Counting occurrences for a finite set of words: combinatorial methods

Frédérique Bassino, Julien Clément, Pierre Nicodème

Counting occurrences for a finite set of words: combinatorial methods

Frédérique BASSINO, LIPN UMR 7030, Université de Paris 13, CNRS, France
Julien CLÉMENT, GREYC UMR 6072, Université de Caen, Ensicaen, CNRS, France
Pierre NICODÈME, LIX, UMR 7161, École polytechnique, Palaiseau, INRIA-Amib Saclay, CNRS, France

In this article, we provide the multivariate generating function counting texts according to their length and to the number of occurrences of words from a finite set. The application of the inclusion-exclusion principle to word counting due to Goulden and Jackson (1979, 1983) is used to derive the result. Unlike some other techniques which suppose that the set of words is *reduced* (*i.e.*, where no two words are factor of one another), the finite set can be chosen arbitrarily. Noonan and Zeilberger (1999) already provided a MAPLE package treating the non-reduced case, without giving an expression of the generating function or a detailed proof. We provide a complete proof validating the use of the inclusion-exclusion principle. Some formulæ for expected values, variance and covariance for number of occurrences when considering two arbitrary sets of finite words are given as an application of our methodology.

Categories and Subject Descriptors: F.2.2. [**Analysis of Algorithms and Problem Complexity**]: Nonnumerical Algorithms and Problems; G.2.1. [**Discrete Mathematics**]: Generating functions, Counting problems

General Terms: Algorithms

Additional Key Words and Phrases: Word Statistics, Inclusion-Exclusion, Generating Functions, Aho-Corasick Automaton

ACM Reference Format:
Bassino, F., Clément, J., Nicodème, P. 2011. Counting occurrences for a finite set of words. ACM Trans. Algor. 9, 4, Article 39 (March 2010), 28 pages.
DOI = 10.1145/0000000.0000000 http://doi.acm.org/10.1145/0000000.0000000

1. INTRODUCTION

Enumerating sequences with given combinatorial properties is rigorously formalized since the end of the seventies and the beginning of the eighties by Goulden and Jackson [Goulden and Jackson 1979; 1983] and by Guibas and Odlyzko [Guibas and Odlyzko 1981a; 1981b].

The former [Goulden and Jackson 1979; 1983] introduce a very powerful method of inclusion-exclusion to count occurrences of words from a *reduced* set of words (*i.e.*, a set where no word is factor of another word of the set) in texts; this method is characterized by counting texts where some occurrences are *marked* (other terms are *pointed* or *anchored*) and then removing multiple counts of the same text (text counted several times with different markings). We refer later to this by *inclusion-exclusion* method.

F. Bassino acknowledges financial support from the French ANR (MAGNUM) under grant ANR-BLAN-2010-0204.
Emails: Frederique.Bassino@lipn.univ-paris13.fr, Julien.Clement@info.unicaen.fr, nicodeme@lix.polytechnique.fr.

DOI 10.1145/0000000.0000000 http://doi.acm.org/10.1145/0000000.0000000

Goulden-Jackson counting is typically multivariate, a formal parameter being associated to each word.

The latter [Guibas and Odlyzko 1981a; 1981b] introduce the notion of autocorrelation of a word that generalizes to correlation between words, this notion being implicit in Goulden and Jackson. Formal non-ambiguous manipulations over languages translate into generating functions; we refer to this later by *formal language* method. Unlike Goulden and Jackson, Guibas and Odlyzko consider univariate cases, like enumerating sequences avoiding a pattern, or sequences terminating with a first occurrence of a pattern in a text (see also [Sedgewick and Flajolet 1996]). Régnier and Szpankowski [Régnier and Szpankowski 1997] extends this further to multivariate analysis and simultaneous counting of several words; following up works of these authors consider a Markovian source on the symbol emission [Régnier and Szpankowski 1998; Régnier 2000]. See also the books of Szpankowski [Szpankowski 2001] and Lothaire [Lothaire 2005]. Bourdon and Vallée [Bourdon and Vallée 2002; 2006] apply the previous analysis to dynamical sources. Prum *et al.* [Prum et al. 1995], Reinert and Schbath [Reinert and Schbath 1998], Reinert *et al.* [Reinert et al. 2000], and Roquain and Schbath [Roquain and Schbath 2007] follow a more probabilistic approach.

Noonan and Zeilberger [Noonan and Zeilberger 1999] extend the inclusion-exclusion method of Goulden and Jackson and solve the general non-reduced case (words may be factor of other words), implementing corresponding MAPLE programs, without however completely publishing the explicit result formulæ. Recently Kong [Kong 2005] applies the results of Noonan and Zeilberger for the reduced case to an asymmetrical Bernoulli (also called memoryless) model for the generation of symbols. He also compares the Goulden and Jackson method to the Régnier and Szpankowski method, emphasizing the conceptual simplicity of the inclusion-exclusion approach. It is however useful to note that the formal language approach provides access to information that the inclusion-exclusion method does not, such as the waiting time for a first match of a word or the time separating two matches of the same word or of two different words (in both cases eventually forbidding matches with other words). There is however no known solutions to the general problem of words counting by the formal language method.

A third approach is possible by use of automata. Nicodème *et al.* [Nicodème et al. 2002] use classical algorithms to (1) build a marked deterministic automaton recognizing a regular expression and (2) translate into generating function by the Chomsky-Schützenberger algorithm [Chomsky and Schützenberger 1963]; this provides the bivariate generating function counting the matches. A variation of the method extends the results to Markovian sources. This result applies immediately to a set of words considered as a regular expression. Nicodème [Nicodème 2003] extends this to multivariate counting by taking the product of marked automata (with an automaton and a mark associated to a word) and to sets of words with possible errors[1]. Notice that, when handling finite languages, step (1) of the automaton approach may be directly done by building the Aho-Corasick automaton, which is specifically designed for pattern-matching.

Each of the three above-mentioned approaches did develop quite independently and partially unaware of each other.

In this article we focus on a fundamental object, called multivariate generating function, which allows a concise mathematical description of occurrences statistics in random texts. More precisely, we describe two approaches to compute the multivariate generating function $F_{\mathcal{U}}$ counting texts according to their length and to their number of

[1]Algorithms implemented in the package `regexpcount` of `algolib`, Algorithms Project, INRIA

occurrences of words from a pattern or set $\mathcal{U} = \{u_1, \ldots, u_r\}$ of r words. The resulting generating function is rational; once computed, it is a simple task to obtain all kind of statistics (see Section 6 for some examples). The reader is also referred to [Flajolet and Sedgewick 2009] for a general background on generating functions.

Historically, research on counting occurrences for finite cases considered separately the so-called "reduced" case, which is easier, and where no word of the pattern is factor of another word of the pattern; in the opposite or "non-reduced" case, there are no conditions on the pattern. We focus on methods which solve the problem in this latter case (as example the pattern $\mathcal{U}$ can contain $u_1 = abbababa$ and $u_2 = baba$ although u_2 is a factor of u_1). Note that in the non-reduced case, the count of matches that we consider here may exceed the count of positions of the texts at which an occurrence terminates; in contrary, in the reduced case, these two counts are identical. We aim at presenting for the general counting problem a novel approach and a full proof of results partially in Noonan and Zeilberger [Noonan and Zeilberger 1999].

This article is organized as follows. We define in Section 2 our notations. In Section 3 we present an approach using the Aho-Corasick automaton that solves the general (non-reduced) problem; we also consider the complexity of this method. We present in Section 4 an intuitive approach to the inclusion-exclusion method that [Goulden and Jackson 1983] applied to reduced sets of words. We describe and prove in Section 5 our results that follow from the analytic inclusion-exclusion principle in the general case of word counting; algorithmic aspects are also considered in this section. As an application of our methodology, Section 6 provides precise formula for some statistics (expectation and variance of any set of finite words, and covariance for number of occurrences when considering two arbitrary sets of finite words).

2. BASIC NOTATIONS

Let $\mathcal{A}$ be the alphabet on which the words are written and $\mathcal{U} = \{u_1, u_2, \ldots, u_r\}$ be a finite set (or pattern) of distinct words on the alphabet $\mathcal{A}$. *By convention, in this article, words in a set $\mathcal{U}$ are always indexed in lexicographic order. We will also consider that in a single word pattern $\mathcal{U} = \{u\}$, the word u has index* 1.

Weights. We denote $\pi(w)$ the weight of the word w. The weight could be a formal weight over the commutative monoid $\mathcal{A}^\star$ (*i.e.*, $\pi(ababab) = \alpha^3\beta^3$), or the probability generating function in the Bernoulli (also called *memoryless*) setting, $\pi(w) = \Pr(w)$ (the probability of w in this model), or even $\pi(w) = 1$ for a uniformly weighted model over all words (enumerative model).

Representing occurrences. This article is focused on statistics of occurrences of words of $\mathcal{U}$ with possible overlaps in texts. A convenient way to represent occurrences, adopted throughout this article, is to associate to a text w of length n a sequence $\mathcal{O} = (\mathcal{O}_i)_{i=1}^n$ called the *occurrence index* of the pattern $\mathcal{U}$ in the text w, defined, for $1 \le i \le |w|$, $\mathcal{O}_i \subset \{1, \ldots, r\}$, where $\mathrm{Card}(\mathcal{U}) = r$, as

$$\mathcal{O}_i = \{j \mid u_j \text{ has an occurrence ending at position } i \text{ of } w\}.$$

For instance let us consider the case of a text $w = aababaabbbabaa$ and a simple pattern formed with one word $u = aba$. Then the occurrence index $\mathcal{O}$ of the pattern $\{u\}$ in w verifies

$$\mathcal{O}_i = \begin{cases} \{1\} & \text{if } i \in \{4, 6, 13\} \\ \emptyset & \text{otherwise} \end{cases}$$

Notice that we have selected (what we refer next as distinguished) all occurrences of u in w. Later on, along our needs, we will select or distinguish only a subset of those occurrences.

Generating functions. For any (possibly infinite) set of words $\mathcal{H}$, we denote $H(z) = \sum_{h\in\mathcal{H}} \pi(h) z^{|h|}$ the univariate generating function of $\mathcal{H}$, where z is a formal variable marking the length of the words. For instance the generating function of the alphabet $\mathcal{A}$ is $A(z) = \sum_{\alpha\in\mathcal{A}} \pi(\alpha) z$.
We consider also multivariate generating functions which take into account statistics of occurrences. When considering a pattern (denoted as a set) $\{u_1, \ldots, u_j, \ldots, u_r\}$, we will typically use the variables t_j and x_j to count the number of occurrences of the word u_j; as we shall see later there will be a need for two variables, although they are in a very simple relation to each other. If the pattern is composed of a single word $\mathcal{U} = \{u\}$, we use the variables t and x. Given a r-row vector $\mathbf{x} = (x_1, \ldots, x_r)$ of formal variables and a r-row vector $\mathbf{j} = (j_1, \ldots, j_r)$ of integers, we will denote by $\mathbf{x}^{\mathbf{j}}$ the product $\prod_{i=1}^{r} x_i^{j_i}$. To any set of words $\mathcal{X}$, we can associate a formal series or generating function that gathers the counts statistics

$$X(z) = \sum_{w\in\mathcal{X}} \pi(w) z^{|w|} \mathbf{x}^{\boldsymbol{\tau}(w)},$$

where $\boldsymbol{\tau}(w) = (|w|_1, \ldots, |w|_r)$, and $|w|_i$ is the total number of occurrences of u_i in w (with possible overlaps). For instance, the generating function of the (composed of a single text) set $\mathcal{X} = \{abaaabaabb\}$ with $\mathcal{U} = \{u_1 = aa, u_2 = baa\}$ is $X(z, x_1, x_2) = z^{10}\pi(a)^6\pi(b)^4 x_1^3 x_2^2$.

In this article we describe how to compute in the most general case the multivariate generating function $F_{\mathcal{U}}(z, \mathbf{x})$ counting texts from $\mathcal{A}^\star$ according to their length and to the number of occurrences (with overlap) of words from a set $\mathcal{U}$,

$$F_{\mathcal{U}}(z, \mathbf{x}) = F(z, \mathbf{x}) := \sum_{w\in\mathcal{A}^\star} \pi(w) z^{|w|} \mathbf{x}^{\boldsymbol{\tau}(w)}. \tag{1}$$

Autocorrelation and correlation of words. We recall here the classical definitions of autocorrelation of one word and of correlation of a word with another word.
The autocorrelation set $\mathcal{C}_u$ of a word u is defined as usual as

$$\mathcal{C}_u = \{h, \quad u \cdot h = y \cdot u \quad \text{with } |y| < |u|\};$$

note that the empty word ε belongs to $\mathcal{C}_u$. We use the notations $C_u(z)$ or $C(z)$ if there is no ambiguity about the word considered for the autocorrelation polynomial of the word u.

We define similarly the correlation set $\mathcal{C}_{u,v}$ from a word u to a word v as

$$\mathcal{C}_{u,v} = \{h; \quad u \cdot h = y \cdot v, \quad |y| < |u|; \} \tag{2}$$

Note that $\mathcal{C}_{u,u}$ is the autocorrelation set of the word u and that if $u \neq v$ the empty word ε does not belong to the set $\mathcal{C}_{u,v}$.

3. AUTOMATON APPROACH

We resort in this section to the well-known Aho-Corasick algorithm [Aho and Corasick 1975; Crochemore and Rytter 2002] which builds from a finite set of words $\mathcal{U}$ a (not necessarily minimal) deterministic complete automaton recognizing the language $\mathcal{A}^\star\mathcal{U}$. This automaton denoted by $\mathcal{A}_{\mathcal{U}}$ is the basis of many efficient algorithms on string matching problems and is often called the *string matching automaton*. It is usually described by the trie built upon the set of input words together with a failure function. Let $\mathcal{T}_{\mathcal{U}}$ be the ordinary trie representing the set $\mathcal{U}$, seen as a finite deterministic automaton $(Q, \delta, \varepsilon, T)$, where the set of states is $Q = \mathrm{Pref}(\mathcal{U})$ (prefixes of words in $\mathcal{U}$), the initial state is ε (denoting ε the empty word), the set of final states is $T = \mathrm{Pref}(\mathcal{U}) \cap \mathcal{A}^\star\mathcal{U}$,

and the transition function δ is defined on $\mathrm{Pref}(\mathcal{U}) \times \mathcal{A}$ by

$$\delta(p,x) = \begin{cases} px & \text{if } px \in \mathrm{Pref}(\mathcal{U}), \\ \mathrm{Border}(px) & \text{otherwise,} \end{cases}$$

where the failure function $\mathrm{Border}()$ is defined by

$$\mathrm{Border}(v) = \begin{cases} \text{the longest proper suffix of } v \text{ in } \mathrm{Pref}(\mathcal{U}) \text{ if it is defined,} \\ \text{or } \varepsilon \text{ otherwise.} \end{cases}$$

In the following we identify a word $v \in \mathrm{Pref}(\mathcal{U})$ with the node reached by reading the letters of v while following the corresponding transitions on the tree seen as an automaton, so that $\mathrm{Border}()$ defines also a map from the set $\mathrm{Pref}(\mathcal{U})$ on the set of nodes of the tree. There are efficient $\mathcal{O}(|\mathcal{U}|)$ algorithms [Aho and Corasick 1975; Crochemore and Rytter 2002] linear both in time and space to build such a tree structure and the auxiliary $\mathrm{Border}()$ function.

