfies $M \leq 1$, and it can be lower-bounded by a positive constant, as in the proof of [7, Lemma 3] (in our case, relying on the strong connectivity of $\tilde{G}_T$ to satisfy the conditions of [7, Lemma A.1], and on the fact that all non-zero normalized counts are bounded away from zero). Thus, the claim follows by applying Stirling's approximation to (9) with respect to T_{c} and the fact, established in Section II-B, that $m \leq 2n + d$. ■

VI. The number of type classes

We study the number of type classes induced by a tree T on sequences of length n, which we denote by $\mathcal{N}_T$. The following theorem presents the main result of the section, which determines the rate of growth of $\mathcal{N}_T$ tightly, up to a multiplicative constant.

Theorem 4: Let T be a tree and let T_{c} be its MCE. Then,

$$\mathcal{N}_T = \Theta\left(n^{|E_{T_{\mathrm{c}}}| - |S_{T_{\mathrm{c}}}|}\right) . \tag{21}$$

Once again, when T is FSM, we have $T = T_{\mathrm{c}}$, $|E_{T_{\mathrm{c}}}| - |S_{T_{\mathrm{c}}}| = (\alpha - 1)|S_T|$, and (21) reduces to a known result for FSMs, as mentioned in Section I.

Let $\mathcal{N}_T^*$ denote the number of T-classes* induced by T. Clearly, $\mathcal{N}_T \leq \mathcal{N}_T^* \leq |S_T| \mathcal{N}_T$. Thus,

$$\mathcal{N}_T^* = \Theta(\mathcal{N}_T) , \tag{22}$$

and, for the purposes of Theorem 4, counting T-classes is equivalent to counting T-classes*. We will switch freely between the two partitions of $\mathcal{A}^n$. Furthermore, in view of Corollary 1, and what needs to be proved in Theorem 4, for the remainder of the section we assume, without loss of generality, that T is canonical.

By Lemma 2, there are at most as many close-ended type classes as different state transition matrixes compatible with sequences of states (not necessarily corresponding to any sequence of symbols). Thus, the upper bound part of (21) follows immediately by applying the analogous result for Markov chains, regarding state transitions in T as transitions in a Markov chain over S_T. Proving the lower bound, however, is more involved since some sequences of states may not correspond to a valid symbol sequence (as in Example 8). For completeness, we provide independent proofs of both the lower and upper bound parts of Theorem 4. For the lower bound, we will count the number of different matrixes $\tilde{N}(x^n)$ that arise as x^n varies in $\mathcal{A}^n$. We note that for sequences x^n, y^n in different type classes, the summation of all entries in $\tilde{N}(x^n)$ may be different from the summation over $\tilde{N}(y^n)$, depending on the number of context-dropping transitions in the pseudo-state sequences $\tilde{\mathbf{s}}(x^n)$ and $\tilde{\mathbf{s}}(y^n)$. Therefore, this problem is not equivalent to determining the number of first order Markov type classes over $\tilde{S}_T$, and will require additional tools to handle context-dropping transitions.

We define an *assignment of counters* for G as a vector in $\mathbb{N}^{|E|}$, indexed by elements of E. The assignment η is said to be *cyclic* if and only if it satisfies the flow conservation equations $\sum_{u:e=(u,v)} \eta_e = \sum_{w:e=(v,w)} \eta_e$ for all $v \in V$; η is said to be *connected* if eliminating edges e with $\eta_e = 0$ from E (while retaining all the vertices in V) results in a strongly connected graph. A *weight function* for G, ψ, assigns a nonnegative integer weight to each edge of G. We extend a weight function ψ to paths by defining the weight of a path as the sum of the weights of its edges, and to assignments of counters η by defining $\psi(\eta) = \sum_{e \in E} \psi(e)\eta_e$.

Notice that weight functions and assignment of counters are similar objects, as both assign a nonnegative integer to each edge of a graph. We distinguish between them, however, since they are intended for conceptually different purposes. We will regard weight functions as fixed objects defined on a given tree T. On the other hand, we will associate different cyclic assignments of counters on a certain graph to different T-classes*, and we will be interested in counting them to prove Theorem 4.

Lemma 12: Let $G = (V, E)$ be a strongly connected graph and ψ a fixed weight function for G such that G has no circuits of weight zero and at least one circuit of weight one. Then,

(i) The number of cyclic connected assignments of counters of weight n for G is $\Omega\left(n^{|E|-|V|}\right)$ as $n \to \infty$.

