Relevant Representations for the Inference of Rational Stochastic Tree Languages*

François Denis[1], Édouard Gilbert[2], Amaury Habrard[1], Faïssal Ouardi[1], and Marc Tommasi[2]

[1] Laboratoire d'Informatique Fondamentale, CNRS, Aix-Marseille Université
{francois.denis,amaury.habrard,faissal.ouardi}@lif.univ-mrs.fr
[2] Laboratoire d'Informatique Fondamentale de Lille (L.I.F.L.), INRIA and É.N.S. Cachan
{edouard.gilbert,marc.tommasi}@inria.fr

Abstract. Recently, an algorithm - DEES- was proposed for learning rational stochastic tree languages. Given a sample of trees independently and identically drawn according to a distribution defined by a rational stochastic language, DEES outputs a linear representation of a rational series which converges to the target. DEES can then be used to identify in the limit with probability one rational stochastic tree languages. However, when DEES deals with finite samples, it often outputs a rational tree series which does not define a stochastic language. Moreover, the linear representation can not be directly used as a generative model. In this paper, we show that any representation of a rational stochastic tree language can be transformed in a reduced normalised representation that can be used to generate trees from the underlying distribution. We also study some properties of consistency for rational stochastic tree languages and discuss their implication for the inference. We finally consider the applicability of DEES to trees built over an unranked alphabet.

1 Introduction

In this paper, we consider the problem of learning probability distribution over trees from a sample of trees independently and identically distributed (i.i.d.), in a given class of models. In this context, the learning process has two main objectives: Finding the correct structure of the representation and estimating precisely the parameters of the model. Because we adopt a machine learning standpoint, we restrict ourselves to classes of probabilistic languages that can be somehow finitely presented. Probabilistic tree automata (PTA) are a usual representations for rational stochastic tree languages (RSTL). In a PTA, each rule is equipped with a weight in $[0, 1]$ and a per state normalisation is imposed. Nonetheless, a first drawback is that it may be not decidable to know whether a PTA is consistent *i.e.* whether it represents a probability distribution on trees.

* This work was partially supported by the Atash project ANR-05-RNTL00102 and the Marmota project ANR-05-MMSA-0016.

A. Clark, F. Coste, and L. Miclet (Eds.): ICGI 2008, LNAI 5278, pp. 57–70, 2008.

One difficulty comes from the fact that a RSTL may be such that the average size of trees may be undefined. A second drawback of PTA is that they admit no canonical representation. Thus, most of learning algorithms approaches based on grammatical inference fail for the class of PTA.

Recent approaches have proposed to work in a larger class of representation: The class of rational stochastic tree languages that can be represented under a linear form of a tree series. The models of this class can be equivalently representing by weighted tree automata with parameters in $\mathbb{R}$ (hence with weights that can be negative and without any per state normalisation condition). This class has two interesting properties: It has a high level of expressiveness since it strictly includes the class of RSTL and it admits a canonical form with a minimal number of parameters. Based on these properties, linear representations of RSTL are a good candidate from a grammatical inference standpoint. A recent algorithm, DEES, able to identify in the limit with probability one the class of rational stochastic tree languages RSTL was proposed in [1]. However, this algorithm has two main drawbacks when working with finite samples. It often outputs a rational tree series that does not define a stochastic language, and the representation of the series can not be directly used as a generative model. This comes from the fact that the canonical representation is more adapted for finding the structure of the model and estimating the parameters. We do not obtain a representation of a probability distribution that factorises into a product of probabilities associated with each state. When we need a generative model, we claim that we have to use another representation. Our first contribution is to show that any canonical representation of a rational stochastic tree language admits a normalised reduced representation of the same size which can be easily used in a generative process. Then, we examine some conditions of consistency for rational stochastic languages. Indeed, as for probabilistic context-free grammars [2,3], the consistency can not be ensured only with syntactical properties. We discuss then the influence of these conditions to the problem of inferring rational stochastic tree languages. We finish by studying the applicability of our approach to trees that are built from an unranked alphabet. Actually, a bijection can be made between the unranked representation and a ranked one, allowing us to apply our algorithm to the unranked case.

The paper is organized as follows. Definitions and notations are presented in Section 2. Section 3 deals with the normalised reduced representation of rational stochastic tree language. The consistency conditions are evoked in Section 4. The paper terminates by Section 5 on unranked trees.

2 Preliminaries

In this section, we recall definitions of trees, (rational) tree series, weighted automata and (rational) stochastic tree languages. We mainly follow notations and definitions from [4] about trees and tree automata. Formal power tree series have been introduced in [5] where the main results appear.

Trees and Contexts. Let $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1 \cup \cdots \cup \mathcal{F}_p$ be a ranked alphabet where the elements in $\mathcal{F}_m$ are the function symbols of rank m. Let $\mathcal{X}$ be a countable set of variables. The set $T(\mathcal{F}, \mathcal{X})$ is the smallest set satisfying: $\mathcal{F}_0 \cup \mathcal{X} \subseteq T(\mathcal{F}, \mathcal{X})$, for $f \in \mathcal{F}_m,\ m \geq 1$, and $t_1, \ldots, t_m \in T(\mathcal{F}, \mathcal{X})$, $f(t_1, \ldots, t_m) \in T(\mathcal{F}, \mathcal{X})$.

We call *trees*, elements in $T(\mathcal{F}, \emptyset) = T(\mathcal{F})$. For any tree t, let us denote by $|t|_f$ the number of occurrences of the symbol $f \in \mathcal{F}$ in t and by $|t|$, the size $\sum_{f \in \mathcal{F}} |t|_f$ of t. The *height* of a tree t is defined by: $\text{height}(t) = 0$ if $t \in \mathcal{F}_0$ and $\text{height}(t) = 1 + \max\{\text{height}(t_i) | i = 1..m\}$ if $t = f(t_1, \ldots, t_m)$. We suppose given a total order $\leq$ on $T(\mathcal{F})$ which satisfies $\text{height}(t) < \text{height}(s) \Rightarrow t < s$.