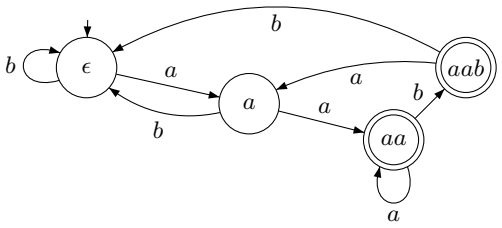
The matrix $\mathbb{T}(\mathbf{x})$ (with $\mathbf{x}$ an r-vector of formal variables) denotes the weighted transition matrix of the Aho-Corasick automaton where the variable x_i marks the states accepting the word u_i. The generating function is expressed as

$$F(z,\mathbf{x}) = \sum_{w \in \mathcal{A}^\star} \pi(w) z^{|w|} \mathbf{x}^{\boldsymbol{\tau}(w)} = (1,\ 0,\ \cdots,\ 0)\ (\mathbb{I} - z\mathbb{T}(\mathbf{x}))^{-1} \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}, \tag{3}$$

where $\pi(w)$ can be viewed as the weight of the word w.

Example 3.1. Let $\mathcal{U} = \{aa, aab\}$. Ordering the states of the automaton following the lexicographical order, we have, with $\alpha = \pi(a)$, $\beta = \pi(b)$, and $\mathbf{x} = (x_1, x_2)$

$$\mathbb{T}(x_1,x_2) = \begin{pmatrix} \beta & \alpha & 0 & 0 \\ \beta & 0 & \alpha x_1 & 0 \\ 0 & 0 & \alpha x_1 & \beta x_2 \\ \beta & \alpha & 0 & 0 \end{pmatrix},$$



and

$$F(z,x_1,x_2) = \frac{1-\alpha(x_1-1)z}{1-z(\alpha x_1 + \beta - \alpha\beta(x_1-1)z + \alpha^2\beta x_1(x_2-1)z^2)}.$$

As mentioned in the introduction, a myriad of information can be extracted from such a generating function. The next few examples illustrate basic uses of generating functions.

— The coefficient $[z^n x_1^{n_1} x_2^{n_2}]F(z,x_1,x_2)$ is the probability in the Bernoulli model (where $\alpha+\beta=1$) that a random text of size n has n_1 occurrences of aa and n_2 occurrences of aab.

— In the enumerative case ($\alpha=\beta=1$), the coefficient $[z^n x_1^{n_1} x_2^{n_2}]F(z,x_1,x_2)$ counts the number of words of length n with n_1 occurrences of aa and n_2 occurrences of aab. Any computer algebra system can compute the first terms of the Taylor series of a rational function. In the example above, we have

$$\begin{aligned} F(z,x_1,x_2) =& 1 + 2z + (x_1+3)z^2 + (x_1x_2 + x_1 + x_1^2)z^3 \\ &+ (3x_1x_2 + 2x_1 + x_1^2 + x_1^2x_2 + x_1^3 + 8)z^4 + O(z^5); \end{aligned}$$

this entails for instance that amongst the words of length 4 (corresponding to the term in z^4), we have the following correspondence between the terms of the generating function, the texts, and the occurrences statistics $\tau(w)$ of aa and aab in a text w.

Term	Texts w of length 4	$\tau(w)$
$3x_1x_2$	$\{baab, aabb, aaba\}$	$(1,1)$
$2x_1$	$\{bbaa, abaa\}$	$(1,0)$
x_1^2	$\{baaa\}$	$(2,0)$
$x_1^2x_2$	$\{aaab\}$	$(2,1)$
x_1^3	$\{aaaa\}$	$(3,0)$
8	$\{bbbb, bbba, bbab, babb, baba, abbb, abba, abab\}$	$(0,0)$

Complexity. Let $L = \sum_{u\in\mathcal{U}} |u|$ be the sum of the lengths of the words of $\mathcal{U}$. We first have to compute the Aho-Corasick automaton and this can be done classically in time $\mathcal{O}(L)$ for a finite alphabet. The automaton can have up to L states. Denoting by N the number of states of the Aho-Corasick automaton, the transitions matrix $\mathbb{T}$ is of size N^2, but in general this matrix is sparse: only $N \times \operatorname{Card}\mathcal{A}$ entries are non-zero (since the automaton is complete and deterministic with $\operatorname{Card}\mathcal{A}$ transitions from each state).

So the complexity to obtain the counting multivariate generating function by this approach is basically the one of inverting a relatively sparse matrix of the form $\mathbb{I}-z\mathbb{T}(\mathbf{x})$ all terms of which are linear polynomials in z with coefficients that are monomials of the form $\alpha\prod x_i^{\varepsilon_i}$ (with $\alpha = \pi(\ell)$ for $\ell \in \mathcal{A}$ and $\varepsilon_i \in \{0,1\}$); these coefficients correspond to the transition matrix of the automaton. The limit of this approach is the fact that the size of the transition matrix can grow rapidly if we consider many rather long words. In the two next sections, we adopt the analytic inclusion-exclusion approach which leads also to solve a system of equations, but then the size of the system is $r \times r$ (where r is the number of words in $\mathcal{U}$).

4. REDUCED CASE OF WORD COUNTING BY INCLUSION-EXCLUSION

We give in this section an intuitive introduction to the inclusion-exclusion method for words counting of [Goulden and Jackson 1983]. This method uses a principle of *overcounting* that is later reversed by a simple *algebraic substitution*; the overall process is known as *analytic inclusion-exclusion*. Note that the language approach [Régnier 2000] that follows previous work [Régnier and Szpankowski 1998] provides the same multivariate generating functions as Goulden and Jackson do. The principle of overcounting and inclusion-exclusion however has the property of extending nicely to the general case of non-reduced patterns that we present in the next section.

4.1. Intuitive approach to counting by inclusion-exclusion

The idea behind inclusion-exclusion counting is that it is sometimes harder to specify a set of objects satisfying simultaneously a collection of conditions than a set of objects which violates some of these conditions[2].

Thus we introduce the notion of decorated text which allows for distinguishing only a subset of the occurrences of the pattern.

Definition 4.1 (*Decorated text*). Let $\mathcal{U} = \{u_1, \ldots, u_r\}$ be a pattern. A *decorated text* w of length n with respect to $\mathcal{U}$ is a pair $\mathsf{w} = (w, \mathcal{D})$ where $w \in \mathcal{A}^\star$ is a text of length n, and denoting by $\mathcal{O} = (\mathcal{O}_i)_{i=1}^n$ the occurrence index, we have $\mathcal{D} = (\mathcal{D}_i)_{i=1}^n$ such that $\mathcal{D}_i \subseteq \mathcal{O}_i$ for all $1 \le i \le n$. The occurrences signaled by the n-tuple $\mathcal{D}$ are called *distinguished*.

[2]An application of the principle "it is easier to forget something than to remember everything".

The weight of a decorated text $\pi(\mathsf{w})$ is inherited from the weight of the underlying text, *i.e.*, $\pi(\mathsf{w}) = \pi(w)$. When $\mathcal{D} = \mathcal{O}$, we say that the text is *fully decorated*. The text w is called the support of w, and we write $|\mathsf{w}|$ for the length of w, that we define as $|\mathsf{w}| = |w|$.

A visual and succinct way to represent decorated texts is to represent the text while adding above the letter at position i the indices from the corresponding set $\mathcal{D}_i$. For instance considering the text $w = aababaabbbabaa$ and the pattern $\mathcal{U} = \{aba\}$ we obtain as representation when all occurrences are distinguished

$$a\overset{\bullet 1}{a}\overset{\bullet 1}{b}\overset{\bullet 1}{a}abbb\overset{\bullet 1}{a}baa$$

This representation readily generalizes to the case of several words for $\mathcal{U}$; we however have to label the marks according to the corresponding occurrences. Considering the text $w = abaaabaabb$ and the pattern $\mathcal{U} = \{u_1 = aa, u_2 = baa\}$, we get, when distinguishing all occurrences, the following decorated word

$$aba\overset{\bullet 1}{\overset{\bullet 2}{a}}\overset{\bullet 1}{a}ba\overset{\bullet 1}{\overset{\bullet 2}{a}}bb, \tag{4}$$

where ❶ and ❷ are the indices that signal the end positions of the occurrences of u_1 and u_2 respectively; we remark that two words can end at the same position.

CONVENTION 4.2. *When considering a word u, the associated decorated word built upon u, where the only distinguished occurrence is u itself, is denoted by the sans-serif letter* u .

A graphical representation for four examples of decorated texts corresponding to the text in (4) is depicted below

$$aba\overset{\bullet 1}{a}aba\overset{\bullet 1}{a}bb, \quad aba\overset{\bullet 1}{a}aba\overset{\bullet 1}{a}bb, \quad abaaabaabb, \quad aba\overset{\bullet 1}{\overset{\bullet 2}{a}}\overset{\bullet 1}{a}ba\overset{\bullet 1}{\overset{\bullet 2}{a}}bb.$$

The last two decorated texts correspond respectively to the respective cases where no occurrence is distinguished and where all occurrences are distinguished (the fully decorated case).

Naturally, a text with exactly k occurrences of a pattern will give rise to 2^k decorated texts (each occurrence may be distinguished or not). It is important to note that two texts decorated differently are considered distinct.

Our initial problem was to count the set of all texts together with all occurrences considered. We will instead count the set all of decorated texts (considering all ways to distinguish occurrences). Indeed this appear to be a significantly easier task. Going back from the counts of decorated texts (where texts are overcounted) to the counts of texts is done by use of the inclusion-exclusion principle (see among others [Goulden and Jackson 1983, 2.2.28, 2.2.29], [Szpankowski 2001, 3.2], and [Flajolet and Sedgewick 2009, III.7.4] for details). This gives an elegant solution to the problem.

4.1.1. Toy examples. Let us consider the simple case of a text $\mathcal{P} = aaaa$ and a pattern with a single word $\mathcal{U} = \{u = aaa\}$. We get the fully decorated text by considering all occurrences of aaa in $aaaa$

$$\overset{\bullet 1}{\overset{\bullet 1}{a}}\overset{\bullet 1}{a}\overset{\bullet 1}{a}a.$$

This yields the generating function $P(z, x) = \pi(a)^4 z^4 x^2$ (where x counts the number of occurrences of the word u, and z the length of the text). The set of the four decorated texts for this example is accordingly

$$\mathsf{Q} = \{\overset{\bullet 1}{\overset{\bullet 1}{a}}\overset{\bullet 1}{a}\overset{\bullet 1}{a}a, a\overset{\bullet 1}{a}\overset{\bullet 1}{a}a, aa\overset{\bullet 1}{a}\overset{\bullet 1}{a}, aaaa\};$$

this gives the generating function of the decorated texts for $aaaa$

$$Q(z,t) = \sum_{\mathsf{w}\in\mathsf{Q}} \pi(\mathsf{w}) z^{|\mathsf{w}|} t^{\#\text{distinguished occurrences}} = \pi(a)^4 z^4 (t^2 + t + t + 1),$$

(where the variable t counts the distinguished occurrences ❶ and z the length of the decorated text). The relation between $P(z,x)$ and $Q(z,t)$ is simply $Q(z,t) = P(z,1+t)$ since the substitution $x \to t+1$ parallels the fact that an occurrence may (or not) be distinguished (and then counted by the variable t). This relation can be used the other way around $P(z,x) = Q(z,x-1)$.

This variable change $t \to x-1$ is in fact quite general and is the essence of the inclusion-exclusion principle for generating functions. It readily extends to the case of a pattern with several words. Consider the text $\mathcal{P} = aaaaba$ and the pattern $\mathcal{U} = \{u_1 = aaa, u_2 = aba\}$; the fully decorated text (signaling all occurrences)

$$\overset{\textbf{❶❶},\textbf{❷}}{aaaaba}$$

gives the generating function $P(z,x_1,x_2) = \pi(a)^5\pi(b)z^6x_1^2x_2$ that counts all occurrences. There are 2^3 decorated texts for the word $\mathcal{P}$ (each occurrence may be distinguished or not), forming the set

$$\{aaaaba, \overset{\textbf{❶}}{aaaaba}, \overset{\textbf{❶}}{aaaaba}, \overset{\textbf{❶❶}}{aaaaba}, \overset{\textbf{❷}}{aaaaba}, \overset{\textbf{❶},\textbf{❷}}{aaaaba}, \overset{\textbf{❶}\ \textbf{❷}}{aaaaba}, \overset{\textbf{❶❶},\textbf{❷}}{aaaaba}\},$$

so that the generating function of decorated texts is $Q(z,t_1,t_2) = \pi(a)^5\pi(b)z^6(1+t_1+t_1+t_1^2+t_2+t_1t_2+t_1t_2+t_1^2t_2) = P(z,t_1+1,t_2+1)$.

4.1.2. Combinatorial description of decorated texts. To put into application the inclusion-exclusion principle, a general construction of all decorated texts has to be derived. We consider an alphabet $\mathcal{A}$ and a set of patterns $\mathcal{U} = \{u_1,\dots,u_r\}$, with $u_1 \prec \cdots \prec u_r$ for the lexicographic order.

We define hereafter the fundamental notion of cluster.

Definition 4.3 (*Cluster*). A cluster c with respect to a pattern $\mathcal{U}$ is a decorated text such that

— all positions are covered by at least a distinguished occurrence,
— and, either there is only one distinguished occurrence, or any distinguished occurrence has an overlap with another distinguished occurrence.