(ii) A cyclic assignment of counters η, of weight n, is fully determined by the values η_e for a set E^* of $|E| - |V|$ edges.

To avoid disrupting the flow of arguments leading to the proof of Theorem 4, we defer the proof of Lemma 12 to Appendix C. We notice that the proof of Theorem 4 for FSMs only requires a trivial weight function, in which $\psi(e) = 1$ for all edges e. A more general function ψ will allow us to deal with context-dropping transitions. The following lemma proves the upper bound part of Theorem 4, applying Lemma 12 to G_T, the state transition support graph of T defined in Section V.

Lemma 13: Let T be a canonical tree. Then, $\mathcal{N}_T = O\left(n^{|E_T|-|S_T|}\right)$.

Proof: Let ψ be a weight function for G_T, with $\psi(e) = 1$ for all edges e of G_T. Clearly, G_T is strongly connected, and it has no circuits of weight zero. Pick an arbitrary symbol $b \in \mathcal{A}$. The tree T has a state of the form b^ℓ, $\ell \geq 1$, which transitions to itself with the symbol b. Therefore, there is a circuit of weight one in G_T, and the graph satisfies the assumptions of Lemma 12. Now, for each state s of T, let γ_s be a fixed path from s to s_0 in G_T. Consider a string x^n with final state s_n in T. Let η be an assignment of counters for G_T defined as $\eta_{(u,v)} = N_{u,v}(x^n)$ for all $u, v \in S_T$ and, similarly, let the assignment of counters η' count transitions in the path γ_{s_n}. The assignment of counters $\eta'' = \eta + \eta'$ is cyclic by construction. Hence, by Lemma 12(ii), η'' can be fully described by the values η''_e for a set of $|E_T|-|S_T|$ edges. Given s_n, γ_{s_n} is fixed, and therefore this is equivalent to describing the values η_e for the same set of edges. Thus, η can be fully described by giving the final state s_n in T, which determines γ_{s_n}, and the values η_e for a set of $|E_T|-|S_T|$ edges. Since each value η_e is at most n, and there are a constant number of states in T, we have $\mathcal{N}^*_T = O\left(n^{|E_T|-|S_T|}\right)$, and, by (22), $\mathcal{N}_T = O\left(n^{|E_T|-|S_T|}\right)$. ■

As to the lower bound part of Theorem 4, as before, it suffices to show that $\mathcal{N}^*_T = \Omega\left(n^{|E_T|-|S_T|}\right)$. For an FSM tree, this amounts simply to apply Lemma 12(i) to G_T. In general, as mentioned, the proof is more involved since some closed paths in G_T may not correspond to a valid state sequence (in analogy to Example 8). Instead, we will apply Lemma 12 to the pseudo-state transition support graph of T, $\tilde{G}_T = (\tilde{S}_T, \tilde{E}_T)$. We make use of the following auxiliary lemma.

Lemma 14: Let η be a cyclic connected assignment of counters for a graph $G = (V, E)$ and let $v_0 \in V$ be an arbitrary vertex. Then, there exists a closed path, $\mathbf{v} = v_0, v_1, \ldots, v_r$, whose edge occurrence counts are given by η.

Proof: Consider a directed multigraph, i.e., a graph allowed to contain multiple copies of the same edge, with set of vertices V and edges taken from E, with η_e copies of each edge e. By the definition of cyclic connected assignment, it follows from well known results (see, e.g., [23]), that this multigraph contains an Eulerian circuit, i.e., a path that traverses each edge e as many times as the number of copies

of the edge. Moreover, since η is connected, the circuit visits all vertices and, thus, we can assume that the initial (and final) vertex is v_0. The claim then follows by letting $\mathbf{v}$ be one of such paths. ■

Next, we apply Lemma 12 to $G = \tilde{G}_T$ with $\psi(e) = |\omega(e)|$, where we extend the definition of the tagging function ω, defined in Section III-C, to edges of $\tilde{G}_T$, assigning to an edge (v, w) the same tag as a transition $v \to w$ in a sequence of pseudo-states. We are interested in cyclic connected assignments of counters of weight n in $\tilde{G}_T$, and we show in the proof of the following lemma that the number of such assignments of counters lower bounds $\mathcal{N}_T^*$. This provides the last argument needed for the proof of Theorem 4, which follows immediately after the lemma.

Lemma 15: For a canonical tree T we have $\mathcal{N}_T^* = \Omega\left(n^{|\tilde{E}_T|-|\tilde{S}_T|}\right)$.