Contexts are elements c of $C_n(\mathcal{F}) \subset T(\mathcal{F}, \mathcal{X})$ where n distinct variables $\$_1, \ldots \$_n$ appears exactly once in c. Let c be a context in $C_n(\mathcal{F})$ and $t_1, \ldots, t_n$ be trees. In the following, the notation $c[\$_1 \leftarrow t_1, \ldots, \$_n \leftarrow t_n]$ or simply $c[t_1, \ldots, t_m]$ represents the tree that results from substituting the $\$_i$'s by the t_i's in c. $C_1(\mathcal{F})$ is simply denoted by $C(\mathcal{F})$. We say that a set A is *prefixial* whenever for any $c \in C(\mathcal{F})$ and any $t \in T(\mathcal{F})$, $c[t] \in A \Rightarrow t \in A$.

Formal Power Tree Series. A *(formal power) tree series* on $T(\mathcal{F})$ is a mapping $r : T(\mathcal{F}) \to \mathbb{R}$. The vector space of all tree series on $T(\mathcal{F})$ is denoted by $\mathbb{K}\langle\langle \mathcal{F} \rangle\rangle$.

Let V be a finite dimensional vector space over $\mathbb{R}$. We denote by $\mathcal{L}(V^p; V)$ the set of p-linear mappings from V^p to V. Let $\mathcal{L} = \cup_{p \geq 0} \mathcal{L}(V^p; V)$. We denote by V^* the dual space of V, *i.e.* the vector space composed of all the linear forms defined on V.

A linear representation of $T(\mathcal{F})$ is a couple (V, μ), where V is a finite dimensional vector space over $\mathbb{R}$, and where $\mu : \mathcal{F} \to \mathcal{L}$ maps $\mathcal{F}_p$ into $\mathcal{L}(V^p; V)$ for each $p \geq 0$. Thus for each $f \in \mathcal{F}_p,\ \mu(f) : V^p \to V$ is p-linear. Function μ extends uniquely to a morphism $\mu : T(\mathcal{F}) \to V$ by: $\mu(f(t_1, \ldots, t_p)) = \mu(f)(\mu(t_1), \ldots, \mu(t_p))$. Let $V_{T(\mathcal{F})}$ be the vector subspace of V spanned by $\mu(T(\mathcal{F}))$: $(V_{T(\mathcal{F})}, \mu)$ is a linear representation of $T(\mathcal{F})$.

Let r be a tree series over $T(\mathcal{F})$, r is said to be *recognizable* if there exists a triple (V, μ, λ), where (V, μ) is a linear representation of $T(\mathcal{F})$, and $\lambda : V \to \mathbb{R}$ is a linear form, such that $r(t) = \lambda(\mu(t))$ for all t in $T(\mathcal{F})$. The triple (V, μ, λ) is called a *linear representation for* r.

It has been shown in [6] that the notions of recognizable tree series and rational tree series (*i.e.* tree series characterized by rational tree expressions) coincide. From now on, we shall refer to them by using the term of *rational* tree series.

Definition 1. *A* stochastic tree language *over* $T(\mathcal{F})$ *is a tree series* $r \in \mathbb{R}\langle\langle \mathcal{F} \rangle\rangle$ *such that for any* $t \in T(\mathcal{F})$, $0 \leq r(t) \leq 1$ *and* $\sum_{t \in T(\mathcal{F})} r(t) = 1$. *Let* p *be a stochastic language, let* $c \in C(\mathcal{F})$ *be such that there exists a tree* t *such that* $p(c[t]) \neq 0$. *We define the stochastic language* $c^{-1}p$ *by* $c^{-1}p(t) = \frac{p(c[t])}{\sum_{t' \in T(\mathcal{F})} p(c[t'])}$.

A *rational stochastic tree language* (RSTL) is a stochastic tree language which admits a linear representation. The set of rational stochastic tree languages is denoted by $\mathcal{S}^{rat}(\mathcal{F})$.

Weighted Tree Automata. A *weighted tree automaton*[1](WTA) over $\mathcal{F}$ is a tuple $\mathcal{A} = (Q, \mathcal{F}, \tau, \delta)$ where Q is a set of states, τ is a mapping from Q to $\mathbb{R}$ and δ is a mapping from $\cup_{m\geq 0}\mathcal{F}_m \times Q^m \times Q$ to $\mathbb{R}$. The mapping δ can be interpreted as a set Δ of rules which can be written in a bottom-up or a top-down way:

$f(q_1, \ldots, q_m) \stackrel{w}{\rightarrow} q \in \Delta$(or $q \stackrel{w}{\rightarrow} f(q_1, \ldots, q_m) \in \Delta$) iff $\delta(f, q_1, \ldots, q_m, q) = w \wedge w \neq 0$.

The weight w of a rule r is denoted by $w(r)$. For any $q \in Q$, we denote by Δ_q the subset of δ composed of the (top-down) rules whose lhs is q and by $\Delta_{f,q}$ the subset of Δ_q composed of rules containing the symbol $f \in \mathcal{F}$ in the rhs. A series r_q can be associated with any state q by: $r_q(f(t_1, \ldots, t_n)) = \sum_{r\in\Delta_q} w(r) \prod_{i=1}^{n} r_{q_i}(t_i)$. Then the WTA $\mathcal{A}$ computes the series r defined by: $r(t) = \sum_{q\in Q} \tau(q) r_q(t)$.

WTA and linear representations are two equivalent ways to represent rational series. For example, let (V, μ, λ) be a linear representation of the tree series $r \in \mathbb{R}\langle\langle\mathcal{F}\rangle\rangle$ and let $B = \{e_1, \ldots, e_n\}$ be a basis of V. A WTA $\mathcal{A} = (Q, \mathcal{F}, \lambda, \delta)$ can be associated with (V, μ, λ, B) where $Q = \{e_1, \ldots, e_n\}$, and $\delta(f, e_{i_1}, \ldots, e_{i_m}, e_j) = w_j$ for any $f \in \mathcal{F}_m$ where $\mu(f)(e_{i_1}, \ldots, e_{i_m}) = \sum_j w_j e_j$. Conversely, an equivalent linear representation can be associated with any weighted tree automaton (see Example 1 below).

Note here that a *probabilistic tree automaton* (PTA) is a specific case of WTA $\mathcal{A} = (Q, \mathcal{F}, \tau, \delta)$ satisfying the following conditions: (i) δ and τ take their values in $[0, 1]$, (ii) $\sum_{q\in Q} \tau(q) = 1$, (iii) for any $q \in Q$, $\sum_{r\in\Delta_q} w(r) = 1$.

It can be shown that any PTA computes a rational tree series r that satisfies $r(t) \geq 0$ for any tree t and $\sum_t r(t) \leq 1$.