Let us denote by $\mathsf{C}_{\mathcal{U}}$ the class of all clusters for a pattern $\mathcal{U}$ (or C when the context is clear).

Then the set of decorated texts T decomposes as sequences of either arbitrary letters of the alphabet $\mathcal{A}$ or clusters

$$\mathsf{T} = (\mathcal{A} + \mathsf{C})^\star. \tag{5}$$

Figure 1 illustrates a particular decorated text that is an element of T. To apply directly the generating function methodology (see [Flajolet and Sedgewick 2009]), it is essential that this decomposition is unambiguous: for a given decorated text, there is a unique way to decompose it along the language Equation (5). This property is actually true because the expansion of the right member of Equation (5) is composed of non-intersecting sets: the sets $\mathcal{A}$ and C are always distinct (as decorated texts), the concatenation is a non-commutative product, and finally clusters are well delimited since $\mathsf{C} \cap \mathsf{C}\cdot\mathsf{C} = \emptyset$. As a remark Figure 1 illustrates the fact that two clusters may appear one immediately after the other.

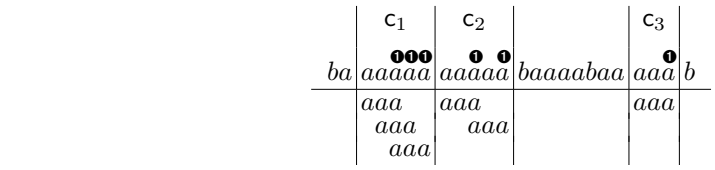


Fig. 1. We consider the text $w = baaaaaaaaaaabaaaabaaaaab$, the pattern $\mathcal{U} = \{aaa\}$, and a particular decorated text with three clusters c_i ($i = 1, 2, 3$). The alphabet is $\mathcal{A} = \{a, b\}$. On this graphical representation, we write below the text the distinguished occurrences to stress out the fact that these occurrences overlap. Since $\mathrm{Card}(\mathcal{U}) = 1$ the symbol ❶ signals distinguished occurrences of u (the label is here redundant).

Now, let us assume that we know how to compute the generating function $\xi(z, \mathbf{t})$ of the set of clusters C,

$$\xi(z, \mathbf{t}) = \sum_{\mathsf{w} \in \mathsf{C}} \pi(\mathsf{w}) z^{|\mathsf{w}|} \mathbf{t}^{\boldsymbol{\tau}(\mathsf{w})}, \tag{6}$$

where $\boldsymbol{\tau}(\mathsf{w}) = (|\mathsf{w}|_1, \ldots, |\mathsf{w}|_r)$ and, by analogy with Equation (1) p. 4, the quantity $|\mathsf{w}|_i$ denotes the number of *distinguished* occurrences of u_i in w. It then follows from Equations (5) and (6) and general principles [Flajolet and Sedgewick 2009] that the generating function $T(z, \mathbf{t})$ of all decorated texts is

$$T(z, \mathbf{t}) = 1 + \Big(A(z) + \xi(z, \mathbf{t}\Big) + \Big(A(z) + \xi(z, \mathbf{t})\Big)^2 + \cdots = \frac{1}{1 - A(z) - \xi(z, \mathbf{t})}. \tag{7}$$

so that the sought generating function is

$$F_{\mathcal{U}}(z, \mathbf{x}) = \frac{1}{1 - A(z) - \xi(z, \mathbf{x} - \mathbf{1})}. \tag{8}$$

Therefore, we have reduced the problem of computing the generating function $F_{\mathcal{U}}(z, \mathbf{t})$ to the one of computing the generating function of the set of clusters $\xi(z, \mathbf{t})$. This is quite simple when the pattern $\mathcal{U}$ is reduced, and more difficult in the non-reduced case as will be shown in Section 5.

4.2. Clusters for one word patterns

Let us explore the case of counting occurrences of one word u in texts over the alphabet $\{a, b\}$. Considering clusters for this case appears first in [Jacquet and Szpankowski 1994]. To build the set of clusters C of $u = aaa$ in the present case, we can write[3]

$$\mathsf{C} = aa\overset{❶}{a} \cdot \Big(\overset{❶}{a} + a\overset{❶}{a}\Big)^{\star}. \tag{9}$$

Remark that $\{a, aa\} = \mathcal{C}_u - \varepsilon$ where ε is the empty word. Note also that in this expression the symbol ❶ states which occurrences are distinguished in the clusters. The bivariate generating function $\xi(z, t)$ of C is obtained from this expression by counting the distinguished occurrences, *i.e.*, symbols ❶, with the variable t. Accordingly, Equa-

[3] Strictly speaking, the decorated words $\overset{❶}{a}$ and $a\overset{❶}{a}$ obtained upon the words a and aa are not valid *per se* since the word aaa is neither a factor of a nor of aa. We use here a slight abuse of language, as we write decorated suffixes in the context of a cluster, so that decorations always correspond to valid occurrences.

tion (9) translates to

$$\xi(z,t) = \sum_{\mathsf{c}\in\mathsf{C}} \pi(\mathsf{c}) z^{|\mathsf{c}|} t^{\#\text{ distinguished occurrences in c}}$$
$$= \frac{tu(z)}{1-t(C(z)-1)} = \frac{t\pi(a)^3 z^3}{1-t(\pi(a)z+\pi(a)^2z^2)},$$

where t counts the number of distinguished occurrences and $u(z)$ and $C(z)$ respectively are the generating functions of the word u and of the autocorrelation set $\mathcal{C}$ of u. Then making use of the symbolic exclusion-inclusion principle and of Equation (8) and denoting by $|w|_u$ the number of occurrences of u in w, we directly get

$$F(z,x) = \sum_{w\in\mathcal{A}^\star} \pi(w) z^{|w|} x^{|w|_u} = \frac{1}{1-A(z)-\xi(z,x-1)},$$

where $A(z)$ is the generating function of the alphabet $\mathcal{A}$. Considering again the word $u = aaa$ and the binary alphabet $\mathcal{A} = \{a,b\}$, and posing $\pi(a) = \pi(b) = 1$ (to get the enumerative generating function), we have $C(z) = 1+z+z^2$ and we obtain

$$F(z,x) = \cfrac{1}{1-2z-\cfrac{(x-1)z^3}{1-(x-1)(z+z^2)}}.$$

4.3. Clusters for several words in the reduced case

When considering the reduced case for several words, the method described in the preceding section readily applies; the only difference is that we need matrix products to describe the clusters.

Considering the example $\mathcal{U} = \{u_1 = aaab, u_2 = baaa\}$. A typical cluster is

$$\begin{array}{l} baaa\overset{\bullet 2}{a}\overset{\bullet 1}{a}baaa\overset{\bullet 2}{a}\overset{\bullet 1}{a}b \\ \hline baaa \\ \quad aaab \\ \quad\quad baaa \\ \quad\quad\quad aaab \end{array}$$

We see on this example that to extend a cluster to the right in the reduced case, we only need to consider the last distinguished occurrence (say u_1) in the cluster and append a word from the correlation sets $\mathcal{C}_{u_1,u_2}$ or $\mathcal{C}_{u_1,u_1}$ defined in Equation (2) p. 4 to obtain the next distinguished occurrence still overlapping the previous one (by definition of the correlation sets). Informally we make clusters grow by starting from a seed (a word of $\mathcal{U}$ which is distinguished) and concatenating words of correlations sets, adding each time a new distinguished occurrence.

Let us consider the set of decorated words $\{\mathsf{u}_1,\ldots,\mathsf{u}_r\}$ corresponding to the pattern $\mathcal{U} = \{u_1,\ldots,u_r\}$ and the matrix of decorated correlation sets $\mathsf{Q} = (\mathsf{Q}_{i,j})$ defined for $1 \le i,j \le r$ by

$$\mathsf{Q}_{i,j} = \bigcup_{\substack{e\in\mathcal{C}_{i,j}\\ e\neq\varepsilon}} \{\mathrm{Suff}_{|e|}(\mathsf{u}_j)\} \quad \text{where} \quad \mathcal{C}_{i,j} = \mathcal{C}_{u_i,u_j}; \tag{10}$$

the notation $\mathrm{Suff}_\ell(u)$ denotes here the suffix of length ℓ of a word u and the notation $\mathrm{Suff}_\ell(\mathsf{u})$ is the corresponding decorated word; we use here again an abuse of notation (see the footnote 3 p.9). Then the matrix formula giving the set of all decorated clusters

is

$$\mathsf{C} = (\mathsf{u}_1, \ldots, \mathsf{u}_r)\ \mathsf{Q}^\star \begin{pmatrix} \varepsilon \\ \varepsilon \end{pmatrix}, \tag{11}$$

where, for a matrix $\mathbb{M}$, we write

$$\mathbb{M}^\star = (\mathbb{I} - \mathbb{M})^{-1} = \mathbb{I} + \mathbb{M} + \mathbb{M}^2 + \ldots.$$

For instance with $\mathcal{U} = \{u_1 = aaab, u_2 = baaa\}$, the set of clusters C is given by

$$\mathsf{C} = (aaa\overset{\bullet 1}{b}, baa\overset{\bullet 2}{a}) \begin{pmatrix} \emptyset & \left\{aa\overset{\bullet 2}{a}\right\} \\ \left\{\overset{\bullet 1}{b}, a\overset{\bullet 1}{b}, aa\overset{\bullet 1}{b}\right\} & \emptyset \end{pmatrix}^\star \begin{pmatrix} \varepsilon \\ \varepsilon \end{pmatrix}.$$

The translation to the generating function $\xi(z, \mathbf{t})$ is extremely easy and mirrors the previous combinatorial expression. For the last example the generating function $\xi(z, t_1, t_2)$ of clusters (setting $\pi(a) = \pi(b) = 1$ for clarity, and applying the map $\bullet 1 \mapsto t_1, \bullet 2 \mapsto t_2, \alpha \mapsto z$ with $\alpha \in \mathcal{A}$) verifies

$$\xi(z, t_1, t_2) = \begin{pmatrix} z^4 t_1, & z^4 t_2 \end{pmatrix} \left(\mathbb{I} - \begin{pmatrix} 0 & z^3 t_2 \\ (z + z^2 + z^3) t_1 & 0 \end{pmatrix} \right)^{-1} \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

Equation (8) applies then again, yielding the generating function of occurrences.

Short bibliographic note about applications of the Goulden-Jackson method

The inclusion-exclusion method of Goulden-Jackson is extremely powerful. Its applications go far beyond enumeration of texts. As a direct application of the basic definitions, [Goulden and Jackson 1983, 2.2.30] provide the number of derangements for permutations of size m. As another application, [Flajolet and Sedgewick 2009, III,7,4] count rises in permutations.

Among the articles applying the inclusion-exclusion method and the use of clusters to count texts with forbidden patterns, we mention the following. [Noonan 1998] evaluates connective constants of self-avoiding walks. [Edlin and Zeilberger 2000] consider cyclic words with forbidden patterns and [Zeilberger 2002] words with non-regular infinite forbidden patterns. Considering also words with forbidden patterns, [Wen 2005] uses symmetries of the set of words of the pattern to shrink the size of the matrix or linear system, while [Kupin and Yuster 2010] handle Markov sources. Finally, which will be the topic of the next Section, [Noonan and Zeilberger 1999] consider the general case of words counting. However, we point out that considering forbidden patterns corresponds to the case of reduced patterns; none of the articles mentioned here provide equivalents of the proofs and formulas that we give in the next Sections.

5. GENERAL CASE OF WORD COUNTING BY INCLUSION-EXCLUSION

We remark first that in the general *non-reduced* case, there is no known method of language decompositions similar to the approach of [Régnier and Szpankowski 1998], where a text is "scanned" with respect of *all* the occurrences of the pattern. Our goal is therefore to generalize the process of inclusion-exclusion to any finite set of words. The preceding section provides the main lines of the inclusion-exclusion method for reduced patterns as given in [Goulden and Jackson 1983].

This section extends this approach to the non-reduced case. See also [Noonan and Zeilberger 1999] that provides Maple scripts for this non-reduced case. Note also that if the language decomposition of Régnier and Szpankowski in the reduced case is a relatively easy combinatorial step, the combinatorial decomposition in the non-reduced

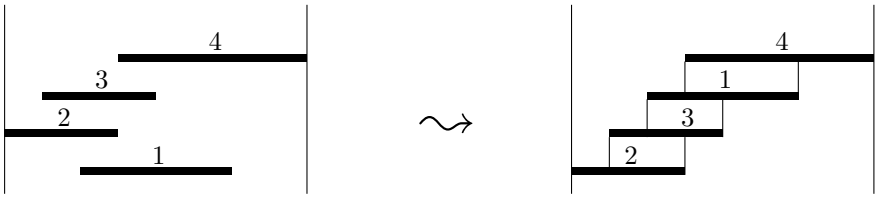


Fig. 2. The nice property of *double staircase* shape for reduced patterns. (Left) Random ordering of the occurrences falling like dominoes in a Tetris game. (Right) Double staircase reordering.

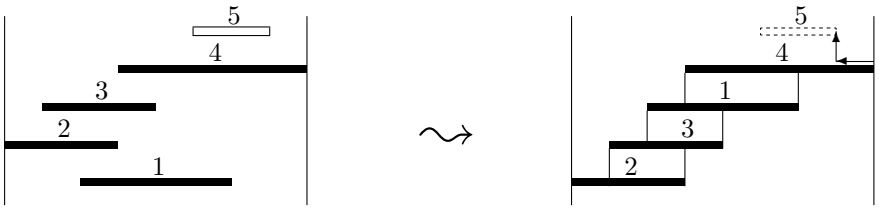


Fig. 3. Occurrence 5 breaks the *double staircase* property; There is no reordering of the five occurrences that do not break the property. A *skeleton* (or reduced cluster) of the cluster will be built with occurrences 1 to 4. Occurrence 5 is a *factor* occurrence of occurrence 4 that will be add to the skeleton in a *flip-flop* manner that corresponds to the fact that this occurrence can be marked or left unmarked.

case is harder; in both cases a trivial analytic manipulation follows and yields the sought generating function.

We introduce this section by mentioning an apparently trivial property verified in the reduced case. In general, this property is violated in the non-reduced case, but there are subsets of the distinguished occurrences that still verify it; the construction used in the non-reduced case will built upon one of these subsets that we will call *skeleton* of the cluster.