Proof: First, we recall from Corollary 2, that $\tilde{G}_T$ is strongly connected. Let ψ be the weight function for $\tilde{G}_T$ defined by $\psi(e) = |\omega(e)|$. Since there are no circuits formed by edges of the form $(v, \rho(v))$ in $\tilde{G}_T$, by the definition of ω, there are no circuits of weight zero in $\tilde{G}_T$. Also, for $b \in \mathcal{A}$ and the state $v = b^\ell$, $\ell \geq 1$, of T, there is a circuit of weight one starting from v. Indeed, either $vb \notin \tilde{S}_T$, in which case v, v is a single-edge circuit of weight one or, by Lemma 10, $(v, \tilde{\sigma}(\overleftarrow{v}\, b))$ is an edge of $\tilde{G}_T$ (of type (14)). In the latter case, $v = \rho(bv)$ and therefore, by Lemma 10, (bv, v) is an edge of $\tilde{E}_T$, of type (15), with weight zero. Hence, the assumptions of Lemma 12 hold for $\tilde{G}_T$.

Let Φ be the set of cyclic connected assignments of counters of weight n for $\tilde{G}_T$, and let $\mathcal{N}_T^\circ$ denote the number of close-ended type classes for sequences of length n with final state $s_n = s_0$. We claim that $\mathcal{N}_T^\circ \geq |\Phi|$. Let $\eta \in \Phi$ and, by Lemma 14 applied to $\tilde{G}_T$, let $\tilde{\mathbf{v}}$ be a sequence of pseudo-states starting and ending at s_0 with transition counts given by η. The sequence of pseudo-states $\tilde{\mathbf{v}}$ defines, via the tagging function ω, a string $x^n = \omega(\tilde{\mathbf{v}})$, which is of length n by the definitions of ψ and ω. By Lemma 5(i), the final state of x^n is s_0, the final pseudo-state of $\tilde{\mathbf{v}}$. In addition, $\omega(\tilde{\mathbf{s}}(x^n)) = x^n$ and the last pseudo-state of $\tilde{\mathbf{s}}(x^n)$ is also s_0, which implies, by Lemma 5(ii), that $\tilde{\mathbf{s}}(x^n) = \tilde{\mathbf{v}}$. Thus, for every $\eta \in \Phi$, there exists a string x^n with pseudo-state transition matrix $\tilde{N}(x^n) = \eta$, where by the latter we mean $\tilde{N}_{v,w}(x^n) = \eta_{(v,w)}$ for all $(v, w) \in \tilde{E}_T$. Now, if $\eta' \in \Phi$, with $\eta' = \tilde{N}(y^n)$, and $\eta' \neq \eta$, then we must have $\mathcal{T}^*(x^n) \neq \mathcal{T}^*(y^n)$ since, by Lemma 4, $\tilde{N}(x^n)$ is determined by $\mathcal{T}^*(x^n)$. Thus, $\mathcal{N}_T^\circ \geq |\Phi|$, as claimed. The claim of the lemma now follows, since $\mathcal{N}_T^* \geq \mathcal{N}_T^\circ$ and, by Lemma 12(i), which holds for $\tilde{G}_T$, we have $|\Phi| = \Omega\left(n^{|\tilde{E}_T|-|\tilde{S}_T|}\right)$.

■

Proof of Theorem 4: Recall that by Corollary 1, we have $\mathcal{N}_T = \mathcal{N}_{T_c}$. The upper bound $\mathcal{N}_{T_c} = O\left(n^{|E_c|-|S_c|}\right)$ follows from Lemma 13. The lower bound $\mathcal{N}_{T_c} = \Omega\left(n^{|E_c|-|S_c|}\right)$ follows from Lemma 15 and Lemma 11, recalling also that $\mathcal{N}_{T_c} = \Theta(\mathcal{N}_{T_c}^*)$. ■

APPENDIX A
PROOF OF LEMMA 7

We first show that if $s_{n-1}(x^n)$ and x_n determine $s_n(x^n)$, then $s_n(y^n) = s_n(x^n)$. Notice that $s_{n-1}(x^n)$ and x_n determine $s_n(x^n)$ if and only if $\mathsf{tail}(s_n(x^n)) \in T$. Consider a state s such that $\mathsf{tail}(s) \in T$, and let $u = \mathsf{tail}(s)$, $a = \mathsf{head}(s)$. By Lemma 1, *all* state transitions into s are of the form $uv \to s$, $v \in \mathcal{A}^*$, and, by the definition of state selection in trees, the emission of symbol a from a state uv generates a transition to state s. Hence,

$$N_{*au}(x^n) = \sum_{uv \in S_T} n_{uv}^{(a)}(x^n) = \sum_{uv \in S_T} n_{uv}^{(a)}(y^n) = N_{*au}(y^n)\,.$$

