This version is [up to minor typo corrections] the extended abstract submitted to FPSAC'02 (12/15/01).

LATTICE PATHS WITH AN INFINITE SET OF JUMPS

CYRIL BANDERIER AND DONATELLA MERLINI

ABSTRACT. First, this article gives a quick survey of what is known and can be done for the enumeration of walks on $\mathbb{N}$ with a finite set of jumps (via language theory, Brownian motion, generating functions and complex analysis or Riordan arrays). Then, we consider a case for which a language theoretical approach fails: walks with an infinite set of jumps; however, their generating function can be made explicit and the asymptotics is obtained (via the kernel method and singularity analysis). We give numerous examples of such walks related to combinatorics or theoretical computer science. Finally, we present several classes of open problems (for these rewriting rules over an infinite alphabet) related to algebraicity of associated bivariate generating functions.

RÉSUMÉ. Cet article offre dans un premier temps un survol des propriétés énumératives et asymptotiques des marches sur $\mathbb{N}$ avec un nombre fini de sauts (pour lesquelles on peut utiliser la théorie des langages, du mouvement brownien, des tableaux de Riordan, ou encore les séries génératrices en combinaison avec l'analyse complexe). Nous considérons ensuite des marches avec un nombre infini de sauts, pour lesquelles il n'est plus possible d'utiliser des grammaires algébriques ; cependant, ces règles de récriture sur un alphabet infini offrent une surprenante structure. On peut expliciter leur série génératrice (via la méthode du noyau) et faire une étude asymptotique (via l'analyse de singularité). Nous illustrons l'intêret de telles marches en combinatoire et informatique théorique par de nombreux exemples. Nous finissons par quelques problèmes ouverts reliés à la rationalité, l'algébricité des séries génératrices bivariées associées et présentons également quelques cas de modèles bidimensionnels résolubles.

1. WALKS ON $\mathbb{N}$ WITH A FINITE SET OF JUMPS

1.1. **Lattice paths and generating trees.** A considerable number of problems from computer science deals with a sum of independent identical distributed random variables $\Sigma_n = X_1 + X_2 + \cdots + X_n$ (where each of the X_i's assumes integer values). We will consider here the following model of random walks: the walk starts (at time 0) from a point Σ_0 of $\mathbb{Z}$ and at time n, one makes a jump $X_n \in \mathbb{Z}$; so the new position is given by the recurrence $\Sigma_n = \Sigma_{n-1} + X_n$ where, when $\Sigma_{n-1} = k$, the X_n's are constrained to belong to a fixed set $\mathcal{P}_k$ (that is, the possible jumps depend on the position of the walk).

We considerer in this section the case of walks homogeneous in space, i.e., all the sets $\mathcal{P}_k$'s are equal to a fixed set $\mathcal{P}$ (the simplest interesting case being $\mathcal{P} = \{-1, +1\}$). We call them "walks on $\mathbb{Z}$ with a finite set of jumps". In the next section, we consider "walks on $\mathbb{Z}$ with a infinite set of jumps", which are not space-homogeneous. In both cases, the walks are homogeneous in time. When the positions Σ_n's are constrained to be nonnegative, we talk about "walks on $\mathbb{N}$".

The probabilistic model under consideration here is the uniform distribution on all paths of length n (the combinatorial approach that we present hereafter can also easily deal with a Bernoulli model for the distribution of the jumps in $\mathcal{P}_k$).

In combinatorics, it is classical to represent a particular walk as a path in a two dimensional lattice. Thus the drawing corresponds to the walk (of length n) linking the points $\big((0,\Sigma_0),(1,\Sigma_1),\ldots,(n,\Sigma_n)\big)$. It is also convenient to represent all the walks of length $\leq n$ as a tree of height n, where the root (at level 0 by convention) is labeled with the starting point of the walks and where the label of each node at level n encodes a possible position of the walk (see Figure 1).

We note $w_{n,k}$ the number of walks on $\mathbb{N}$ of length n going from 0 to k (or, equivalently, the number of nodes with label k at level n in the tree) and we want to find the bivariate generating function

$$W(z,u) = \sum_{n\geq 0} w_n(u) z^n = \sum_{k\in\mathbb{Z}, n\geq 0} w_{n,k} u^k z^n ,$$

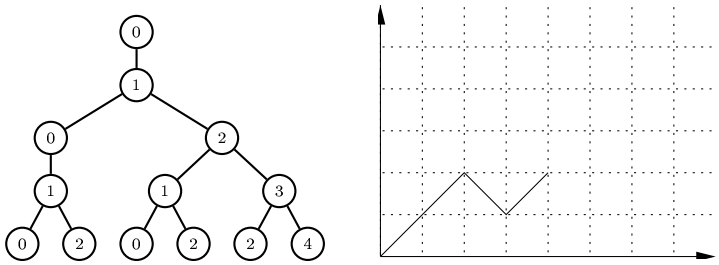


FIGURE 1. The generating tree of the walk on $\mathbb{N}$ with jumps $\mathcal{P} = \{+1, -1\}$ starting in 0 (and up to length $n = 4$). Each branch corresponds to a path. The branch $(0, 1, 2, 1, 2)$ corresponds to the path drawn on the lattice.

where u encodes the final altitude of the walk (the label in the tree), z the length of the walk (the level in the tree), and where $w_n(u)$ is a Laurent polynomial (that is, a polynomial with a finite number of monomials of negative degree).