It can be shown that there exist rational stochastic tree languages that cannot be computed by any probabilistic automaton (see [7] for an example in the case of word stochastic languages).

Example 1. A WTA *representing a rational stochastic tree language.* Let $\mathcal{A} = (Q, \mathcal{F}, \tau, \delta)$ be the WTA defined by $Q = \{q_1, q_2\}$, $\mathcal{F} = \{a, f(\cdot, \cdot)\}$, $\tau(q_1) = 2, \tau(q_2) = -1$ and $\Delta = \{q_1 \stackrel{2/3}{\rightarrow} a, q_1 \stackrel{1/3}{\rightarrow} f(q_1, q_1), q_2 \stackrel{3/4}{\rightarrow} a, q_2 \stackrel{1/4}{\rightarrow} f(q_2, q_2)\}$.

p_{q_1} and p_{q_2} are RSTL, that the series $p = 2p_{q_1} - p_{q_2}$ computed by $\mathcal{A}$ takes only positive values. And since $\sum_t p(t) = 1$, p is an RSTL. It admits the following linear representation: $(\mathbb{R}^2, \mu, \lambda)$ where $e_1 = (1, 0)$ and $e_2 = (0, 1)$ is a basis of $\mathbb{R}^2$, $\lambda(e_1) = 2, \lambda(e_2) = -1$, $\mu(a) = 2e_1/3 + 3e_2/4$, $\mu(f)(e_1, e_1) = e_1/3, \mu(f)(e_2, e_2) = e_2/4$ and $\mu(f)(e_i, e_j) = 0$ if $i \neq j$.

2.1 Canonical Linear Representation of Rational Tree Series

We now define the canonical representation of a rational tree series.
Let $c \in C(\mathcal{F})$. We define the linear mapping $\dot{c} : \mathbb{R}\langle\langle\mathcal{F}\rangle\rangle \rightarrow \mathbb{R}\langle\langle\mathcal{F}\rangle\rangle$ by

$$\dot{c}(r)(t) = r(c[t]) \ .$$

[1] These automata are also referred to as *multiplicity tree automata* in the literature.

Let $r \in \mathbb{R}\langle\langle \mathcal{F} \rangle\rangle$. Let us denote by W_r the vector subspace of $\mathbb{R}\langle\langle \mathcal{F} \rangle\rangle$ spanned by $\{\dot{c}r | c \in C(\mathcal{F})\}$. It can be shown that r is rational if and only if the dimension of W_r is finite [1]. Let W_r^* be the dual space of W_r, i.e. the set of all linear forms on W_r. For any $t \in T(\mathcal{F})$, let $\overline{t} \in W_r^*$ be defined by: $\forall s \in W_r,\ \overline{t}(s) = s(t)$. It can be shown that there exist trees $t_1, \ldots, t_n$ such that $\{\overline{t_1}, \ldots, \overline{t_n}\}$ forms a basis of W_r^*. Let us define the linear representation (W_r^*, μ, λ) as follows:

- for any $f \in \mathcal{F}_m$, define $\mu(f)(\overline{t_{i_1}}, \ldots, \overline{t_{i_m}}) = \overline{f(t_{i_1}, \ldots, t_{i_m})}$.
- $\lambda \in (W_r^*)^* = W_r$ by $\lambda(\overline{t}) = r(t)$.

Theorem 1. *[1] (W_r^*, ν, τ) is a linear representation of r which is called the* canonical linear representation *of r. It can be embedded in any linear representation of r; in particular, its dimension is minimal.*

Example 2. Consider the rational stochastic tree language p defined in Example 1. It can easily be shown that $p_1(t) = \frac{2^{|t|_f+1}}{3^{2|t|_f+1}}$, $p_2(t) = \frac{3^{|t|_f+1}}{4^{2|t|_f+1}}$ and $p(t) = \frac{2^{5|t|_f+4} - 3^{3|t|_f+2}}{3^{2|t|_f+1} \times 4^{2|t|_f+1}}$. Thus, for any context c and any tree t:

$$\overline{t}(\dot{c}p) = p(c[t]) = \frac{2^{5|t|_f+5|c|_f+4} - 3^{3|t|_f+3|c|_f+2}}{3^{2|t|_f+2|c|_f+1} \times 4^{2|t|_f+2|c|_f+1}}.$$

Since p has a 2-dimensional linear representation, the dimension of W_r^* is ≤ 2. Let $c_0 = \$$ and $c_1 = f(a, \$)$, we have:

$$\overline{a}(\dot{c}_0 p) = \frac{7}{3 \times 2^2}, \overline{a}(\dot{c}_1 p) = \overline{f(a,a)}(c_0) = \frac{269}{3^3 \times 2^6}, \text{ and } \overline{f(a,a)}(\dot{c}_1 p) = \frac{9823}{3^5 \times 2^{10}}.$$

Since $\overline{a}(\dot{c}_0 p) \times \overline{f(a,a)}(\dot{c}_1 p) \neq \overline{a}(\dot{c}_1 p) \times \overline{f(a,a)}(\dot{c}_0 p)$, $\overline{a}$ and $\overline{f(a,a)}$ are linearly independent. Then, $\{\overline{a}, \overline{f(a,a)}\}$ is a basis of W_r^*. We define λ and μ by:

$$\lambda(\overline{a}) = p(a) = \tfrac{7}{3 \times 2^2} \text{ and } \lambda(\overline{f(a,a)}) = p(f(a,a)) = \tfrac{269}{3^3 \times 2^6} \text{ and}$$
$$\mu(a) = \overline{a},\ \mu(f)(\overline{a}, \overline{a}) = \overline{f(a,a)},$$
$$\mu(f)(\overline{a}, \overline{f(a,a)}) = \mu(f)(\overline{f(a,a)}, \overline{a}) = \tfrac{-54}{2^4 \times 3^4}\overline{a} + \tfrac{59}{2^4 \times 3^2}\overline{f(a,a)},$$
$$\mu(f)(\overline{f(a,a)}, \overline{f(a,a)}) = \tfrac{-3186}{2^8 \times 3^6}\overline{a} + \tfrac{2617}{2^8 \times 3^4}\overline{f(a,a)}.$$

We can justify here why the canonical form of a stochastic language p may not be relevant for generating trees according to p. Indeed, one can remark here that $\mu(f(a, f(a,a))) = \mu(f)(\overline{a}, \overline{f(a,a)}) = \frac{-54}{2^4 \times 3^4}\overline{a} + \frac{59}{2^4 \times 3^2}\overline{f(a,a)}$. Thus, if we consider the weights of trees according to $\overline{a}$, $f(a, f(a,a))$ has a negative weight and then $\overline{a}$ does not define by itself a stochastic language. As a consequence, the canonical form does not have a relevant structure if one aims at using it according to a generative model.