Also, by the flow conservation equations (2) on the transition matrices, we have

$$\begin{aligned} N_{*au}(x^n) &= N_{au*}(x^n) + \delta_{au,s_n(x^n)} - \delta_{au,s_0} \\ N_{*au}(y^n) &= N_{au*}(y^n) + \delta_{au,s_n(y^n)} - \delta_{au,s_0}\,. \end{aligned}$$

Since the total number of symbols emitted from au in x^n and y^n is the same, we have $N_{au*}(x^n) = N_{au*}(y^n)$, and therefore $\delta_{au,s_n(x^n)} = \delta_{au,s_n(y^n)}$, which proves the claim.

As a consequence of the claim above, if $s_n(x^n) \neq s_n(y^n)$, we must have $s_n(x^n) = auv$ with $a \in \mathcal{A}$, $u \in S_T$, and $v \in \mathcal{A}^+$. Notice that au is therefore an internal node of T. Now, by Lemma 1, all state transitions into a state of the form auw, $w \in \mathcal{A}^+$, originate from state u and, by the definition of state selection in trees, the emission of symbol a from a state u generates a transition to a state of the form auw. Hence,

$$\begin{aligned} n_u^{(a)}(x^n) &= \sum_{auw \in S_T} N_{*auw}(x^n) \\ &= \sum_{auw \in S_T} \left[N_{auw*}(x^n) + \delta_{auw,s_n(x^n)} - \delta_{auw,s_0}\right]\,, \end{aligned}$$

and also,

$$n_u^{(a)}(y^n) = \sum_{auw \in S_T} \left[N_{auw*}(y^n) + \delta_{auw,s_n(y^n)} - \delta_{auw,s_0}\right]\,.$$

Since $n_u^{(a)}(y^n) = n_u^{(a)}(x^n)$, and the total number of symbols emitted from auw in x^n and y^n is the same, we have,

$$\sum_{auw \in S_T} \delta_{auw,s_n(y^n)} = \sum_{auw \in S_T} \delta_{auw,s_n(x^n)} = 1\,.$$

We conclude that the final state of y^n has the form auw. Thus, $s_{n-1}(x^n) = s_{n-1}(y^n) = u$, and also $x_n = y_n = a$, as claimed. ■

APPENDIX B
PROOF OF LEMMA 11

We describe a procedure to transform the graph G_T into the graph $\tilde{G}_T$ and we show that each intermediate graph in the transformation, $G = (V, E)$, satisfies $|E| - |V| = |E_T| - |S_T|$. By Lemma 3 and Lemma 10, $\tilde{E}_T$ contains all edges (v, w) with v, w of the form (13)–(15), and, by Lemma 1, all edges (s, t) of E_T are either of the form $s \preceq \mathsf{tail}(t)$ or $\mathsf{tail}(t) \prec s$. For each edge $(s, t) \in E_T$ of the form $s \preceq \mathsf{tail}(t)$, let $\mu_1, \ldots, \mu_\ell$ be the longest sequence of consecutive suffixes of t, ending at $\mu_\ell = t$, such that all these suffixes are pseudo-states (hence, $\mathsf{tail}(\mu_1) \in \mathcal{I}(T)$). Notice that, since $s \preceq \mathsf{tail}(t)$, $\mathsf{tail}(t) = \mu_{\ell-1}$ and we must have $\ell > 1$. If none of these pseudo-states, other than t itself, are vertices of the graph under construction, insert the edges (μ_{j-1}, μ_j), $1 < j \leq \ell$, together with the vertices μ_j, $1 \leq j < \ell$. In addition, since, under these assumptions, $\mu_1 \notin S_T$, insert the edges $(v_1, v_2), (v_2, v_3), \ldots, (v_{k-1}, v_k)$, together with the vertices $v_2, \ldots, v_{k-1}$, that comprise a context-dropping sequence from $\mu_1 = v_1$ to the first vertex, v_k, that already belongs to the graph constructed so far (v_k is well defined because some state must be a prefix of μ_1, and all states belong to the initial set of vertices S_T). Finally delete the original edge (s, t). Clearly, accounting for the number of vertices and edges that were added, and the edge that was deleted, the quantity $|E| - |V|$ remains unchanged. Otherwise, if some μ_j other than t is already a vertex of the graph constructed so far, let μ_i be the last such vertex in the sequence $\mu_1, \ldots, \mu_\ell$. Suppress the edge (s, t) and add the edges (μ_{j-1}, μ_j), $i < j \leq \ell$, together with the vertices μ_j, $i < j < \ell$. We claim that none of these edges could have been added before. Indeed, if $i < \ell - 1$, at least one endpoint of the edge did not belong to the graph. If, instead, $i = \ell - 1$, then a previous insertion of the edge $(\mu_{\ell-1}, t)$ would have required t to be a pseudo-state in a sequence $\mu'_1, \ldots, \mu'_{\ell'} = t'$ for some $t' \in S_T$, and the vertex associated with t to not belong to the graph, in contradiction with $t \in S_T$. Thus, again, the quantity $|E| - |V|$ is preserved by this transformation.