The double staircase property. Assume that we randomly number the occurrences of a cluster, and that each occurrence is represented by a thin domino, where the horizontal position of each domino is the position of the corresponding occurrence. Let now fall the dominoes from above, like in a Tetris game (Figure 2 (left)). In the case of reduced patterns, there is a simple and obvious property; there is a reordering of the occurrences such that letting fall the dominoes produces a double *staircase* shape (where the steps have unit height), one corresponding to the left side of the dominoes, and the other to the right side (Figure 2 (right)). Coming back to words, the progression from a given domino to the next one in this ordering corresponds to append a word belonging from the correlation set from the word whose occurrence corresponds to the given domino to the word whose occurrence corresponds to the next domino; this appears clearly in the example $\mathcal{U} = \{u_1 = aaba, u_2 = baaa\}$ of Section 4.3. This property can be violated in the case of non-reduced patterns, but as seen in the cluster of Figure (3), it is possible to build what will be called in the following the *skeleton* of the cluster that verifies the staircase property.

5.1. Combinatorial description of clusters in the general case

We exhibit a property of decorated texts which will prove useful for factorizing clusters.

Definition 5.1 (*Reduced decorated text*). A decorated text is said to be reduced if no distinguished occurrence is a factor of another distinguished one.

Note that this property is automatically granted if the pattern $\mathcal{U}$ is reduced. We define a particular class of clusters called skeletons, which have this property.

Definition 5.2 (*Skeleton*). A skeleton is a cluster such that no distinguished occurrence is a factor of another distinguished occurrence.

We introduce also two dual operations, denoted by Skel and Flip which relate clusters and skeletons.

Definition 5.3 (*Skeletization and flip operation*). The two dual operations Skel and Flip are defined as:

— Let c be a cluster, the skeleton $\mathrm{Skel}(\mathsf{c})$ (denoted also $\underline{\mathsf{c}}$) of a decorated text c is obtained from c by undistinguishing (moving the status of an occurrence from "distinguished" to "not distinguished") the factor occurrences in c.
— Let $\underline{\mathsf{c}}$ be a skeleton, the Flip operation associates to $\underline{\mathsf{c}}$ the set $\mathrm{Flip}(\underline{\mathsf{c}})$ of all clusters c such that $\mathrm{Skel}(\mathsf{c}) = \underline{\mathsf{c}}$.

We have the following lemma for clusters.

LEMMA 5.4. *The skeleton* $\mathrm{Skel}(\mathsf{c})$ *of a cluster* c *is uniquely defined. It is a cluster and the distinguished occurrences in* $\mathrm{Skel}(\mathsf{c})$ *can be increasingly ordered with respect to their end positions such that each occurrence overlap the following one, when it exists. This ordering is unique.*

PROOF. We omit the proof which is very simple. □

Example 5.5. Let us consider the pattern $\mathcal{U} = \{u_1 = ab, u_2 = ba, u_3 = baba\}$ and the clusters

$$\begin{array}{lll} \mathsf{c}_1 = \overset{\bullet 1}{a}\overset{\bullet 1}{b}\overset{\bullet 3}{a}\overset{\bullet 3}{b}a, & \mathsf{c}_2 = \overset{\bullet 1}{a}b\overset{\bullet 2}{a}\overset{\bullet 2}{b}\overset{\bullet 3}{a}\overset{\bullet 3}{b}a, & \mathsf{c}_3 = \overset{\bullet 1}{a}\overset{\bullet 2}{b}\overset{\bullet 1}{a}\overset{\bullet 2}{b}\overset{\bullet 2}{a}\overset{\bullet 3}{b}a \\ \hline \begin{array}{ll} ab & baba \\ baba & \\ ab & \end{array} & \begin{array}{ll} ab & ba \\ baba & \\ baba & \\ ba & \end{array} & \begin{array}{ll} ab & ba \\ ba & \\ ab & \\ baba & \end{array} \end{array}$$

We have

$$\mathrm{Skel}(\mathsf{c}_1) = \mathrm{Skel}(\mathsf{c}_2) = \overset{\bullet 1}{a}b\overset{\bullet 3}{a}\overset{\bullet 3}{b}a, \quad \mathrm{Skel}(\mathsf{c}_3) = \overset{\bullet 1}{a}\overset{\bullet 2}{b}\overset{\bullet 1}{a}\overset{\bullet 3}{b}a.$$

This example illustrates that two different clusters with same support (here $abababa$) can have different skeletons.

Now, the general strategy to describe clusters is to build reduced clusters, and along this process to identify all factor occurrences (mirroring the Flip operation).

We therefore introduce a notation aimed at representing factor occurrences produced by the Flip operation for a skeleton.

Definition 5.6 (*Bicolored decorated cluster*). Let $\underline{\mathsf{c}} = (c, \mathcal{D})$ be a skeleton with respect to a pattern $\mathcal{U} = \{u_1, \ldots, u_r\}$. We denote by $\mathcal{O}$ the occurrence index of $\mathcal{U}$ in c, and, as previously, by $\mathcal{D} \subseteq \mathcal{O}$ the set of indices of distinguished occurrences in $\underline{\mathsf{c}}$. The fully bicolored decorated word $\widetilde{\underline{\mathsf{c}}}$ associated to $\underline{\mathsf{c}}$ is a pair $\widetilde{\underline{\mathsf{c}}} = (\underline{\mathsf{c}}, \mathcal{F})$ where $\mathcal{F} \subset \mathcal{O} \setminus \mathcal{D}$ is the set of indices of the distinguished *factor* occurrences within the occurrences indexed by $\mathcal{D}$.

In the graphical representation, we will denote by a different mark (white filled circles) factor occurrences of distinguished occurrences. Hence for instance, considering the skeleton $\underline{\mathsf{c}}$ for $\mathcal{U} = \{ab, ba, baba\}$

$$\underline{\mathsf{c}} = \overset{\bullet 1}{a}b\overset{\bullet 3}{a}\overset{\bullet 3}{b}a,$$

the set $\mathrm{Flip}(\underline{c})$ is the set of clusters having $\underline{c}$ as skeleton and can be identified to the following *bicolored decorated word*

$$\widetilde{\underline{c}} = \mathrm{Flip}(\underline{c}) = a\overset{\bullet}{b}\overset{\circ}{a}\overset{\circ}{b}\overset{\bullet}{a}\overset{\circ}{b}\overset{\bullet}{a}, \tag{12}$$

where end positions of occurrences belonging to the skeleton are signaled by black filled circles, and factor occurrences are signaled by white filled circles. This notation gives us a way to represent all the clusters sharing the same skeleton. As a matter of facts, there is no conceptual difference between bicolored decorated words and the set of decorated words with the same skeleton obtained by examining all ways of distinguishing factor occurrences. For instance the fully bicolored decorated cluster of (12) is strictly equivalent to the set containing $2^5 = 32$ (there are five factor occurrences) differently decorated clusters.

Remark 5.7 (*Integrity rule for Flips of skeletons*). In Definition 5.6 we flip *only* occurrences which are *factors* of the *distinguished* occurrences of the skeleton. So, by Definition 5.6, for two different skeletons $\underline{c}_1$ and $\underline{c}_2$, we have

$$\mathrm{Flip}(\underline{c}_1) \cap \mathrm{Flip}(\underline{c}_2) = \emptyset.$$

We provide an example for the last remark by considering the pattern $\mathcal{U} = \{u_1 = aaa, u_2 = aaaaaaa\}$, a skeleton $\underline{c}_1$, and the corresponding $\mathrm{Flip}(\underline{c}_1)$. In order to improve the readability here we decompose clusters according to their skeletons.

$$\underline{c}_1 = \begin{array}{l} \overset{\bullet}{a}aaaaaaa\overset{\bullet}{a} \\ \overline{aaa} \\ \quad aaaaaaa \end{array} \Bigg| \Longrightarrow \mathrm{Flip}(\underline{c}_1) = aa\overset{\bullet}{a} \cdot a\overset{\circ}{a}\overset{\circ}{a}\overset{\circ}{a}\overset{\circ}{a}\overset{\bullet}{a}. \tag{13}$$

We remark here that the fourth position has no label $\circ$ signaling a factor occurrence aaa; indeeds considering a factor occurrence aaa at this position would break the integrity rule and correspond to a skeleton $\underline{c}_2$ different of $\underline{c}_1$, namely,

$$\underline{c}_2 = \begin{array}{l} \overset{\bullet}{a}\overset{\bullet}{a}aaaaaaa\overset{\bullet}{a} \\ \overline{aaa} \\ \quad aaa \\ \quad\quad aaaaaaa \end{array} \Bigg| \Longrightarrow \mathrm{Flip}(\underline{c}_2) = aa\overset{\bullet}{a} \cdot \overset{\bullet}{a} \cdot a\overset{\circ}{a}\overset{\circ}{a}\overset{\circ}{a}\overset{\circ}{a}\overset{\bullet}{a}. \tag{14}$$

Right extensions. The skeletons will basically be built on a matrix construction similar to the one of the reduced case (see Equation (10)). Next, in a second step, the factor occurrences are added to the skeleton. As shown in the following example, the classical definition of correlations of words is not compatible with the definition of skeletons.

Considering the words $u = a^3$ and $v = a^7$, following the definition of correlation (Equation (2) p. 4), we have $\mathcal{C}_{a^3,a^7} = \{a^4, a^5, a^6\}$. How however can we go from an occurrence of a^3 to an occurrence of a^7 in a skeleton? We have the three cases

(i) $\underline{\begin{array}{l} aaa \\ \hline aaaa \end{array}}$ (ii) $\underline{\begin{array}{l} aaa \\ \hline aaaaa \end{array}}$ (iii) $\underline{\begin{array}{l} aaa \\ \hline aaaaaa \end{array}}$

where the black rules underline the positions of the last occurrence of a^7. As seen in case (i), it is not possible to progress in a skeleton from an occurrence of a^3 to an occurrence a^7 by the word a^4; in this case the first occurrence of a^3 is a factor of the underlined occurrence of a^7, which is contradictory to the definition of skeletons. On the contrary, cases (ii) and (iii) correspond to valid extensions.

In order to properly generate the skeletons we therefore introduce the notion of *right extension* of a pair of words (u, v). This notion is a generalization of the correlation set of two words u and v but differs in that:

(i) overlapping is not allowed to start at the beginning of u.
(ii) extension has to add some letters to the right of u.

These two conditions ensure that, while scanning a text from left to right, going from one distinguished occurrence to another in a skeleton, both ending and beginning positions are changing, hence preventing from considering factor occurrences.

More formally we have

Definition 5.8 (*Right extension set*). The right extension set of a pair of words (u, v) is

$$\mathcal{E}_{u,v} = \{\, e \quad | \quad \text{there exists } e' \in \mathcal{A}^+ \quad \text{such that} \quad ue = e'v \quad \text{with } 0 < |e| < |v| \}.$$

Note that, when u and v have no factor relation, the right extension set $\mathcal{E}_{u,v}$ is the correlation set of u to v. Moreover, when $u = v$, the set $\mathcal{E}_{u,v}$ is the strict autocorrelation set of u (the empty word does not belong to $\mathcal{E}_{u,u}$). We can also define a decorated variant.

Definition 5.9 (*Bicolored decorated right extension set*). Let u and v be two words, and u and v be defined by Convention 4.2 p. 7. The bicolored decorated right extension set of the pair of words (u, v) is

$$\mathsf{E}_{u,v} = \bigcup_{e \in \mathcal{E}_{u,v}} \mathrm{Suff}_{|e|}(\mathrm{Flip}(\mathsf{v})),$$

where for a set V of bicolored decorated words, $\mathrm{Suff}_\ell(\mathsf{V})$ is the set of (bicolored decorated) suffixes of length ℓ from V.

We use again here the abuse of notation mentioned in footnote 3 p. 9.
We apply the last definition in the following example.

Example 5.10. We consider the pattern $\mathcal{U} = \{u_1, u_2\} = \{aa, aaa\}$ together with two particular clusters c_1 and c_2 as an illustration. We have

$$\mathsf{c}_1 = \begin{array}{l} \overset{\textcircled{1}\textcircled{1}}{\underset{\textcircled{1}\textcircled{2}\textcircled{2}}{aaaa}} \\ aaa \\ \quad aaa \end{array}, \quad \mathsf{c}_2 = \begin{array}{l} \overset{\textcircled{1}\quad\textcircled{1}\textcircled{1}}{\underset{\textcircled{1}\textcircled{2}\textcircled{1}\textcircled{2}}{aaaaa}} \\ aaa \\ \quad aaa \end{array}.$$

We observe that

$$\mathrm{Suff}_{|a|}(\mathrm{Flip}(\mathsf{u}_2)) = \{\overset{\textcircled{1}}{\underset{\textcircled{2}}{a}}\}, \mathrm{Suff}_{|aa|}(\mathrm{Flip}(\mathsf{u}_2)) = \{\overset{\textcircled{1}}{\underset{\textcircled{1}\textcircled{2}}{aa}}\}, \quad \text{and} \quad \mathsf{E}_{a^3,a^3} = \{\overset{\textcircled{1}}{\underset{\textcircled{2}}{a}}, \overset{\textcircled{1}}{\underset{\textcircled{1}\textcircled{2}}{aa}}\}.$$

As for correlation matrices, we define right extension matrices with respect to a pattern $\mathcal{U} = \{u_1, \ldots, u_r\}$ with indices of words dictated by the lexicographical order

$$\mathcal{E} = \left(\mathcal{E}_{u_i,u_j}\right)_{1 \le i,j \le r}.$$

We define also accordingly the decorated variant E of the right extension matrix $\mathcal{E}$ of dimension $r \times r$.

Example 5.11. We give some examples of patterns and their right extension matrices (non-decorated and decorated).