2.2 DEES

DEES is an inference algorithm which identifies any rational stochastic language in the limit with probability one (see [1]). Let us show how DEES works on the

previous example. Let S be a sample of trees independently drawn according to p and let p_S be the empirical distribution defined on $T(\mathcal{F})$: $p_S(t)$ is the frequency of t in S. For any confidence parameter δ, there exists $\epsilon > 0$ such that with probability at least $1-\delta$, $|p(t)-p_S(t)| \leq \epsilon$ for any tree t. Statistical tests, based on this property, are used to accept or reject hypotheses of the form: $\overline{t}$ is a linear combination of $\overline{t_1}, \ldots, \overline{t_n}$. Parameters ϵ and δ can be chosen, depending on the size of the sample S, such that with probability one, the correct hypothesis will always be chosen from some sample size.

In order to find the basis of the canonical representation, the algorithm first tests whether $\overline{a}$ and $\overline{f(a,a)}$ are linearly independent. With probability one, this will be detected from some step: $\overline{a}$ and $\overline{f(a,a)}$ are elements of the canonical basis. Then, the algorithm tests whether $\overline{f(a,f(a,a))}$ is a linear combination of $\overline{a}$ and $\overline{f(a,a)}$. As this is true, this will be detected with probability one from some step. Therefore, $f(a,f(a,a))$ will not be added to the basis. And so on. The algorithm terminates when it has checked that no more elements can be added to the basis.

It can be proved that with probability one, there exists an integer N such that for any sample S containing more than N examples, a basis of W_p^* will be identified from S. DEES will compute a linear representation $(W_p^*, \mu_S, \lambda_S)$, such that μ_S and λ_S converge respectively to μ and λ when S tends to infinity.

Hence, DEES identifies in the limit the canonical linear representation of any rational tree stochastic language with probability one. However:

- Given the canonical linear representation of a stochastic language p does not help to generate trees according to p.
- Moreover, the series output by DEES from some sample S can be not a stochastic language. The possibility to transform it in a stochastic language is then an important issue.
- The series output by DEES converges to the target p as the size of S increases, but what is the rate of convergence?

We propose to address of all of these questions in the present paper.

3 Normalised Linear Representation for Rational Stochastic Tree Languages

3.1 Normalised Representation

Let r be a rational stochastic tree language represented by a WTA $(Q, \mathcal{F}, \tau, \delta)$. Tree series r_q associated with each state in Q may not be stochastic tree languages. This is illustrated by Example 2. Trivially, the same remark can be made for the equivalent linear representation of tree series, considering the series associated with every basis vector. However, as stated by the following theorem, there exist equivalent representations, called normalised, where each tree series associated with basis vectors are indeed stochastic tree languages.

Let δ_{ij} be Kronecker symbols, $\delta_{ij} = 1$ if $i = j$ and 0 otherwise.

Theorem 2. *Let p be an* RSTL *over $T(\mathcal{F})$ and let (W_p^*, μ, λ) be the canonical linear representation of p. Then, W_p^* admits a basis $B = \{e_1, \ldots, e_n\}$ such that each series p_i defined by (V, μ, λ_i) where $\lambda_i(e_j) = \delta_{ij}$ is stochastic.*

Proof. Let $c_1, \ldots, c_n \in C(\mathcal{F})$ such that $\{c_1^{-1}p, \ldots, c_n^{-1}p\}$ is a basis of W_p. Let $\{\ell_1, \ldots, \ell_n\}$ be a basis of W_p^* such that $\ell_i(c_j^{-1}p) = \delta_{ij}$ for $1 \leq i, j \leq n$.
We show below that $\{\ell_1, \ldots, \ell_n\}$ is a normalised basis of W^*. Let $\lambda_1, \ldots, \lambda_n$ be the linear forms defined on W^* by $\lambda_i(\ell_j) = \delta_{ij}$. Let us show that for any $t \in T(\mathcal{F})$ and any $1 \leq i \leq n$, $\lambda_i(\mu(t)) = c_i^{-1}p(t)$.

Let $\{\overline{t}_1, \ldots, \overline{t}_n\}$ be a basis of W_p^* and let $\overline{t_i} = \sum_{j=1}^{n} \gamma_i^j \ell_j$ for any $1 \leq i \leq n$. We have:

$$\overline{t_i}(c_j^{-1}p) = \sum_{k=1}^{n} \gamma_i^k \ell_k(c_j^{-1}p) = \sum_{k=1}^{n} \gamma_i^k \delta_{kj} = \gamma_i^j. \tag{1}$$

Let $t \in T(\mathcal{F})$ and $\overline{t} = \sum_{i=1}^{n} \beta_i \overline{t_i}$, then:

$$\overline{t} = \sum_{i=1}^{n} \left(\beta_i \sum_{j=1}^{n} \gamma_i^j \ell_j \right) = \sum_{j=1}^{n} \left(\sum_{i=1}^{n} \beta_i \gamma_i^j \right) \ell_j \ . \tag{2}$$

Because (W_p^*, μ, λ) is the canonical representation of p, we have $\mu(t) = \overline{t}$ by definition. Hence,

$$\begin{aligned} \lambda_j(\mu(t)) = \lambda_j(\overline{t}) &\overset{(2)}{=} \sum_{i=1}^{n} \beta_i \gamma_i^j \overset{(1)}{=} \sum_{i=1}^{n} \beta_i \overline{t_i}(c_j^{-1}p) \\ &= \overline{t}(c_j^{-1}p) && \text{since } \overline{t} = \sum_{i=1}^{n} \beta_i \overline{t_i} \\ &= c_j^{-1}p(t). \end{aligned}$$

Hence, (W_p^*, μ, λ_i) represents a stochastic language for $1 \leq i \leq n$. □

Definition 2. *Let $\mathcal{A} = (Q, \mathcal{F}, \tau, \delta)$ be a* WTA. *We say that $\mathcal{A}$ is in* normalised *form if and only if (i) $\sum_{q \in Q} \tau(q) = 1$, (ii) for any $q \in Q$, $\sum_{r \in \Delta_q} w(r) = 1$ and (iii) for any $q \in Q$ and any $f \in \mathcal{F}$, $\sum_{r \in \Delta_q} w(r) \in [0, 1]$. Moreover, we say that $\mathcal{A}$ is in* reduced *normalised form if the series r_q are linearly independent.*

Any rational stochastic tree language can be represented by a normalised reduced WTA $\mathcal{A} = (Q, \mathcal{F}, \tau, \delta)$ such that each r_q defines a stochastic language. Note also that any PTA is in normalised form (but not necessarily in reduced normalised form).