After having replaced all edges (s, t) of the form $s \preceq \mathsf{tail}(t)$, by the definition of $\tilde{S}_T$ in (5), all pseudo-states have been added to the set of vertices and, by Lemma 3, all edges (v, w) of $\tilde{G}_T$, of the form $v = \mathsf{tail}(w)$ and of the form $w = \rho(v)$ with $\mathsf{tail}(v) \in \mathcal{I}(T)$, have been added to the set of edges. To complete the construction of $\tilde{G}_T$ it remains to add edges $(v, \rho(v))$ with $\mathsf{tail}(v) \in S_T$, as well as edges (v, w) satisfying (14) with $v \neq \mathsf{tail}(w)$. We add these edges next, in replacement of edges (s, t) of E_T of the form $\mathsf{tail}(t) \prec s$. Let (s, t) be one such edge of E_T and let $b = \mathsf{head}(t)$. If $bs \in \tilde{S}_T$ we replace (s, t) by an edge $(bs, \rho(bs))$, which is well defined since $t \prec bs$, and satisfies $\mathsf{tail}(bs) \in S_T$. If, otherwise, $\tilde{\sigma}(\overleftarrow{s}\, b) \neq bs$, we replace the edge (s, t) by an edge $(v, w) = (s, \tilde{\sigma}(\overleftarrow{s}\, b))$ (where t and w may coincide),

where we notice that (v, w) satisfies (14) and $v \neq \mathsf{tail}(w)$. The replacement of all edges (s, t) of E_T of the form $\mathsf{tail}(t) \prec s$, as described, completes the construction of $\tilde{G}_T$. Indeed, each edge $(v, \rho(v))$ with $\mathsf{tail}(v) \in S_T$ has been inserted in replacement of the edge (s, t) of E_T, where $s = \mathsf{tail}(v)$ and $t = \sigma(\overleftarrow{v})$, which is of the form $\mathsf{tail}(t) \prec s$ and, since $v = \mathsf{head}(t)s$, satisfies $bs \in \tilde{S}_T$, where $b = \mathsf{head}(t)$. Similarly, each edge (v, w) satisfying (14) with $v \neq \mathsf{tail}(w)$ has been inserted in replacement of the edge (s, t) of E_T, where $s = v$ and $t = \sigma(\overleftarrow{w})$, which is of the form $\mathsf{tail}(t) \prec s$ and satisfies $\tilde{\sigma}(\overleftarrow{s}\, b) \neq bs$, where $b = \mathsf{head}(t)$. Since, clearly, these substitutions do not alter the value of $|E| - |V|$, the lemma is proved. ■

APPENDIX C
PROOF OF LEMMA 12

Before we proceed to the proof of Lemma 12, we review some additional graph-theoretic tools. We loosely follow [24]. Consider a directed graph $G = (V, E)$. A *chain* is an alternating sequence of vertices and edges $v_1, e_1, v_2, e_2, \ldots, v_m, e_m, v_{m+1}$ satisfying either $e_i = (v_i, v_{i+1})$ or $e_i = (v_{i+1}, v_i)$. A chain is *closed* if $v_1 = v_{m+1}$, it is *simple* if $e_i \neq e_j$ for $i \neq j$, and it is *elementary* if all vertices are different, except possibly for v_1 and v_{m+1}, which may coincide. The number of edges in a chain is called the *length* of the chain. A *cycle* is a closed simple chain. Notice that a path corresponds to a chain where every edge is traversed in the forward direction, i.e., $e_i = (v_i, v_{i+1})$ for all $i = 1, \ldots, m$ (for a path the alternation of vertices between edges is unnecessary for disambiguation and, thereby, omitted). Similarly, a circuit is a cycle where every edge is traversed in the forward direction.