(1) For $\mathcal{U} = \{ab, aba\}$, we have $\mathcal{E} = \begin{pmatrix} \emptyset & \emptyset \\ b & ba \end{pmatrix}$. This gives, by Convention 4.2 p. 7, $\mathsf{u}_1 = \overset{\textcircled{1}}{ab}$ and $\mathsf{u}_2 = \overset{\textcircled{2}}{aba}$, so that $\mathrm{Flip}(\mathsf{u}_1) = \{\overset{\textcircled{1}}{ab}\}$ and $\mathrm{Flip}(\mathsf{u}_2) = \{\overset{\textcircled{1}}{\underset{\textcircled{2}}{aba}}\}$, when using

the bicolored decorated notation[4], while the decorated right extension matrix verifies

$$\mathsf{E} = \begin{pmatrix} \emptyset & \emptyset \\ \overset{\bullet 1}{\{b\}} & \{\overset{\circ 1}{b}\overset{\bullet 2}{a}\} \end{pmatrix}.$$

(2) For $\mathcal{U} = \{a^3, a^7\}$, we have[5]

$$\mathsf{u}_1 = \overset{\bullet 1}{aaa}, \mathsf{u}_2 = \overset{\bullet 2}{aaaaaaa} \text{ so that } \mathrm{Flip}(\mathsf{u}_1) = \{\overset{\bullet 1}{aaa}\}, \quad \mathrm{Flip}(\mathsf{u}_2) = \{\overset{\circ 1\circ 1\circ 1\circ 1\circ 1\bullet 2}{aaaaaaa}\},$$

and $\mathcal{E} = \begin{pmatrix} a + aa & a^5 + a^6 \\ a + aa & a + a^2 + a^3 + a^4 + a^5 + a^6 \end{pmatrix}$, so that

$$\mathsf{E} = \begin{pmatrix} \{\overset{\bullet 1}{a}, \overset{\bullet 1}{aa}\} & \{\overset{\circ 1\circ 1\circ 1\circ 1\bullet 2}{aaaaa}, \overset{\circ 1\circ 1\circ 1\circ 1\bullet 2}{aaaaaa}\} \\ \{\overset{\bullet 1}{a}, \overset{\bullet 1}{aa}\} & \{\overset{\bullet 2}{a}, \overset{\circ 1\bullet 2}{aa}, \overset{\circ 1\circ 1\bullet 2}{aaa}, \overset{\circ 1\circ 1\circ 1\bullet 2}{aaaa}, \overset{\circ 1\circ 1\circ 1\circ 1\bullet 2}{aaaaa}, \overset{\circ 1\circ 1\circ 1\circ 1\circ 1\bullet 2}{aaaaaa}\} \end{pmatrix}.$$

(3) As a slightly more complicated example the pattern $\mathcal{U} = \{aa, ab, ba, baaab\}$ gives

$$\mathsf{u}_1 = \overset{\bullet 1}{aa}, \qquad \mathsf{u}_2 = \overset{\bullet 2}{ab}, \qquad \mathsf{u}_3 = \overset{\bullet 3}{ba}, \qquad \mathsf{u}_4 = \overset{\bullet 4}{baaab},$$

$$\mathrm{Flip}(\mathsf{u}_1) = \{\overset{\bullet 1}{aa}\}, \quad \mathrm{Flip}(\mathsf{u}_2) = \{\overset{\bullet 2}{ab}\}, \quad \mathrm{Flip}(\mathsf{u}_3) = \{\overset{\bullet 3}{ba}\}, \quad \mathrm{Flip}(\mathsf{u}_4) = \{\overset{\circ 3\circ 1\circ 1\bullet 4\circ 2}{baaab}\},$$

$$\mathcal{E} = \begin{pmatrix} a & b & \emptyset & \emptyset \\ \emptyset & \emptyset & a & aaab \\ a & b & \emptyset & \emptyset \\ \emptyset & \emptyset & a & aaab \end{pmatrix} \text{ and } \quad \mathsf{E} = \begin{pmatrix} \{\overset{\bullet 1}{a}\} & \{\overset{\bullet 2}{b}\} & \emptyset & \emptyset \\ \emptyset & \emptyset & \{\overset{\bullet 3}{a}\} & \{\overset{\circ 3\circ 1\circ 1\bullet 4\circ 2}{aaab}\} \\ \{\overset{\bullet 1}{a}\} & \{\overset{\bullet 2}{b}\} & \emptyset & \emptyset \\ \emptyset & \emptyset & \{\overset{\bullet 3}{a}\} & \{\overset{\circ 3\circ 1\circ 1\bullet 4\circ 2}{aaab}\} \end{pmatrix}.$$

We introduce here the notion of $(k+1)$-skeleton.

Definition 5.12 ($(k+1)$-*skeleton*). We denote by $(k+1)$-skeleton a skeleton that is composed of $k+1$ occurrences and by $(k+1)$-cluster a cluster whose skeleton is a $(k+1)$-skeleton.

We state now the link between clusters and bicolored decorated right extensions.

PROPOSITION 5.13. *The set* C *of all clusters verifies*

$$\mathsf{C} = (\mathrm{Flip}(\mathsf{u}_1), \ldots, \mathrm{Flip}(\mathsf{u}_r)) \cdot \mathsf{E}^\star \cdot \begin{pmatrix} \varepsilon \\ \vdots \\ \varepsilon \end{pmatrix}. \tag{15}$$

PROOF. We recall that,

(1) given any cluster c, undistinguishing the factor occurrences leads to a skeleton $\underline{\mathsf{c}}$ such that $\mathsf{c} \in \mathrm{Flip}(\underline{\mathsf{c}})$,
(2) and, given two different[6] skeletons $\underline{\mathsf{c}}$ and $\underline{\mathsf{c}}'$, we have $\mathrm{Flip}(\underline{\mathsf{c}}) \cap \mathrm{Flip}(\underline{\mathsf{c}}') = \emptyset$.

---

[4] For instance $\{\overset{\circ 1\bullet 2}{aba}\}$ is equivalent to $\{\overset{\bullet 2}{aba}, \overset{\bullet 1\bullet 2}{aba}\}$.
[5] See the Remark 5.7 p. 14 and the example that follows the remark.
[6] See (13) and (14) p. 14 illustrating the integrity rule for Flips of skeletons (Remark 5.7).

These two properties imply that taking the Flip of all possible skeletons generates all possible clusters in a way where each cluster is generated exactly once.

Given a pattern $\mathcal{U} = \{u_1, \ldots, u_r\}$ and any $(k+1)$-skeleton $\underline{\mathsf{c}}$, the definition of skeletons yields that there are a unique sequence $(i_1, \ldots, i_{k+1})$ and a unique decomposition

$$\underline{\mathsf{c}} = (u \cdot w_1 \cdot w_2 \cdot \ \ldots \ \cdot w_k, (\mathcal{D}_d)_{1 \le d \le |\underline{\mathsf{c}}|}), \tag{16}$$

$$\text{where } \begin{cases} u = u_{i_1} \in \mathcal{U} \qquad \text{and,} \qquad \text{for } 1 \le j \le k, \quad w_i \in \mathcal{E}_{u_{i_j}, u_{i_{j+1}}}, \\ \mathcal{D}_{|u|} = i_1, \\ \mathcal{D}_{|u \cdot w_1 \cdot \ \ldots \ \cdot w_e|} = i_{j+1} \quad \text{if } w_e = \mathcal{E}_{u_{i_j}, u_{i_{j+1}}}, \\ \mathcal{D}_d = \emptyset \quad \text{if not defined previously.} \end{cases}$$

In this last equation, the sequence $(\mathcal{D}_d)$ records the distinguished positions of the skeleton $\underline{\mathsf{c}}$ and the corresponding indices.

For $1 \le j \le r$ we denote $\underline{\mathsf{u}}_j$ (resp. $\underline{\mathsf{E}}_{i,j}$) the monocolor words where there is only a label ❶ upon the last position (factor positions have been undistinguished). We also denote $\underline{\mathsf{E}} = (\underline{\mathsf{E}}_{i,j})$. Considering the set $\underline{\mathsf{C}}_{k+1}$ of $(k+1)$-skeletons, Equation (16) yields by considering all possible $(k+1)$-skeletons

$$\underline{\mathsf{C}}_{k+1} = (\underline{\mathsf{u}}_1, \ldots, \underline{\mathsf{u}}_r) \cdot \underline{\mathsf{E}}^k \cdot \begin{pmatrix} \varepsilon \\ \vdots \\ \varepsilon \end{pmatrix}, \text{ which implies that } \underline{\mathsf{C}} = (\underline{\mathsf{u}}_1, \ldots, \underline{\mathsf{u}}_r) \cdot \underline{\mathsf{E}}^\star \cdot \begin{pmatrix} \varepsilon \\ \vdots \\ \varepsilon \end{pmatrix}, \tag{17}$$

where $\underline{\mathsf{C}}$ is the set of all skeletons.

Now we need to lift up Equation (17) to clusters, *i.e.*, we consider factor occurrences. This is done thanks to the Flip operation. Indeed, applying the Flip operator on the words u_i and using Definition 5.9 to compute the entries $\mathsf{E}_{i,j}$ of the matrix E do not modify the skeleton. Moreover no factor occurrence can be missed in the resulting bicolored words. This implies that the set of clusters $\mathsf{C} = \mathrm{Flip}(\underline{\mathsf{C}})$ verifies Equation (15). □

5.2. Generating functions of clusters

We need now to compute the multivariate generating functions $U_i(z, \mathbf{t})$ of the bicolored words $\mathrm{Flip}(\underline{\mathsf{u}}_i)$ and $E_{i,j}(z, \mathbf{t})$ of the bicolored right extensions $\mathsf{E}_{i,j}$ to get the generating function $\xi(z, \mathbf{t})$. The following lemma gives the correspondence between bicolored decorated texts and their generating functions.

LEMMA 5.14. *We consider a pattern $\mathcal{U} = \{u_1, \ldots, u_r\}$ and a fully bicolored decorated cluster $\widetilde{\underline{\mathsf{c}}} = (c, \mathcal{D}, \mathcal{F})$ with skeleton $\underline{\mathsf{c}}$ and length $|\underline{\mathsf{c}}| = \ell$, such that $c = \alpha_1 \alpha_2 \ldots \alpha_\ell$ with $\alpha_i \in \mathcal{A}$, and where $\mathcal{D} = (\mathcal{D}_i)_{1 \le i \le \ell}$ (resp. $\mathcal{F} = (\mathcal{F}_i)_{1 \le i \le \ell}$) is the occurrence index of the distinguished occurrences defining the skeleton (resp. occurrence index of the factor occurrences), the positions being relative to the beginning of the cluster as in Equation (16). The generating function $\widetilde{\underline{\mathsf{c}}}(z, \mathbf{t})$ of the set of clusters $\mathrm{Flip}(\underline{\mathsf{c}})$ built upon the skeleton $\underline{\mathsf{c}}$ is computed thanks to the bicolored representation $\widetilde{\underline{\mathsf{c}}}$ and verifies*

$$\widetilde{\underline{\mathsf{c}}}(z, \mathbf{t}) = \prod_{i=1}^{\ell} \left[\pi(\alpha_i) z \times \left(\prod_{j \in \mathcal{D}_i} t_j \right) \times \left(\prod_{s \in \mathcal{F}_i} (1 + t_s) \right) \right], \tag{18}$$

where the variable t_i counts the occurrences of the word u_i.

PROOF. Indeed, distinguished occurrences which define the skeleton are signaled thanks to $\mathcal{D}$ whereas, once the skeleton is fixed, factor occurrences can be distinguished or not, giving for each i, if $s \in \mathcal{F}_i$, a term[7] $(1+t_s)$. □

Using the notations defined in the proof of Proposition 5.13, to compute the sequence $(U_i(z,\mathbf{t}))_{1\le i\le r}$ and the matrix $\mathbb{E}(z,\mathbf{t}) = (E_{i,j}(z,\mathbf{t}))_{1\le i,j\le r}$, we apply the last lemma and Equation (18) successively to the clusters $\mathrm{Flip}(\underline{\mathsf{u}}_i)$ and $\mathrm{Flip}(\underline{\mathsf{u}}_i \cdot \underline{\mathsf{E}}_{i,j})$; these last expressions give access to the multivariate generating functions of the bicolored sets $\mathsf{E}_{i,j}$ deriving from the sets $\mathcal{E}_{i,j}$; (it is not possible to apply directly the lemma on $\mathsf{E}_{i,j}$ since it is neither a cluster nor a skeleton). We then have

$$E_{i,j}(z,\mathbf{t}) = \frac{e(z,\mathbf{t})}{U_i(z,\mathbf{t})}, \quad \text{where } e(z,\mathbf{t}) \text{ is the generating function of } \mathrm{Flip}(\underline{\mathsf{u}}_i \cdot \underline{\mathsf{E}}_{i,j}).$$

Example 5.15. We develop further the Example 5.11 from p. 15 by taking $\pi(a) = \pi(b) = 1$.

(1) For $(u_1,u_2) = (ab, aba)$, we have

$$\begin{aligned}
\mathrm{Flip}(\mathsf{u}_1) = \{\overset{\bullet 1}{ab}\} &\mapsto \quad U_1(z,t_1,t_2) = z^2 t_1\\
\mathrm{Flip}(\mathsf{u}_2) = \{\overset{\circ 1}{a}b\overset{\bullet 2}{a}\} &\mapsto \quad U_2(z,t_1,t_2) = z^3 t_2(1+t_1)\\
\mathsf{E} = \begin{pmatrix} \emptyset & \emptyset \\ \{\overset{\bullet 2}{a}\} & \{\overset{\circ 1}{b}\overset{\bullet 2}{a}\} \end{pmatrix} &\mapsto \quad \mathbb{E}(z,t_1,t_2) = \begin{pmatrix} 0 & 0 \\ zt_2 & z^2t_2(1+t_1) \end{pmatrix}.
\end{aligned}$$

(2) For $(u_1,u_2) = (a^3, a^7)$, from Example 5.11 again, we have

$$\mathrm{Flip}(\mathsf{u}_1) \quad\mapsto\quad U_1(z,t_1,t_2) = z^3t_1, \quad \mathrm{Flip}(\mathsf{u}_2) \quad\mapsto\quad U_2(z,t_1,t_2) = z^7t_2(1+t_1)^5,$$

$$\mathbb{E}(z,t_1,t_2) = \begin{pmatrix} t_1(z+z^2) & t_2(1+t_1)^5(z^5+z^6) \\ t_1(z+z^2) & t_2\big((1+t_1)z+(1+t_1)^2z^2+(1+t_1)^3z^3+(1+t_1)^4z^4+(1+t_1)^5(z^5+z^6)\big) \end{pmatrix}.$$

(3) Finally, for $(u_1,u_2,u_3,u_4) = (aa, ab, ba, baaab)$, the last pattern of Example 5.11, we get

$$\mathbb{E}(z,t_1,t_2,t_3,t_4) = \begin{pmatrix} t_1z & t_2z & 0 & 0 \\ 0 & 0 & t_3z & t_4z^4(1+t_1)^2(1+t_2)(1+t_3) \\ t_1z & t_2z & 0 & 0 \\ 0 & 0 & t_3z & t_4z^4(1+t_1)^2(1+t_2)(1+t_3) \end{pmatrix}.$$

With these notations, we get to the following proposition.