Example 3. Let us consider the rational stochastic tree language p presented in the previous examples, we show how to compute a normalised WTA that computes it. Let $c_0 = \$$, $c_1 = f(\$, a)$ and let $s_0 = \alpha_0\overline{a} + \beta_0\overline{f(a, a)}$ and $s_1 = \alpha_1\overline{a} +$

$\beta_1\overline{f(a,a)}$ where $s_i(c_j^{-1}p) = \delta_{ij}$. Remarking that $\sum_t \dot{c_0}p(t) = 1$ and $\sum_t \dot{c_1}p(t) = \sum_t p(f(t,a)) = 2\sum_t p_1(f(t,a)) - \sum_t p_2(f(t,a)) = 37/144$ one can check that $\alpha_0 = \frac{-9823}{300}, \alpha_1 = \frac{3228}{25}, \beta_0 = \frac{9953}{300}$ and $\beta_1 = \frac{-3108}{25}$.

Now, by expressing, $\overline{a}$ and $\overline{f(a,a)}$ in the basis s_0, s_1, we get the following set of rules:

$$
\begin{array}{llll}
s_0 \xrightarrow{7/12} a, & & s_1 \xrightarrow{269/444} a, & \\
s_0 \xrightarrow{-269/50} f(s_0,s_0), & & s_1 \xrightarrow{-3024/925} f(s_0,s_0), & \\
s_0 \xrightarrow{259/50} f(s_0,s_1), & & s_1 \xrightarrow{2664/925} f(s_0,s_1), & \\
s_0 \xrightarrow{259/50} f(s_1,s_0), & & s_1 \xrightarrow{2664/925} f(s_1,s_0), & \\
s_0 \xrightarrow{-1369/300} f(s_1,s_1), & & s_1 \xrightarrow{-23273/11100} f(s_1,s_1). &
\end{array}
$$

Let $\lambda(s_0) = 1$ and $\lambda(s_1) = 0$. It is easy to verify that this representation is in normalised form.

3.2 A Generation Process

A generation process of trees can be done using normalized WTA as given in Algorithm 1. Each tree is built top-down. The process is different from the classical approach with PTA since instead of drawing a transition rule to apply at each step, we rather draw a symbol according to the distributions of the symbols defined by the rules.

Comments of the steps numbered by **(1)**, **(2)**, **(3)** and **(4)** in Algorithm 1:

(1) Δ_{gen} contains n rules of the form $q_t \xrightarrow{w_i} c[q_1^i, \ldots, q_m^i]$ where c is a linear context over m variables and where $1 \leq i \leq n$.
(2) It can be proved that $\sum_{f\in\mathcal{F}} \alpha_{f,j}^c = 1$ for any $1 \leq j \leq m$.
(3) The numbers $\alpha_{f,j}^c$ define a probability distribution over $\mathcal{F}_j$.
(4) There exists a unique tree t such that all the rules of Δ_{gen} are of the form $q_t \xrightarrow{w_i} t$; t is the output of the algorithm.

4 Learning Rational Stochastic Tree Languages

We consider the question of learning a rational stochastic tree language (RSTL) p from an i.i.d. sample of trees drawn according to p. An RSTL can be such that the average size of trees generated from p is unbounded, i.e. $\sum_t p(t)|t| = \infty$. For example, this is the case for the RSTL defined by the PTA whose rules are: $\{q \xrightarrow{1/2} a, q \xrightarrow{1/2} f(q,q)\}$. To our knowledge, it is still unknown whether a PTA defines a RSTL and it is much better to deal with the stronger notion of *strongly consistent* stochastic language: A RSTL p is strongly consistent if $\sum_t |t|p(t) < \infty$. Next section investigates some properties of strongly consistent RSTL.

Data : An WTA $\mathcal{A} = (Q, \mathcal{F}, \tau, \delta)$ in normalised form
Result : A tree $t \in T(\mathcal{F})$
begin
  Let q_t be a new state ;
  Let $\Delta_{gen} = \{q_t \overset{\tau(q)}{\to} q | q \in Q\}$ **(1)**;
  while *the rhs of some rule of* Δ_{gen} *contains states* **do**
    Let m be the number of rules in Δ_{gen} and n be the number of states in each the rules;
    for $1 \leq j \leq m$ **do**
      for any $f \in \mathcal{F}$, let $\alpha^c_{f,j} = \sum_{i=1}^{n} w_i \sum_{r \in \Delta_{q^i_j, f}} w(r)$ **(2)**;
      draw randomly $f_j \in \mathcal{F}$ according to $\alpha^c_{f,j}$ **(3)**;
    let n_j be the rank of f_j;
    let $c' = c(f_1(\$^1_1, \ldots, \$^{n_1}_1), \ldots, f_m(\$^1_m, \ldots, \$^{n_m}_m))$ a linear context;
    in Δ_{gen}, replace each rule $q_t \overset{w_i}{\to} c[q^i_1, \ldots, q^i_m]$ by the rules
    $q_t \overset{w_i w_{r_1} \ldots w_{r_m}}{\longrightarrow} c[f_1(q^1_{r_1}, \ldots, q^{n_1}_{r_1}), \ldots, f_m(q^1_{r_m}, \ldots, q^{n_m}_{r_m})]$;
    where $r_j : q_j \overset{w_{r_j}}{\to} f_j(q^1_{r_j}, \ldots, q^{n_1}_{r_j}) \in \Delta_{q^i_j, f_j}$, $1 \leq j \leq m$, $1 \leq i \leq n$;
  Outputs the tree of Δ_{gen} **(4)**;
end

Algorithm 1. Drawing a tree according to a RSTL

4.1 Strongly Consistent Rational Stochastic Languages

Let $\mathcal{A} = (Q = \{q_1, \ldots, q_n\}, \mathcal{F}, \tau, \delta)$ be a WTA and let $A = (a_{ij})_{1 \leq i,j \leq n}$ be the matrix defined by $a_{ij} = \sum_{r \in \Delta_{q_i}} n_r(j) w(r)$ where $n_r(j)$ is the number of occurrences of q_j in the rhs of r.