We say that a graph is *connected* if for any two vertices, u, w, there exists a chain that joins u with w, i.e., a chain of the form $u = v_1, e_1, v_2, e_2, \ldots, v_m, e_m, v_{m+1} = w$. In this setting, a tree is a graph that is connected and has no cycles.[9] A *spanning tree* of $G = (V, E)$ is a tree, $G' = (V, E')$, with the same set of vertices as G and with $E' \subseteq E$. A *sink* of a tree $\mathbb{T}$ is a vertex $u \in \mathbb{T}$ such that there exists a path from v to u for all $v \in \mathbb{T}$ (a tree can have at most one sink; if we reverse the direction of all the tree edges, a sink becomes a root and vice versa).

[9]For conciseness, and with a slight abuse of terminology, so far we had used the term 'tree' as shorthand for *context tree*, which includes a tree as defined here, with some additional properties such as the existence of a root, and the labeling of the branches. We will continue to use the term 'tree' in these two senses, with the meaning being clearly determined by the context and the notation. Also, notice that since the direction of an edge is not relevant for the construction of a chain or a cycle, the notions of connected graph and tree coincide with the usual definitions for non-directed graphs. Our derivations, however, will still be based on directed graphs.

We define a function ζ mapping each chain $\gamma = v_1, e_1, \ldots, v_r, e_r, v_{r+1}$ in G to a vector $\zeta(\gamma) \in \mathbb{Z}^{|E|}$, indexed with elements from E, and defined by

$$\zeta(\gamma)_e = |\{i = 1, \ldots, r : e = (v_i, v_{i+1})\}| - |\{i = 1, \ldots, r : e = (v_{i+1}, v_i)\}| .$$

We denote by $\mathbb{R}$ the set of real numbers. The subspace of $\mathbb{R}^{|E|}$ spanned by $\{\zeta(\gamma) : \gamma \text{ is a cycle }\}$ is called the *cycle space* of G and it is known to have dimension $|E| - |V| + 1$ for strongly connected graphs [24]. A *circuit basis* for the cycle space is a basis formed by vectors $\zeta(c_i)$ where every c_i is an elementary circuit.

Proposition 1 ([24]): Every strongly connected graph has a circuit basis.

Notice that if $\zeta(c_i)$ is a vector in a circuit basis for G, then $\zeta(c_i)$ defines a cyclic assignment of counters for G.

Proof of Lemma 12: Let c' be a circuit of G of weight one. Exactly one edge of c' has weight one, and the rest, if any, have weight zero. Hence, c' must be elementary, since otherwise it could be split into two circuits, one of which would be of weight zero, in contradiction with the assumptions of the lemma. Since $\zeta(c')$ belongs to the cycle space of G, we can assume, without loss of generality, that it is one of the elements in a circuit basis, $C = \{\zeta(c_1), \ldots, \zeta(c_{|E|-|V|+1})\}$, of G, with, say $c_1 = c'$.

Consider arbitrary vertices $u, v \in V$. Since G is strongly connected, there exists a circuit γ that passes through u and v. Now, since $\zeta(\gamma)$ belongs to the cycle space of G, we can expand it in terms of the basis C as $\zeta(\gamma) = \sum_{i=1}^{|E|-|V|+1} \alpha_i \zeta(c_i)$, with $\alpha_i \in \mathbb{R}$. Thus, the set of edges of γ is a subset of the union of the set of edges of all circuits c_i. Let $\eta = \sum_{i=1}^{|E|-|V|+1} k_i \zeta(c_i)$ be any linear combination with coefficients $k_i \in \mathbb{N}_+$. Clearly, η defines a cyclic assignment of counters for G, since each basis element $\zeta(c_i)$ does. Moreover, since from the set of all edges in the circuits c_i we can construct a circuit γ that passes through u and v for *any* pair of vertices u, v, the cyclic assignment η is also connected. Now, C is a basis of the cycle space of G, so different linear combinations must generate different cyclic connected assignments of counters for G. Thus, since $\psi(\eta) = \sum_{i=1}^{|E|-|V|+1} \psi(c_i) k_i$, there are *at least* as many cyclic connected assignments of counters of weight no greater than n, as compositions of $\ell = \lfloor n / \max_i \psi(c_i) \rfloor$ of the form