PROPOSITION 5.16. *The generating function $\xi(z,\mathbf{t})$ of clusters built from the set $\mathcal{U} = \{u_1,\dots,u_r\}$ is given by*

$$\xi(z,\mathbf{t}) = (U_1(z,\mathbf{t}),\dots,U_r(z,\mathbf{t}))\cdot\Big(\mathbb{I} - \mathbb{E}(z,\mathbf{t})\Big)^{-1}\cdot\begin{pmatrix}1\\ \vdots\\ 1\end{pmatrix}, \tag{19}$$

where $U_i(z,\mathbf{t})$ is the generating function for the (bicolored) decorated word $\mathrm{Flip}(\mathsf{u}_i)$, and $\mathbb{E}(z,\mathbf{t})$ is the matrix of generating functions for the (bicolored) right extension sets.

PROOF. This expression follows from general principles on generating functions applied to the combinatorial description of clusters from (15). □

[7]We remark that with this methodology it would be easy to get the generating functions of clusters counting separately occurrences forming the skeleton and factor occurrences with two sets of formal variables $\mathbf{t}$ and $\mathbf{s}$ for instance.

5.3. Particular cases

We examine for special cases of interest the generating functions of clusters.

One word. For $\mathcal{U} = \{u\}$, we get

$$\xi(z,t) = \frac{t\pi(u)z^{|u|}}{1 - t\widehat{C}(z)} = \frac{t\pi(u)z^{|u|}}{1 - t(C(z)-1)}, \tag{20}$$

where $C(z)$ and $\widehat{C}(z)$ respectively are the autocorrelation polynomial and the strict autocorrelation polynomial (empty word ε omitted) of u.

Two words. For a general set of two words $\{u_1, u_2\}$, we can compute explicitly $\xi(z, t_1, t_2)$ by the Cramer's rule,

$$\xi(z,t_1,t_2) = \frac{U_1 + U_2 - \big(U_1\big[E_{2,2} - E_{1,2}\big] + U_2\big[E_{1,1} - E_{2,1}\big]\big)}{1 - E_{2,2} - E_{1,1} + \big(E_{1,1}E_{2,2} - E_{2,1}E_{1,2}\big)}, \tag{21}$$

where U_1, U_2, $(E_{i,j})_{1\le i,j\le j}$ respectively are the generating functions in (z, t_1, t_2) obtained by application of Lemma 5.14 for (bicolored) decorated words $\mathsf{u}_1, \mathsf{u}_2$ and the (bicolored) decorated matrix of right extensions $(\mathsf{E}_{i,j})_{1\le i,j\le j}$.

Example 5.17. Let us consider again the pattern $\mathcal{U} = \{a^3, a^7\}$ from Example 5.15(2) p. 18 and $\pi(a) = \pi(b) = 1$; we therefore have $\pi(w) = 1$ for all words w (the unweighted "enumerative" model where each word has weight 1). Evaluating Equation (21) with these weights, the generating function of clusters $\xi(z, t_1, t_2)$ verifies

$$\xi(z,t_1,t_2) = \frac{z^3\left(t_2(1+t_1)^4z^4 - t_2t_1(1+t_1)z^3 - t_2t_1(1+t_1)^2z^2 - t_2t_1(1+t_1)z + t_1\right)}{\left(1 - z^3t_2(1+t_1)\left((1+t_1)^3z^3 + \left({t_1}^3 + 3\,{t_1}^2 + 2\,t_1 + 1\right)z^2 + \left({t_1}^3 + 2\,{t_1}^2 + t_1 + 1\right)z + 1 - {t_1}^2 - 2\,t_1\right) - (t_2t_1 + t_2 + t_1)(z^2 + z)\right)}$$

Reduced set. When the set $\mathcal{U}$ is reduced, that is, no word of $\mathcal{U}$ is factor of another, we do not need to consider the Flip operation as clusters are always reduced. The distinguished occurrences are implicitly specified. So that the generating function can be stated with the help of the generating function of the correlation set

$$\xi(z,\mathbf{t}) = \Delta(\mathbf{t})(\pi(u_1)z^{|u_1|}, \ldots, \pi(u_r)z^{|u_r|}) \ \left(\mathbb{I} - \Delta(\mathbf{t})\widehat{\mathbb{C}}(z)\right)^{-1} \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix},$$

where $\Delta(\mathbf{t})$ is a square matrix with diagonal $\mathbf{t} = (t_1, \ldots, t_r)$ and $\widehat{\mathbb{C}}(z)$ is the matrix of generating functions for strict (*i.e.*, empty word omitted) correlation sets. This is another formulation of the result of Goulden and Jackson [Goulden and Jackson 1983].

Generating function of texts. As mentioned in Section 4.1.2, a text is decomposed as a sequence of letters from $\mathcal{A}$ (with generating function $A(z)$) and clusters from $\mathsf{C}_{\mathcal{U}}$ (with generating function $\xi(z, \mathbf{t})$ from Equation (6)). The multivariate generating function $F(z, \mathbf{x})$ given by Equation (1) is derived by substituting $t_i \mapsto x_i - 1$ for $i \in \{1, \ldots, r\}$. To summarize, we have the following theorem.

THEOREM 5.18. *Let $\mathbf{u} = (u_1, \ldots, u_r)$ be a finite vector of words in $\mathcal{A}^\star$ and $\mathcal{E}$ the associated right extension matrix. The multivariate generating function $F(z, \mathbf{x})$ counting texts whose length is counted by the variable z and where the occurrences of u_i are counted by the vector of formal variables $\mathbf{x} = (x_1, \ldots, x_r)$ is*

$$F(z,\mathbf{x}) = \frac{1}{1 - A(z) - \xi(z, \mathbf{x} - \mathbf{1})}, \tag{22}$$

where $A(z) = \sum_{\alpha\in\mathcal{A}} \pi(\alpha)z$ is the generating function of the alphabet and $\xi(z, \mathbf{t})$ is defined in Equation (19).

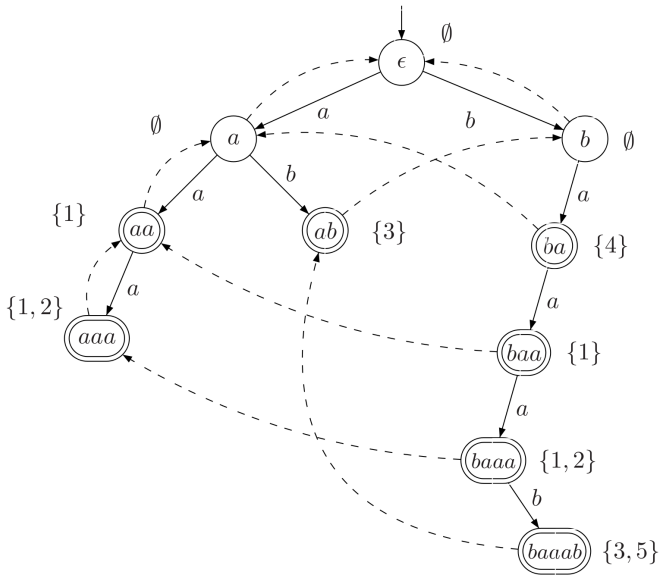


Fig. 4. Illustration of the Aho-Corasick construction, considering the pattern $\mathcal{U} = \{aa, aaa, ab, ba, baaab\}$ of Example 5.15(3). Nodes are labeled with words from $\mathrm{Pref}(\mathcal{U})$. Indices are to be understood (following the convention used in all this article) as indices of words in $\mathcal{U}$ ordered in the lexicographical order. Double circled terminal nodes correspond to states where an occurrence of $\mathcal{U}$ is found. However, for our particular purpose, we need a more precise information and associate to each node w the set $\mathcal{S}_w$ which is the set of labels of words from $\mathcal{U}$ accepted at state w. These sets are represented next to each state. Suffix links are represented with dashed lines.

PROOF. The proof relies on two main points. On one hand, the generating function $\xi(z, \mathbf{t})$ counts all the clusters (see Proposition 5.16). On the other hand, the inclusion-exclusion principle yields the final result by the substitutions $t_i \mapsto x_i - 1$. □

The application of the standard techniques of analytic combinatorics (see [Flajolet and Sedgewick 2009]) to the multivariate generating function $F(z, \mathbf{x})$ gives access to statistics such as mean, variance, covariance. (See Section 6).

5.4. An algorithmic construction based on the Aho-Corasick automaton

We consider a pattern $\mathcal{U} = \{u_1, \ldots, u_r\}$. We recall that we assumed previously that $u_i \prec u_{i+1}$ with respect to the lexicographical order for i from 1 to $r - 1$; therefore k is the index of the word u_k in the pattern considered as a list.

We want an efficient way to compute the generating function $\xi(z, \mathbf{t})$ of the clusters of $\mathcal{U}$. In particular, we have to compute the r-tuple $(U_1(z, \mathbf{t}), \ldots, U_r(z, \mathbf{t}))$ and the $r \times r$ elements $E_{i,j}(z, \mathbf{t})$ of the right extension matrix from Proposition 5.16.

As mentioned in Section 3 the Aho-Corasick algorithm upon which we work first constructs a trie $\mathcal{T}_\mathcal{U}$ on the words of the pattern $\mathcal{U}$. We denote Ω the Aho-Corasick automaton constructed upon the trie $\mathcal{T}_\mathcal{U}$.

We *label* and *name* in the following any state of the automaton Ω by the word that leads to this state when starting from the root ε and progressing in the trie $\mathcal{T}_\mathcal{U}$ by successively reading the letters of w; see an example of such labeling in Figure 4. Therefore the term w will refer in a parallel manner to a word w and to the corresponding state in the automaton Ω.

We use the following definitions.

— We denote $\mathrm{Pref}(w)$ the set of prefixes of a word w.

— We denote $\mathcal{S}_w$ the set of indices of words from $\mathcal{U}$ that are suffixes of w; this is considered in a wide sense: we accept the word w as a suffix of itself, although it is not a proper suffix.
— We denote $\lambda(w)$ the suffix link starting from state w, where

$$\lambda(w) = \mathrm{Border}(w),$$

and the function $\mathrm{Border}(w)$ is defined as in Section 3 by

$$\mathrm{Border}(w) = \begin{cases} \text{the longest proper suffix of } w \text{ in } \mathrm{Pref}(\mathcal{U}) \text{ if it is defined,} \\ \text{or } \varepsilon \text{ otherwise.} \end{cases}$$

As an example, considering Figure 4, there is a suffix link from the state $(baaab)$ to the state (ab).
— We denote $\sigma(w)$ the length of the suffix chain[8] starting at a state w.

We proceed in the following by steps.

— *Step (i).* We construct the Aho-Corasick automaton Ω recognizing the pattern $\mathcal{U}$.
— *Step (ii).* We associate to each state w of the automaton the set $\mathcal{S}_w$ containing the indices of words u from $\mathcal{U}$ that are recognized at state w (so that these words are also suffixes of w); beware that along the cases, these words may be or may be not factor occurrences.
— *Step (iii).* By using the suffix links and the information stored in Step (ii), we compute all the needed generating functions.

We detail now each step of the algorithmic computation.

Step (i). This step is a classical construction for text analysis (see for instance [Crochemore et al. 2007; Crochemore and Rytter 2002]). Note that the construction provides the suffix links for the pattern.

Step (ii). We add now more information to the automaton Ω.

The set $\mathcal{S}_w$ can be obtained while building the automaton, since when a state w is terminal, we know which of the words of the pattern $\mathcal{U}$ is accepted. This is a classical modification of the basic Aho-Corasick algorithm (see [Crochemore et al. 2007] for instance). The complexity of this modification is $O(r \times \sum_{u\in\mathcal{U}} |u|)$ if we naively manage subsets $\mathcal{S}_w$ of $\{1, \ldots, r\}$ for each state w of the automaton.

Step (iii). Using the suffix links, we get an alternative way to express the right extension set from a word u_i to a word u_j (see Definition 5.8); we have

$$\mathcal{E}_{i,j} = \Big\{ e \mid \quad h \cdot e = u_j \quad \text{and} \quad h \in \{\lambda^c(u_i),\ 1 \le c < \sigma(u_i)\} \Big\}. \tag{23}$$

Note that $h \cdot e = u_j \Rightarrow h \in \mathrm{Pref}(u_j)$; this leads to consider the sets $H_{i,j}$ such that

$$H_{i,j} = \Big\{ h \neq \varepsilon \quad \Big| \quad h \ \in \mathrm{Pref}(u_j) \cap \big\{\lambda^c(u_i) \mid 1 \le c < \sigma(u_i)\big\} \Big\}. \tag{24}$$

Each word h in this equation is simultaneously a prefix of u_j and a suffix of u_i (by the definition of the suffix link function $\lambda(w)$); moreover the set $H_{i,j}$ is in bijection with the right extension set $\mathcal{E}_{i,j}$.

[8]The suffix chain of a word u is the sequence $(u_1 = u, u_2 = \mathrm{Border}(u_1), u_3 = \mathrm{Border}(u_2), \ldots, u_s = \mathrm{Border}(u_{s-1}) = \varepsilon)$. The length of this chain is $\sigma(u) = s - 1$.