We denote by p_i the rational series defined from state q_i and we let $\gamma_i = \sum_{t \in T(\mathcal{F})} p_i(t)|t|$ (γ_i may be undefined if the sum diverges), $\gamma = (\gamma_1, \ldots, \gamma_n)$ and $B = (1, \ldots, 1)^t$.

Proposition 1. *Let us suppose that for any index* i, $\sum_{t \in T(\mathcal{F})} p_i(t) = 1$ *and* $\sum_{r \in \Delta_{q_i}} w(r) = 1$. *Then* $\gamma = \sum_{n \geq 0} A^n B$.

Proof. The proof is detailed in [8].

The sum $\sum_{n \geq 0} A^n B$ converges iff $A^n B$ converges to 0, which can be decided within polynomial time.

Example 4. Consider the PTA defined by the rules $\{q \overset{1-\alpha}{\to} a, q \overset{\alpha}{\to} f(q,q)\}$ and $\tau(q) = 1$: $A = (2\alpha)$ and $A^n B$ converges iff $\alpha < 1/2$. The average size of trees generated from these PTA is $1/(1-2\alpha)$. When $\alpha = 1/3$ (resp. $1/4$), the PTA

computes the stochastic language p_{q_1} (resp. p_{q_2}) as previously defined in example 1. Then, the average size of trees γ_1 (resp. γ_2) generated from p_{q_1} (resp. p_{q_2}) is 3 (resp. 2). One can deduce the average size of the stochastic language $p = 2p_{q_1} - p_{q_2}$, $\gamma = 2 \times \gamma_1 - \gamma_2 = 4$.
Consider now the normalised form of p as presented in example 3.
The matrix A is $\begin{pmatrix} -2/5 & 37/30 \\ -144/185 & 47/30 \end{pmatrix}$.
It is easy to verify that $(I-A)$ is invertible and $(I-A)^{-1} = \begin{pmatrix} -17/5 & 37/5 \\ -864/185 & 42/5 \end{pmatrix}$.
Thus $(I-A)^{-1}B = \begin{pmatrix} 4 & 690/185 \end{pmatrix}$. Following Prop. 1, the average size γ_0 of trees generated by $c_0^{-1}p$ is 4 and the average size of trees generated by $c_0^{-1}p$ is 690/185. Since $p = c_0^{-1}p$ the average tree size of p is 4.

We show below that when $\mathcal{A}$ is a reduced normalised representation of a strongly consistent rational stochastic language, the spectral radius[2] $\rho(A)$ of A is < 1. We need the following lemma :

Lemma 1. *Let $p_1, \ldots, p_n$ be n independent stochastic languages. Then $\Lambda = \{(\alpha_1, \ldots, \alpha_n) \in \mathbb{R}^n : \sum_{i=1}^{n} \alpha_i p_i$ is a stochastic language$\}$ is a compact convex subset of $\mathbb{R}^n$.*

Proof. See [9] for a similar proof in the case of words.

Proposition 2. *Let $\mathcal{A} = (Q = \{q_1, \ldots, q_n\}, \mathcal{F}, \tau, \delta)$, a reduced normalised representation of a strongly consistent* RSTL *p such that each p_{q_i} is a stochastic language and let $A = (a_{ij})_{1 \leq i,j \leq n}$ be the matrix defined as previously. Then the spectral radius of A satisfies $\rho(A) < 1$.*

Proof. The proof is detailed in [8].

Example 5. The matrix A of Example 4 admits two eigenvalues: $\frac{1}{2}$ and $\frac{2}{3}$, then $\rho(A) = \frac{2}{3} < 1$.

4.2 Effective Normalisation

Let p be a strongly consistent RSTL and let $B = \{\overline{t_1}, \ldots, \overline{t_n}\}$ be the smallest (for the order $\leq$ on $T(\mathcal{F})$) basis of the canonical linear representation (W_p^*, μ, λ) of p. The main result in [1] proves that with probability one, there exists a sample size from which DEES outputs a linear representation $(W_p^*, \mu_S, \lambda_S)$ whose basis is B and such that μ_S and λ_S are arbitrarily close to μ and λ.

Theorem 2 states that there exists a normalised WTA $\mathcal{A}_S$ given its canonical linear representation (W_p^*, μ, λ). In this section we explain how to effectively compute $\mathcal{A}_S$. Choosing a basis written as $\{\dot{c}_1 p, \ldots, \dot{c}_n p\}$ is easily done by recursively enumerating every context, the main technical key point relies in the ability to compute the sums $\sum_{t \in T(\mathcal{F})} p(c_i[t])$ for a given rational series.

[2] The spectral radius of a matrix is the maximum of the norms of its complex eigenvalues.

Let s be the vector defined by $s = \sum_{t\in T(\mathcal{F})} \mu(t) = \sum_{t\in T(\mathcal{F})} \overline{t}$. The ith component of s is $\sum_{t\in T(\mathcal{F})} p_i(t) = \sum_{t\in T(\mathcal{F})} p(c_i[t])$. Moreover, s is a solution of the polynomial system: $v = F(v)$ where $F(v) = \sum_m \sum_{f\in\mathcal{F}_m} \mu(f)(v,\ldots,v)$. This system is not analytically soluble in general. As a consequence, we approximate s using with a direct propagative method.

Let E and E_k be the endomorphisms defined by:

$$E(v) = \sum_m \sum_{l=1}^{m} \sum_{f\in\mathcal{F}_m} \mu(f)(\underbrace{s,\ldots,s}_{l-1}, v, \underbrace{s,\ldots,s}_{m-l})$$

$$E_k(v) = \sum_m \sum_{l=1}^{m} \sum_{f\in\mathcal{F}_m} \mu(f)(\underbrace{s_k,\ldots,s_k}_{l-1}, v, \underbrace{s,\ldots,s}_{m-l}).$$

A propagative method is proposed by Stolcke[10] in the case of probabilistic context-free languages. Let $T^{<k}(\mathcal{F})$ be the set of trees of height lower than k. The idea is to recursively compute the sequence $s_k = \sum_{t\in T^{<k}(\mathcal{F})} \overline{t}$ using the recursion: $s_0 = 0$ and $s_{k+1} = F(s_k)$. Obviously, (s_k) converges towards s. Let us study the convergence rate.