$$\ell = \sum_{i=1}^{|E|-|V|+1} k_i \, . \tag{23}$$

Each assignment η obtained in this way can be completed to an assignment η' of weight n by replacing k_1 with $k_1 + n - \psi(\eta)$, since c_1 is a circuit of weight one. Specifically, if $\eta = \sum_i k_i \zeta(c_i)$, we take

$$\eta' = (k_1 + n - \psi(\eta))\zeta(c_1) + \sum_{i>1} k_i \zeta(c_i) \, .$$

This way, different linear combinations with coefficients k_i satisfying (23) generate different connected assignments of counters of weight *exactly* n. Thus, we have at least as many cyclic connected assignments of counters of weight n, as compositions of ℓ in $|E| - |V| + 1$ positive summands. The proof of (i) is completed by recalling that the number of such compositions is $\binom{\ell-1}{|E|-|V|}$, which, since $|E| - |V|$ is a constant, is $\Omega\left(\ell^{|E|-|V|}\right)$, and also $\Omega\left(n^{|E|-|V|}\right)$, since $\ell = \Theta(n)$.

We now turn to (ii). Let $\gamma = e_1, e_2, \ldots, e_r$ be an elementary circuit of weight one in G, with $\psi(e_r) = 1$, and $\psi(e_i) = 0$ for $i = 1, \ldots, r-1$. We construct a spanning tree, $\mathbb{T}$, of G, as follows:

1. Set $\mathbb{T}$ to the set of edges $\{e_1, \ldots, e_{r-1}\}$, with their r distinct adjacent vertices.
2. If there are vertices of G that are not in $\mathbb{T}$, choose an edge e such that the destination of e is in $\mathbb{T}$, but the source is not in $\mathbb{T}$. Such an edge must exist, since G is strongly connected. Add e to $\mathbb{T}$.
3. Repeat Step 2 until all vertices in G are also in $\mathbb{T}$.

It can readily be verified that $\mathbb{T}$ is a spanning tree of G with a sink at the source of e_r, and with a set of edges $E_{\mathbb{T}} \supseteq \{e_1, \ldots, e_{r-1}\}$. We will show that the values of η_e for $e \in E_{\mathbb{T}} \cup \{e_r\}$ can be computed from the remaining values, which we regard as given. Let V_γ be the set of vertices of γ, and for $v \in V$, let $d(v)$ be the distance in $\mathbb{T}$ from v to V_γ, i.e., the length of the unique path in $\mathbb{T}$ from v to a vertex in V_γ. For each $v \in V \setminus V_\gamma$ we show how to compute η_{e_v} for the unique edge e_v, with source v, which belongs to $E_{\mathbb{T}}$. Take all the vertices $v \in V \setminus V_\gamma$ in decreasing order of $d(v)$. For each edge $e = (u, v)$ with destination v, either $e \notin E_{\mathbb{T}}$, or $d(u) = d(v) + 1$. In any case, the value η_e is known, either because it is given, or, since we take vertices in decreasing order of $d(v)$, because it has already been computed. Since the value η_e is known for all edges $e \neq e_v$ with source v, we can compute η_{e_v} from the flow conservation equation $\sum_{u:e=(u,v)} \eta_e = \sum_{w:e=(v,w)} \eta_e$. After finishing this process, we know η_e for all edges e except for those in γ, the unique circuit of $E_{\mathbb{T}} \cup \{e_r\}$. Since $\psi(e_1) = \ldots \psi(e_{r-1}) = 0$, we can now compute η_{e_r} as $n - \sum_{e \in E \setminus \{e_1, \ldots, e_r\}} \eta_e \psi(e)$ and continue calculating η_e for e=e_1, $\ldots$, e_{r-1}, using the flow conservation equations. ■

ACKNOWLEDGMENT

We thank the anonymous reviewers for their thoughtful and insightful comments, which contributed to improving the presentation of our results.

REFERENCES

[1] I. Csiszár and J. Körner, *Information Theory: Coding Theorems for Discrete Memoryless Systems*. New York: Academic, 1981.

[2] I. Csiszár, "The method of types," *IEEE Trans. Inform. Theory*, vol. 44, no. 6, pp. 2505–2523, Oct. 1998.