We now define, for states $w \in \mathrm{Pref}(u_j) \cup \{\varepsilon\}$ such that there exists $v \in \mathcal{A}^*$ that verifies $w \cdot v = u_j$, the auxiliary functions $\Phi_j^{(w)}(z, \mathbf{t}) = \mathsf{v}(z, \mathbf{t})$, where v is the decorated text associated to the word v. We have immediately

$$U_j(z, \mathbf{t}) = \Phi_j^{(\varepsilon)}(z, \mathbf{t}), \quad \text{and} \quad E_{i,j}(z, \mathbf{t}) = \sum_{h \in H_{i,j}} \Phi_j^{(h)}(z, \mathbf{t}). \tag{25}$$

The generating functions $\Phi_j^{(w)}(z, \mathbf{t})$ can be defined recursively,

$$\Phi_j^{(w)}(z, \mathbf{t}) = \begin{cases} t_j & \text{if } w = u_j \\ \Phi_j^{(w\cdot\alpha)}(z, \mathbf{t}) \times \pi(\alpha) z \prod_{s \in \mathcal{S}_{w\cdot\alpha} \setminus \{j\}} (1 + t_s) & \text{if } w\cdot\alpha \in \mathrm{Pref}(u_j)\ (\alpha \in \mathcal{A}). \end{cases} \tag{26}$$

Thus all the functions $\Phi_j^{(w)}(z, \mathbf{t})$ for $j \in \{1, \ldots, r\}$ and $w \in \mathrm{Pref}(\mathcal{U})$ are computed from the leaves to the root by a postorder traversal of the trie $\mathcal{T}_{\mathcal{U}}$;

Example 5.19. We consider the example given in Figure 4 where we have $\mathcal{U} = \{aa, aaa, ab, ba, baaab\}$. Following the recurrence defined in Equation (26), we have, recalling that 2 is the index of the word aaa in the pattern $\mathcal{U}$, and since $H_{2,2} = \{aa, a\}$,

$$\begin{aligned} &\Phi_2^{(aaa)}(z, \mathbf{t}) = t_2, && \Phi_2^{(aa)}(z, \mathbf{t}) = t_2\pi(a)z(1+t_1), \\ &\Phi_2^{(a)}(z, \mathbf{t}) = t_2\pi^2(a)z^2(1+t_1)^2, && \Phi_2^{(\varepsilon)}(z, \mathbf{t}) = t_2\pi^3(a)z^3(1+t_1)^2 \\ &U_2(z, \mathbf{t}) = t_2\pi^3(a)z^3(1+t_1)^2, && E_{2,2}(z, \mathbf{t}) = t_2\Big(\pi(a)z(1+t_1)+\pi^2(a)z^2(1+t_1)^2\Big). \end{aligned}$$

Similarly, starting from $\Phi_5^{(baaab)} = t_5$, we obtain by using the recurrence

$$E_{3,5}(z, \mathbf{t}) = \Phi_5^{(a)}(z, \mathbf{t}) = \pi(a)^3\pi(b)z^4(1 + t_1)^2(1 + t_2)(1 + t_3)(1 + t_4)t_5,$$

which follows from the fact that the set $H_{3,5} = \{b\}$ has a single element.
Starting now from $\Phi_4^{(ba)}(z, \mathbf{t}) = t_4$, we get $E_{3,4}(z, \mathbf{t}) = \Phi_4^{(b)}(z, \mathbf{t}) = \pi(a)zt_4$.

Complexity. The classical construction of the Aho-Corasick automaton yields a time complexity $O(\sum_{u\in\mathcal{U}} |u|)$. However we need more information on terminal nodes (namely the set of indices of words accepted), and this gives a complexity $O(r \times \sum_{u\in\mathcal{U}} |u|)$ since we manipulate subsets of $\{1, \ldots, r\}$.

We denote by S the size of the longest suffix chain of a word $u \in \mathcal{U}$. The number of the sets $H_{i,j}$ defined in Equation (23) is typically less than $r \times r$. To compute them, we can use sets $\mathcal{P}_w$ associated to each state w of $\mathcal{T}_{\mathcal{U}}$ that record the indices of the words $u \in \mathcal{U}$ such that $w \in \mathrm{Pref}(u)$; for instance, considering Figure 4, we have $\mathcal{P}_a = \{1, 2, 3\}$. The computation of the sets $\mathcal{P}_w$ can be done by a postorder traversal of the tree in time $O(r \times |\Omega|) = O(r \times \sum_{u\in\mathcal{U}} |u|)$. Using these sets and the suffix links, all the sets $H_{i,j}$ can be computed in a largely overestimated overall cost $r \times r \times S$. It is then straightforward to compute $E_{i,j}(z, \mathbf{t})$ by Equation (25).

The auxiliary functions $\Phi_j^{(w)}(z, \mathbf{t})$ for $1 \le j \le r$ and w a prefix of u_j are computed, considering here operations on polynomials (mostly multiplications), in total time $O(r \times \sum_{u\in\mathcal{U}} |u|)$.

As a conclusion of this section, assuming that the size of the alphabet is a constant, the "time complexity", considering elementary operations on automata and operations on polynomials in z and $\mathbf{t}$, is $O(r \times \sum_{u\in\mathcal{U}} |u| + S \times r^2)$ in order to compute the sequence $(U_i(z, \mathbf{t}))_{i=1}^r$ and the matrix $\mathbb{E}(z, \mathbf{t})$. We remark that the coefficients of the matrix are polynomials whose degrees (in any variable) are bounded by $\max_{u\in\mathcal{U}} |u| - 1$. Also we note that the $r \times r$ matrix $\mathbb{E}(z, \mathbf{t})$ is smaller and more compact than the linear system

obtained by applying the Chomsky-Schützenberger algorithm on the Aho-Corasick automaton of Section 3 which has size $\mathcal{O}((\sum_{u\in\mathcal{U}}|u|)^2)$ since there are $O\left(\sum_{u\in\mathcal{U}}|u|\right)$ states in the automaton. Inverting the corresponding sparse matrix in z would imply handling possibly large coefficients that are multivariate polynomials over the counting variables $x_1,\ldots,x_r$; see Section 3 for an example.

6. MOMENTS

Bender and Kochman [Bender and Kochman 1993] consider generalized words where a generalized word $\mathcal{W}$ is a set of words of same length. Considering two generalized words $\mathcal{W}_1$ and $\mathcal{W}_2$, they compute the dominant term of the asymptotic covariance of the number of occurrences of $\mathcal{W}_1$ and $\mathcal{W}_2$ in random texts of size n as n tends to infinity. We consider as previously patterns $\mathcal{U}$ that are *any* finite sets of finite words, disregarding their lengths. We compute here in random texts of size n the dominant asymptotic term of:

— the expectation of the number of occurrences of a pattern $\mathcal{U}$;
— the variance of this number;
— the covariance of the number of occurrences of a pattern $\mathcal{U}$ and of a pattern $\mathcal{V}$.

Bender and Kochman also state limit laws in several cases for sets of generalized words, their proofs relying importantly on previous works of Bender *et al* in a series of articles [Bender 1973; Bender and Richmond 1983; Bender et al. 1983]. Although it is very likely that these results still hold in the case of more general patterns, the corresponding study is beyond the scope of the present article.

6.1. Number of occurrences for a non-reduced pattern

We consider here a Bernoulli model and extend the probability measure π defined for words to sets of words in the following way

$$\pi(\mathcal{U}) = \sum_{u\in\mathcal{U}} \pi(u).$$

We provide for the case of a pattern $\mathcal{U} = \{u_1,\ldots,u_k\}$ expressions for the expected value and variance of the random variable X_n counting the number of occurrences of $\mathcal{U}$ in a text of size n. Section 5 gives a mean to obtain the generating function for clusters $\xi(z,t_1,\ldots,t_k)$ where the occurrences of the word $u_i\in\mathcal{U}$ are counted by the variable t_i. The cluster generating function $\Upsilon(z,t)$ related to occurrences of $\mathcal{U}$ is then defined[9] by

$$\Upsilon(z,t) = \xi(z,t,\ldots,t). \tag{27}$$

Finally the generating function of occurrences is by Equation (22)

$$F(z,x) = \frac{1}{1-z-\Upsilon(z,x-1)},$$

[9]Doing the substitutions $t_i\to t$, we do not count the number of *positions* where there is a match, but add up the number of matches with words of $\mathcal{U}$ at each matching position.

and, since $F(z,1) = 1/(1-z)$, we have $\Upsilon(z,0) = 0$. Setting $\Upsilon_t(z) = \frac{\partial}{\partial t}\Upsilon(z,t)\big|_{t=0}$ and $\Upsilon_{tt}(z) = \frac{\partial^2}{\partial t^2}\Upsilon(z,t)\Big|_{t=0}$ and using basic algebra, we have

$$\sum_{n\geq 0}\mathbf{E}[X_n]z^n = \frac{\partial}{\partial x}F(z,x)\Big|_{x=1} = \frac{\Upsilon_t(z)}{(1-z)^2}$$
$$\sum_{n\geq 0}\mathbf{E}[X_n^2]z^n = \frac{\partial^2}{\partial x^2}F(z,x)\Big|_{x=1} + \frac{\partial}{\partial x}F(z,x)\Big|_{x=1} = \frac{2\Upsilon_t(z)^2}{(1-z)^3} + \frac{\Upsilon_{tt}(z)+\Upsilon_t(z)}{(1-z)^2}.$$

It is easy to see that $\Upsilon_t(z) = \sum_{u\in\mathcal{U}}\pi(u)z^{|u|}$ (the clusters with one and only one marked occurrence). The expression for $\Upsilon_{tt}(z)$ takes into account that some words of $\mathcal{U}$ are factor of other ones

$$\Upsilon_{tt}(z) = 2\sum_{\substack{u,v\in\mathcal{U}\\ u\neq v}}\pi(u)|u|_v z^{|u|} + 2\sum_{u,v\in\mathcal{U}}\sum_{e\in\mathcal{E}_{u,v}}\pi(ue)z^{|ue|}.$$

After some algebra, we get the following result.

PROPOSITION 6.1. *Let $\mathcal{U} = \{u_1,\ldots,u_k\}$ be a pattern. The expected value and the variance of the variable X_n counting the number of occurrences of $\mathcal{U}$ in a random text of size n satisfy*

$$\mathbf{E}[X_n] = \sum_{u\in\mathcal{U}}\pi(u)(n-|u|+1),$$
$$\frac{1}{n}\mathbf{Var}[X_n] = \pi(\mathcal{U}) - \sum_{u,v\in\mathcal{U}}\pi(u)\pi(v)(|u|+|v|-1) + 2\sum_{u,v\in\mathcal{U}}\pi(u)\pi(\mathcal{E}_{u,v}) + 2\sum_{\substack{u,v\in\mathcal{U}\\ u\neq v}}\pi(u)|u|_v + o(1).$$

We point out that the last sum is a correcting factor and is non zero only if the set is non reduced.

If the set contains only one word u, recalling that $\mathcal{E}_{u,u}$ is the *strict* autocorrelation set of u, we obtain (as we should!) the classical result for the variance (see by instance Theorem 7.14 in [Szpankowski 2001] book)

$$\lim_{n\to\infty}\frac{1}{n}\mathbf{Var}(X_n) = \pi(u) + 2\pi(u)\pi(\mathcal{E}_{u,u}) - (2|u|-1)\pi(u)^2. \tag{28}$$

6.2. Covariance of two patterns

We consider again a Bernoulli model and two patterns $\mathcal{U}$ and $\mathcal{V}$. We use again the notion of right extensions sets introduced in this article. The following theorem extends the case handled by [Bender and Kochman 1993] where $\mathcal{U}$ and $\mathcal{V}$ are generalized words.

THEOREM 6.2. *Let $\mathcal{U} = \{u_1,\ldots,u_k\}$ and $\mathcal{V} = \{v_1,\ldots,v_j\}$ be two patterns. The covariance of the variables X_n and Y_n counting respectively the number of occurrences of*

$\mathcal{U}$ and $\mathcal{V}$ in a random text of size n verifies

$$\frac{1}{n}\,\mathbf{Cov}(X_n, Y_n) = \pi(\mathcal{U}\cap\mathcal{V}) - \sum_{u\in\mathcal{U},v\in\mathcal{V}} \pi(u)\pi(v)\big(|u|+|v|-1\big) + \sum_{u\in\mathcal{U},v\in\mathcal{V}} \Big(\pi(u)\pi(\mathcal{E}_{u,v}) + \pi(v)\pi(\mathcal{E}_{v,u})\Big) + \sum_{\substack{u\in\mathcal{U},v\in\mathcal{V}\\ u\neq v}} \Big(|u|_v\,\pi(u) + |v|_u\,\pi(v)\Big) + o(1). \tag{29}$$

PROOF. We consider here the weighted case where $A(z) = z$. Let $\mathcal{U}$ and $\mathcal{V}$ be two sets of words. We first decompose as a direct sum the set $\mathcal{U}\cup\mathcal{V}$:

$$\mathcal{U}\cup\mathcal{V} = (\mathcal{U}\setminus\mathcal{V})\oplus(\mathcal{V}\setminus\mathcal{U})\oplus(\mathcal{U}\cap\mathcal{V}).$$

In order to ease the notations, we index the variables in the generating function $\xi(z,\mathbf{t})$ by words, *i.e.*, the variable t_u corresponds to the word u. Then we consider the generating function of clusters for the three disjoint sets $\mathcal{U}' = \mathcal{U}\setminus\mathcal{V}$, $\mathcal{V}' = \mathcal{V}\setminus\mathcal{U}$, and $\mathcal{W} = \mathcal{U}\cap\mathcal{V}$, with $\mathbf{t} = (t_u)_{u\in\mathcal{U}\cup\mathcal{V}}$ and the respective variables t_1, t_2 and t_3 such that

$$\Upsilon(z,t_1,t_2,t_3) = \xi(z,\mathbf{t})\big|_{\substack{t_u = t_1 \text{ for } u\in\mathcal{U}\setminus\mathcal{V}\\ t_u = t_2 \text{ for } u\in\mathcal{V}\setminus\mathcal{U}\\ t_u = t_3 \text{ for } u\in\mathcal{U}\cap\mathcal{V}}}; \tag{30}$$

that is we simply substitute variables for words appearing in each of the three sets with t_1, t_2 and t_3.