By applying the multi-linearity of $\mu(f)$, $s - s_{k+1}$ can be decomposed in $s - s_{k+1} = F(s) - F(s_k) = E_k(s - s_k)$. Taking into account that for every tree t, the ith component of $\overline{t}$ is $p(c_i[t]) \geq 0$, it is easily shown that for every k:

$$\|s - s_{k+1}\| = \|\prod_{q=0}^{k} E_q(s - s_0)\| \leq \|E^k\|\|(s - s_0)\| \ .$$

By Gerland's formula, we have $\|E^k\| \sim \rho(E)^k$ and thus:

$$\|s - s_k\| = O(\rho(E)^k \|s - s_0\|) \ .$$

Let A be the matrix of E in the basis $\{c_1^{-1}p,\ldots,c_n^{-1}p\}$. It can be proved that A is the same matrix as defined in Section 4.1. Thanks to Proposition 2 and because we made the assumption the series is strongly consistent, we know that $\rho(E) = \rho(A) < 1$.

When tested on the previous example, the propagative method achieved precision of 10^6 in approximately 30 iterations. In near future, we intend to study the use of Newton's method, which could at least theoretically achieve faster convergence.

4.3 Learning a Strongly Consistent Rational Stochastic Language: The Road Map

The normalised WTA $\mathcal{A}_S$ obtained at the end of the previous section computes an RSTL p_S such that the spectral radius ρ_S of the matrix A_S associated with $\mathcal{A}_S$ satisfies $\rho_S < 1$ which is a strong property. We have still some results to prove in order to complete the learning process. We present them below as conjectures.

Conjecture 1: It is possible to modify Algorithm 1 in order to be used to generate trees from a normalised WTA (even when it does not define a stochastic language). The modified algorithm stops (and outputs a tree) with probability one, as soon as S is sufficiently large. Hence, it defines a stochastic language $\hat{p}$.

Conjecture 2: with probability one, $\sum_t |p(t) - \hat{p}(t)| \cdot |t|$ converges to 0 with the size of S.

These two conjectures generalize results proved in the word case. Note that the convergence type described in Conjecture 2 is stronger than L_1-convergence.

5 Unranked Trees

In this section we consider trees where the rank constraint has been dropped: Every symbol in unranked trees may have from 0 to an unbounded but finite number of (ordered) children. Unranked trees are the common abstract representation of semi-structured data like XML.

Let Σ be a finite set of symbols. The set $T(\Sigma)$ of unranked trees is the smallest set such that $\Sigma \subseteq T(\Sigma)$, and $f(t_1, \ldots, t_m) \in T(\Sigma)$ provided $f \in \Sigma$ and $t_1, \ldots, t_m \in T(\Sigma)$. An algebraic definition of unranked trees can be given by means of the extension operator @ ([4]). Basically, @ adds a new child at the end of the list of children of an unranked tree: $f \,@\, t = f(t)$, $f(t_1, \ldots, t_{n-1}) \,@\, t_n = f(t_1, \ldots, t_n)$.

The extension operator provides a unique recursive definition of any unranked tree. It can be syntactically represented by a binary (ranked) tree over $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_2$ where $\mathcal{F}_0 = \Sigma$ and $\mathcal{F}_2 = \{@\}$. Let us now define the mapping ext from $T(\Sigma)$ to $T(\mathcal{F})$ by $\mathsf{ext}(f) = f$ and $\mathsf{ext}(f(t_1, \ldots, t_n)) = @(\mathsf{ext}(f(t_1, \ldots, t_{n-1})), \mathsf{ext}(t_n))$. One can show that the mapping ext is a bijection. Hedge automata [11] directly act on unranked trees in $T(\Sigma)$. Briefly, hedge automata rules are of the form $f(L) \to q$ where L is a word language on the alphabet of states. It has be shown that hedge automata and ordinary tree automata on $T(\mathcal{F})$ define the same class of recognizable languages [12]. Extension from hedge automata to weighted hedge automata (there referred to as unranked WTA) is proposed in [13]. In unranked WTA rules are of the form $f(L) \xrightarrow{w} q$ where L is a weighted word language on the alphabet of states.

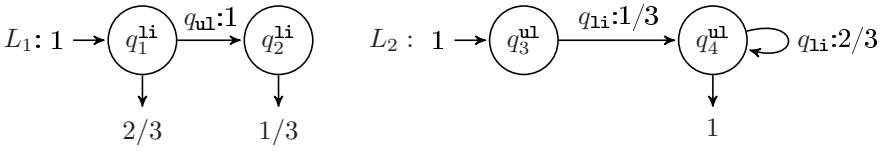
Thanks to the ext mapping, each result presented in this paper can be interpreted in the case of unranked trees. Tree series on $T(\Sigma)$ are simply defined via tree series on $T(\mathcal{F})$. This mapping also suggests a notion of rational unranked tree series and stochastic languages.

Proposition 3. *The class of rational unranked tree series represented via the mapping* ext *coincide with the class of unranked tree series defined by unranked* WTA.

More precisely, let be an unranked WTA which represents a rational unranked tree series r^u. One can build in linear time a (ranked) WTA which represents a

rational tree series r^r such that $\forall t \in T(\Sigma)\ r^u(t) = r^r(\mathtt{ext}(t))$. The converse is also true but to compute the corresponding unranked WTA, one needs to normalise rules following the method given in Section 4.2.

The following example illustrates how one can build a weighted automaton for unranked trees. Let us consider trees that represent nested lists built with the commonly used symbols `ul` and `li`. Let us consider first a stochastic hedge automaton with two states $q_{\mathtt{ul}}$ and $q_{\mathtt{li}}$. Final weights are given by $F(q_{\mathtt{ul}}) = 1$ and $F(q_{\mathtt{li}}) = 0$. Rules are $\mathtt{li}(L_1) \xrightarrow{1} q_{\mathtt{li}}$ and $\mathtt{ul}(L_2) \xrightarrow{1} q_{\mathtt{ul}}$ where



The weight of a tree $\mathtt{ul}(\mathtt{li}, \mathtt{li}(\mathtt{ul}(\mathtt{li})))$ is $2^3/3^6$.