[3] R. B. Ash, *Information Theory*. John Wiley & Sons, Inc., 1967.

[4] L. D. Davisson, G. Longo, and A. Sgarro, "The error exponent for the noiseless encoding of finite ergodic Markov sources," *IEEE Trans. Inform. Theory*, vol. IT-27, no. 4, pp. 431–438, Jul. 1981.

[5] I. Csiszár, T. M. Cover, and B.-S. Choi, "Conditional limit theorems under Markov conditioning," *IEEE Trans. Inform. Theory*, vol. IT-33, no. 6, pp. 788–801, Nov. 1987.

[6] P. Jacquet and W. Szpankowski, "Markov types and minimax redundancy for Markov sources," *IEEE Trans. Inform. Theory*, vol. 50, no. 7, pp. 1393–1402, Jul. 2004.

[7] M. J. Weinberger, N. Merhav, and M. Feder, "Optimal sequential probability assignment for individual sequences," *IEEE Trans. Inform. Theory*, vol. 40, no. 2, pp. 384–396, Mar. 1994.

[8] N. Merhav and M. J. Weinberger, "On universal simulation of information sources using training data," *IEEE Trans. Inform. Theory*, vol. 50, no. 1, pp. 5–20, Jan. 2004.

[9] G. Seroussi, "On universal types," *IEEE Trans. Inform. Theory*, vol. 52, no. 1, pp. 171–189, Jan. 2006.

[10] Á. Martín, N. Merhav, G. Seroussi, and M. J. Weinberger, "Twice-universal simulation of Markov sources and individual sequences," *IEEE Trans. Inform. Theory*, vol. 56, no. 9, pp. 4245–4255, Sep. 2010.

[11] J. Rissanen, "Complexity of strings in the class of Markov sources," *IEEE Trans. Inform. Theory*, vol. IT-32, no. 4, pp. 526–532, Jul. 1986.

[12] M. J. Weinberger, J. Rissanen, and M. Feder, "A universal finite memory source," *IEEE Trans. Inform. Theory*, vol. 41, pp. 643–652, May 1995.

[13] F. M. J. Willems, Y. M. Shtarkov, and T. J. Tjalkens, "The context-tree weighting method: Basic properties," *IEEE Trans. Inform. Theory*, vol. IT-41, pp. 653–664, May 1995.

[14] G. Seroussi and M. J. Weinberger, "On tree sources, finite state machines, and time reversal," in *Proc. International Symposium on Information Theory*, Whistler, BC, Canada, Sep. 1995, p. 390.

[15] Á. Martín, G. Seroussi, and M. J. Weinberger, "Linear time universal coding and time reversal of tree sources via fsm closure," *IEEE Trans. Inform. Theory*, vol. 50, no. 7, pp. 1442–1468, Jul. 2004.

[16] P. L. Buhlmann and A. Wyner, "Variable length Markov chains," *Annals of Statistics*, vol. 27, pp. 480–513, 1998.

[17] Á. Martín, G. Seroussi, and M. J. Weinberger, "Enumerative coding for tree sources," in *Festschrift in Honor of Jorma Rissanen on the Occasion of his 75th Birthday*, ser. 38, P. Grünwald, P. Myllymäki, I. Tabus, M. J. Weinberger, and B. Yu, Eds. Tampere: Tampere University of Technology, Tampere International Center for Signal Processing, 2008, pp. 93–116.

[18] P. Whittle, "Some distribution and moment formulae for the Markov chain," *J. Roy. Statist. Soc. Ser. B*, vol. 17, no. 3, pp. 235–242, 1955.

[19] P. Billingsley, "Statistical methods in Markov chains," *Annals Math. Stat.*, vol. 32, pp. 12–40, 1961.

[20] L. Goodman, "Exact probabilities and asymptotic relationships for some statistics from m-th order Markov chains," *Annals of Mathematical Statistics*, vol. 29, pp. 476–490, 1958.

[21] J. Rissanen and G. G. Langdon, "Universal modeling and coding," *IEEE Trans. Inform. Theory*, vol. IT-27, pp. 12–23, Jan. 1981.

[22] T. M. Cover, "Enumerative source encoding," *IEEE Trans. Inform. Theory*, vol. IT-19, pp. 73–77, Jan. 1973.

[23] S. Even, *Graph Algorithms*. Potomac, Maryland: Computer Science Press, 1979.

[24] C. Berge, *Graphs*. Amsterdam: North-Holland, 1985.