Let $F(z,x,y)$ be the corresponding generating function counting occurrences. We have by Equation (22) and since occurrences in $\mathcal{U}\cap\mathcal{V}$ are marked two times (one x for belonging to $\mathcal{U}$ and one y for belonging to $\mathcal{V}$)

$$F(z,x,y) = \frac{1}{1 - z - \Upsilon(z, x-1, y-1, xy-1)}. \tag{31}$$

By construction, since $F(z,1,1) = \frac{1}{1-z}$, one has $\Upsilon(z,0,0,0) = 0$. To simplify the notations, we set

$$\Upsilon_i(z) = \tfrac{\partial}{\partial t_i}\Upsilon(z,t_1,t_2,t_3)\Big|_{(t_1,t_2,t_3)=(0,0,0)} \text{ for } i\in\{1,2,3\}$$
$$\Upsilon_{ij}(z) = \tfrac{\partial^2}{\partial t_i\partial t_j}\Upsilon(z,t_1,t_2,t_3)\Big|_{(t_1,t_2,t_3)=(0,0,0)} \text{ for } i,j\in\{1,2,3\}.$$

By general mechanisms [Flajolet and Sedgewick 2009] we get from Equation (31)

$$\sum_{n\geq 0}\mathbf{E}(X_n)z^n = \tfrac{\partial}{\partial x}F(z,x,y)\big|_{x=y=1} = \frac{1}{(1-z)^2}\left(\Upsilon_1(z) + \Upsilon_3(z)\right)$$
$$\sum_{n\geq 0}\mathbf{E}(Y_n)z^n = \tfrac{\partial}{\partial y}F(z,x,y)\big|_{x=y=1} = \frac{1}{(1-z)^2}\left(\Upsilon_2(z) + \Upsilon_3(z)\right),$$

which gives

$$\mathbf{E}(X_n) = \sum_{u\in\mathcal{U}}(n-|u|+1)\pi(u), \qquad \mathbf{E}(Y_n) = \sum_{u\in\mathcal{V}}(n-|u|+1)\pi(u).$$

We have also easy access to the covariance since

$$\begin{aligned}\sum_{n\geq 0}\mathbf{E}(X_nY_n)z^n &= \tfrac{\partial^2}{\partial x\partial y}F(z,x,y)\Big|_{x=y=1}\\ &= 2\tfrac{(\Upsilon_1(z)+\Upsilon_3(z))(\Upsilon_2(z)+\Upsilon_3(z))}{(1-z)^3}+\tfrac{\Upsilon_{12}(z)+\Upsilon_{13}(z)+\Upsilon_{23}(z)+\Upsilon_{33}(z)+\Upsilon_3(z)}{(1-z)^2}\\ &= 2\tfrac{\Upsilon_1(z)\Upsilon_2(z)}{(1-z)^3}+\tfrac{\Upsilon_{12}(z)}{(1-z)^2}+2\tfrac{\Upsilon_3(z)^2}{(1-z)^3}+\tfrac{\Upsilon_{33}(z)+\Upsilon_3(z)}{(1-z)^2}\\ &\qquad +2\tfrac{\Upsilon_3(z)(\Upsilon_1(z)+\Upsilon_2(z))}{(1-z)^3}+\tfrac{\Upsilon_{13}(z)+\Upsilon_{23}(z)}{(1-z)^2}.\end{aligned}$$

A Taylor expansion at $z=1$ gives for $i=1,2,3$ (with $f'(z)=\frac{\partial}{\partial z}f(z)$)

$$\Upsilon_i(z)=\Upsilon_i(1)-(1-z)\Upsilon_i'(1)+o(1-z).$$

Hence we get

$$\begin{aligned}\mathbf{E}(X_nY_n) = (n+1)(n+2)&\big(\Upsilon_1(1)+\Upsilon_3(1)\big)\big(\Upsilon_2(1)+\Upsilon_3(1)\big)\\ &+(n+1)\Big(\big(\Upsilon_1'(1)+\Upsilon_3'(1)\big)\big(\Upsilon_2(1)+\Upsilon_3(1)\big)\\ &\quad+\big(\Upsilon_1(1)+\Upsilon_3(1)\big)\big(\Upsilon_2'(1)+\Upsilon_3'(1)\big)\\ &\qquad+\Upsilon_{12}(1)+\Upsilon_{13}(1)+\Upsilon_{23}(1)+\Upsilon_{33}(1)+\Upsilon_3(1)\Big)+o(n).\end{aligned}$$

Now we can interpret each of the coefficients: we have for instance

$$\begin{aligned}\Upsilon_1(1)+\Upsilon_3(1)&=\sum_{u\in\mathcal{U}}\pi(u),\\ \Upsilon_{13}(1)&=\sum_{u\in\mathcal{U}\setminus\mathcal{V}}\sum_{v\in\mathcal{V}\cap\mathcal{U}}\big(\pi(u)|u|_v+\pi(u)\pi(\mathcal{E}_{u,v})+\pi(v)|v|_u+\pi(v)\pi(\mathcal{E}_{v,u})\big).\end{aligned}$$

Summarizing and after some computations, we get to Equation (29). □

Example: variance-covariance matrix of a^3 and a^7. We apply the results of this section to $\mathcal{U}=a^3$ and $\mathcal{V}=a^7$ in a Bernoulli model with $p=\pi(a)$.

Let X_n and Y_n count the number of occurrences of a^3 and a^7 in a random text of size n. We denote

$$\mathbb{B}_{11}=\lim_{n\to\infty}\frac{1}{n}\mathbf{Var}(X_n),\ \mathbb{B}_{22}=\lim_{n\to\infty}\frac{1}{n}\mathbf{Var}(Y_n),\ \mathbb{B}_{12}=\mathbb{B}_{21}=\lim_{n\to\infty}\frac{1}{n}\mathbf{Cov}(X_n,Y_n).$$

We get by Equation (29),

$$\left\{\begin{array}{l}\pi(a^3)\pi(\mathcal{E}_{a^3,a^7})=p^3(p^5+p^6),\\ \pi(a^7)\pi(\mathcal{E}_{a^7,a^3})=p^7(p+p^2),\\ \big|a^7\big|_{a^3}\pi(a^7)=5p^7,\\ \big|a^3\big|_{a^7}\pi(a^3)=0,\\ (\big|a^7\big|+\big|a^3\big|-1)\pi(a^7)\pi(a^3)=9p^{10}\end{array}\right|\qquad\Longrightarrow\qquad \mathbb{B}_{12}=5p^7+2p^8+2p^9-9p^{10}.$$

Computing the variance-covariance matrix $\mathbb{B}^{(a^3,a^7)} = (\mathbb{B}_{ij})$ for $i,j \in \{1,2\}$, and the corresponding determinant Δ, we get

$$\mathbb{B}^{(a^3,a^7)} = \begin{pmatrix} p^3+2p^3(p+p^2)-5p^6 & p^7(5+2p+2p^2-9p^3) \\ p^7(5+2p+2p^2-9p^3) & p^7+2p^7(p+p^2+p^3+p^4+p^5+p^6)-13p^{14} \end{pmatrix},$$

$$\Delta = |\mathbb{B}^{(a^3,a^7)}| = p^{10} + 4p^{11} + 8p^{12} + 5p^{13} - 25p^{14} - 20p^{15} - 24p^{16} + 67p^{17} - 16p^{20}.$$

Let us remark that Δ is zero if $p = 0$ or $p = 1$ and nowhere else. This corresponds to a degeneracy of the system that can also be observed by using [Bender and Kochman 1993] constructions;

— if $p = 0$ the background texts have no letters a and both counts of a^3 and a^7 are zero.
— if $p = 1$ the texts are sequences of a. In such texts of length larger than 7, the number of counts of a^3 exceeds exactly by 4 the number of counts of a^7.

In all other cases ($p \neq 0$ and $p \neq 1$), the results of [Bender and Kochman 1993] imply as limit a Gaussian law of dimension 2 for the two joint counts.

Conclusion and perspectives

We obtained a detailed proof and an explicit expression of the multivariate generating function counting texts according to their length and to their number of occurrences of words from a finite set. This result facilitates access to various moments and may lead to limiting distributions. From Bender and Kochman [Bender and Kochman 1993], we expect to find mostly a multivariate normal law for word counts. Our approach can possibly provide simpler criteria to decide if such a limiting law holds or not. Another nice aspect of the inclusion-exclusion approach is that it provides explicit formulæ like Equation (21) p. 19, whereas the Aho-Corasick construction does not give immediate access to the structure of correlations of the words; this can be a crucial advantage when looking for second moments of structures such as suffix-trees.

Ongoing work is more concerned with the complexity of the diverse approaches presented in this article. Also we plan to extend the analysis to more complex sources, such as Markovian or dynamical sources (see Vallée [Vallée 2001]).

Acknowledgments. The authors thank Jérémie Bourdon, Philippe Dumas, and Julien Fayolle for fruitful discussions and for providing important feedback.

REFERENCES

AHO, A. AND CORASICK, M. 1975. Efficient String Matching: An Aid to Bibliographic Search. *Communications of the ACM 18*, 333–340.

BENDER, E. 1973. Central and local limit theorems applied to asymptotic enumeration. *Journal of Combinatorial Theory 15*, 91–111.

BENDER, E. AND KOCHMAN, F. 1993. The distribution of subword counts is usually normal. *European Journal of Combinatorics 14*, 265–275.

BENDER, E. AND RICHMOND, B. 1983. Central and local limit theorems applied to asymptotic enumeration II: Multivariate Generating Functions. *Journal of Combinatorial Theory Series A,* 34, 255–265.

BENDER, E., RICHMOND, B., AND WILLIAMSON, G. 1983. Central and local limit theorems applied to asymptotic enumeration. III. Matrix recursions. *Journal of Combinatorial Theory 35,* 3, 264–278.

BOURDON, J. AND VALLÉE, B. 2002. Generalized pattern matching statistics. In *Proc. Colloquium on Mathematics and Computer Science: Algorithms, Trees, Combinatorics and Probabilities*. Birkhauser, Trends in Mathematics. 249–265.

BOURDON, J. AND VALLÉE, B. 2006. Pattern matching statistics on correlated sources. In *Proc. of LATIN'06*. LNCS Series, vol. 3887. Springer, 224–237.

CHOMSKY, N. AND SCHÜTZENBERGER, M. 1963. The algebraic theory of context-free languages. *Computer Programming and Formal Languages*, 118–161. P. Braffort and D. Hirschberg, eds, North Holland.

CROCHEMORE, M., HANCART, C., AND LECROQ, T. 2007. *Algorithms on Strings*. Cambridge University Press. 392 pages.

CROCHEMORE, M. AND RYTTER, W. 2002. *Jewels of Stringology*. World Scientific Publishing, Hong-Kong. 310 pages.

EDLIN, A. AND ZEILBERGER, D. 2000. The Goulden-Jackson method for cyclic words. *Advances in Applied Mathematics 25,* 2, 228–232.

FLAJOLET, P. AND SEDGEWICK, R. 2009. *Analytic Combinatorics*. Cambridge University Press.

GOULDEN, I. AND JACKSON, D. 1979. An inversion theorem for clusters decompositions of sequences with distinguished subsequences. *J. London Math. Soc. 2,* 20, 567–576.

GOULDEN, I. AND JACKSON, D. 1983. *Combinatorial Enumeration*. John Wiley. New-York.

GUIBAS, L. AND ODLYZKO, A. 1981a. Periods in strings. *J. Combin. Theory A,* 30, 19–42.

GUIBAS, L. AND ODLYZKO, A. 1981b. Strings overlaps, pattern matching, and non-transitive games. *J. Combin. Theory A,* 30, 108–203.

JACQUET, P. AND SZPANKOWSKI, W. 1994. Autocorrelation on words and its applications. Analysis of Suffix Trees by String Ruler Approach. *J. Combin. Theory A,* 66, 237–269.

KONG, Y. 2005. Extension of Goulden-Jackson cluster method on pattern occurrences in random sequences and comparison with Régnier Szpankowski method. *J. of Difference Equations and Applications 11,* 15, 1265–1271.

KUPIN, E. AND YUSTER, D. 2010. Generalizations of the Goulden-Jackson method. *Journal of Difference Equations and Applications 16,* 12, 1463–1480.

LOTHAIRE, M. 2005. *Applied Combinatorics on Words*. Encyclopedia of Mathematics. Cambridge University Press.

NICODÈME, P. 2003. Regexpcount, a symbolic package for counting problems on regular expressions and words. *Fundamenta Informaticae 56,* 1-2, 71–88.

NICODÈME, P., SALVY, B., AND FLAJOLET, P. 2002. Motif statistics. *Theoretical Computer Science 287,* 2, 593–618.

NOONAN, J. 1998. New Upper Bounds for the Connective Constants of Self-Avoidings Walks. *Journal of Statistical Physics 91,* 5/6, 871–888.

NOONAN, J. AND ZEILBERGER, D. 1999. The Goulden-Jackson Method: Extensions, Applications and Implementations. *J. of Difference Equations and Applications 5,* 4-5, 355–377.

PRUM, B., RODOLPHE, F., AND DE TURCKHEIM, E. 1995. Finding words with unexpected frequencies in deoxyribonucleic acid sequences. *J. R. Statist. Soc. B 57,* 1, 205–220.

RÉGNIER, M. 2000. A unified approach to word occurrences probabilities. *Discrete Applied Mathematics 104,* 1, 259–280. Special issue on Computational Biology.

RÉGNIER, M. AND SZPANKOWSKI, W. 1997. On the approximate pattern occurrences in a text. In *SEQUENCES '97: Proceedings of the Compression and Complexity of Sequences 1997*. IEEE Computer Society, Washington, DC, USA, 253.

RÉGNIER, M. AND SZPANKOWSKI, W. 1998. On Pattern Frequency Occurrences in a Markovian Sequence. *Algorithmica 22,* 4, 631–649.

REINERT, G. AND SCHBATH. 1998. Compound Poisson and Poisson process approximations for occurrences of multiple words in Markov chains. *J. Comp. Biol. 5*, 223–253.

REINERT, G., SCHBATH, S., AND WATERMAN, M. 2000. Probabilistic and statistical properties of words: an overview. *J. Comp. Biol. 7*, 1–46.

ROQUAIN, E. AND SCHBATH, S. 2007. Improved compound Poisson approximation for the number of occurrences of multiple words in a stationary Markov chain. *Adv. Appl. Prob.* 39, 1–13.

SEDGEWICK, R. AND FLAJOLET, P. 1996. *An Introduction to the Analysis of Algorithms*. Addison-Wesley Publishing Company.

SZPANKOWSKI, W. 2001. *Average Case Analysis of Algorithms on Sequences*. Series in Discrete Mathematics and Optimization. John Wiley & Sons.

VALLÉE, B. 2001. Dynamical sources in information theory: Fundamental Intervals and Word Prefixes. *Algorithmica 29,* 1, 262–306.

WEN, X. 2005. The symbolic Goulden-Jackson method. *Journal of Difference Equations and Applications 11,* 2, 173–179.

ZEILBERGER, D. 2002. The Umbral Transfer-Matrix Method. V. the Goulden-Jackson Cluster Method for Infinitely Many Mistakes. *Integers: Electronic Journal of Combinatorial Number Theory 2, #05*.