The corresponding automaton on the expression with the @ operator has 4 states $\{q_1^{\mathtt{li}}, q_2^{\mathtt{li}}, q_3^{\mathtt{ul}}, q_4^{\mathtt{ul}}\}$, $\tau(q_4^{\mathtt{ul}}) = 1$ and the set of rules:

$$\left\{ \begin{array}{ccc} \mathtt{li} \xrightarrow{1} q_1^{\mathtt{li}} & , & \mathtt{ul} \xrightarrow{1} q_3^{\mathtt{ul}} \quad , \quad @(q_1^{\mathtt{li}}, q_4^{\mathtt{ul}}) \xrightarrow{w_1} q_2^{\mathtt{li}}, \\ @(q_3^{\mathtt{ul}}, q_1^{\mathtt{li}}) \xrightarrow{w_2} q_4^{\mathtt{ul}} & , & @(q_3^{\mathtt{ul}}, q_2^{\mathtt{li}}) \xrightarrow{w_3} q_4^{\mathtt{ul}}, \\ @(q_4^{\mathtt{ul}}, q_1^{\mathtt{li}}) \xrightarrow{w_4} q_4^{\mathtt{ul}} & , & @(q_4^{\mathtt{ul}}, q_2^{\mathtt{li}}) \xrightarrow{w_5} q_4^{\mathtt{ul}} \end{array} \right\}$$

The weight w_2 is the weight of adjoining a subtree in state $q_1^{\mathtt{li}}$ to a tree in state $q_3^{\mathtt{ul}}$. The results gives a tree in state $q_4^{\mathtt{ul}}$. It corresponds to the following computation in the hedge automaton: exit from L_1 with state $q_1^{\mathtt{li}}$, then apply the rule $\mathtt{li}(L_1) \xrightarrow{1} q_{\mathtt{li}}$ and finally follow the transition from $q_3^{\mathtt{ul}}$ to $q_4^{\mathtt{ul}}$ in L_2. Hence $w_2 = 2/3 \times 1 \times 1/3$. Similarly $w_3 = 1/3 \times 1 \times 1/3$, $w_4 = 2/3 \times 1 \times 2/3$, $w_5 = 1/3 \times 1 \times 2/3$ and $w_1 = 1 \times 1 \times 1$. The binary tree associated with $\mathtt{ul}(\mathtt{li}, \mathtt{li}(\mathtt{ul}(\mathtt{li})))$ is $@(@(\mathtt{ul}, \mathtt{li}), @(\mathtt{li}, @(\mathtt{ul}, \mathtt{li})))$. One can verify that its weight is also $2^3/3^6$.

Hence, to learn rational unranked tree series, one can simply proceed in the following way: apply `ext` to input trees and then apply DEES. Eventually, a representation of an unranked WTA where weights are estimated can possibly be returned.

6 Conclusion

In this paper, we studied the problem of learning a rational stochastic tree language p from an *i.i.d.* sample of trees drawn from p. An inference algorithm, DEES, was previously proposed for this problem. Using this algorithm leads to two main drawbacks: It often outputs linear representations that do not define stochastic languages and these representations can not be directly used to generate trees from the underlying distribution. We addressed this problem by showing that any rational stochastic tree language admits a normalised reduced representation that can be used as a generative model. We have studied the

notion of strongly consistent rational stochastic languages which corresponds to the fact that the average size of trees generated from a RSTL p is bounded. We showed the relationship between this notion and the normalised reduced representation of a RSTL. We finally justified that the methods presented in this paper can be directly applied to unranked trees.

The next step of this work is to prove the conjectures that was presented for learning strongly consistent rational stochastic languages: First, a probability distribution $\hat{p}$ can be extracted in order to generate trees from a normalised WTA. Second, that $\sum_t |p(t) - \hat{p}(t)| \cdot |t|$ convergences to zero with the size of the learning sample. Note here that this condition is stronger than the L_1-convergence.

References

1. Denis, F., Habrard, A.: Learning rational stochastic tree languages. In: Hutter, M., Servedio, R.A., Takimoto, E. (eds.) ALT 2007. LNCS (LNAI), vol. 4754, pp. 242–256. Springer, Heidelberg (2007)
2. Booth, T., Thompson, R.: Applying probabilistic measures to abstract languages. IEEE Transactions on Computers 22(5), 442–450 (1973)
3. Wetherell, C.S.: Probabilistic languages: A review and some open questions. ACM Comput. Surv. 12(4), 361–379 (1980)
4. Comon, H., Dauchet, M., Gilleron, R., Jacquemard, F., Lugiez, D., Löding, C., Tison, S., Tommasi, M.: Tree automata techniques and applications (2007) (release October 12, 2007), `http://tata.gforge.inria.fr/`
5. Berstel, J., Reutenauer, C.: Recognizable formal power series on trees. Theoretical Computer Science 18, 115–148 (1982)
6. Ésik, Z., Kuich, W.: Formal tree series. Journal of Automata, Languages and Combinatorics 8(2), 219–285 (2003)
7. Denis, F., Esposito, Y.: Rational stochastic languages. Technical report, LIF - Université de Provence (2006), `http://hal.ccsd.cnrs.fr/ccsd-00019728`
8. Denis, F., Gilbert, E., Habrard, A., Ouardi, F., Tommasi, M.: Relevant representations for the inference of rational stochastic tree languages. Technical report, LIF, LIFL, and INRIA (2008), `http://hal.archives-ouvertes.fr/hal-00293511/en/`
9. Denis, F., Esposito, Y., Habrard, A.: Learning rational stochastic languages. In: Lugosi, G., Simon, H.U. (eds.) Learning theory. LNCS, pp. 274–288. Springer, Heidelberg (2006)
10. Stolcke, A.: An efficient probabilistic context-free parsing algorithm that computes prefix probabilities. Computional Linguistics 21(2), 165–201 (1995)
11. Brüggemann-Klein, A., Murata, M., Wood, D.: Regular tree and regular hedge languages over unranked alphabets. Technical report, Hong Kong University Theoretical Computer Science Center, Version 1 (2001)
12. Carme, J., Niehren, J., Tommasi, M.: Querying unranked trees with stepwise tree automata. In: van Oostrom, V. (ed.) RTA 2004. LNCS, vol. 3091, pp. 105–118. Springer, Heidelberg (2004)
13. Droste, M., Vogler, H.: Weighted logics for XML (manuscript, 2007), `http://www.orchid.inf.tu-dresden.de/gdp/monographs/r20.ps`