If the walk is constrained to remain nonnegative (or equivalently if negative labels in the tree are not allowed), we consider similarly the bivariate generating function

$$F(z,u) = \sum_{n \geq 0} f_n(u) z^n = \sum_{k \in \mathbb{N}, n \geq 0} f_{n,k} u^k z^n .$$

Let $W_k(z) := [u^k]W(z,u)$ be the generating function of walks ending at altitude k. Similarly, one sets $F_k(z) := [u^k]F(z,u)$.

We consider here four classes of walks: walks (walks on $\mathbb{Z}$ ending anywhere), bridges (walks on $\mathbb{Z}$ ending in 0), excursions (walks on $\mathbb{N}$ ending in 0), and meanders (walks on $\mathbb{N}$ anywhere). The corresponding generating functions are $W(z) = W(z,1)$, $B(z) = W_0(z) = W(z,0)$, $E(z) = F_0(z) = F(z,0)$, and $M(z) = F(z,1)$.

Generating tree and rewriting rule.

The concept of generating trees has been used from various points of view and has been introduced in the literature by Chung, Graham, Hoggatt and Kleiman [16] to examine the reduced Baxter permutations. This technique has been successively applied to other classes of permutations and the main references on the subject are due to West [19, 48, 49], then followed by the Florentine school [7, 9, 24, 35, 38, 41, 40] and other authors [3, 17, 26]. A generating tree is a rooted labeled tree with the property that if v_1 and v_2 are any two nodes with the same label then, for each label ℓ, v_1 and v_2 have exactly the same number of children with label ℓ. To specify a generating tree it therefore suffices to specify: 1) the label of the root; 2) a set of rules explaining how to derive from the label of a parent the labels of all of its children. Points 1) and 2) define what we call a *rewriting rule*. For example, Figure 1 illustrates the upper part of the generating tree which corresponds to the rewriting rule $[(0), \{(k) \rightsquigarrow (k-1)(k+1)\}]$.

Riordan arrays We introduce now the concept of *matrix associated to a generating tree*: this is an infinite matrix $\{d_{n,k}\}_{n,k \in \mathbb{N}}$ where $d_{n,k}$ is the number of nodes at level n with label $k + r$, r being the label of the root. For example, the matrix associated to the generating tree of the Figure 1 is the following:

n/k	0	1	2	3	4
0	1				
1	0	1			
2	1	0	1		
3	0	2	0	1	
4	2	0	3	0	1

Many such matrices can be studied from a *Riordan array* viewpoint. In fact, the concept of a Riordan array provides a remarkable characterization of many lower triangular arrays that arise in combinatorics and algorithm analysis. The theory has been introduced in the literature in 1991 by Shapiro, Getu, Woan and Woodson [44] and then examined closely from a theoretical and practical viewpoint in [32, 37, 45]. This study has pointed out that Riordan arrays are a powerful tool in the study of many counting problems.

A Riordan array is an infinite lower triangular array $\{d_{n,k}\}_{n,k\in\mathbb{N}}$, defined by a pair of formal power series $D = (d(z), h(z))$, such that $d(z)(zh(z))^k$, i.e.:

$$d_{n,k} = [z^n]d(z)(zh(z))^k, \qquad n, k \geq 0.$$

From this definition we have $d_{n,k} = 0$ for $k > n$. The bivariate generating function for D is:

$$\sum_{n,k\geq 0} d_{n,k} u^k z^n = \frac{d(z)}{1 - uzh(z)}.$$

In what follows, we always assume that $d(0) \neq 0$; if we also have $h(0) \neq 0$ then the Riordan array is said to be *proper*; in the proper-case the diagonal elements $d_{n,n}$ are different from zero for all $n \in \mathbb{N}$. The most simple example is the Pascal triangle for which we have

$$\binom{n}{k} = [z^n]\frac{1}{1-z}\left(\frac{z}{1-z}\right)^k,$$

where we recognize the proper Riordan array with $d(z) = h(z) = 1/(1-z)$. Proper Riordan arrays are characterized [42, 45] by the existence of a sequence $A = \{a_i\}_{i\in\mathbb{N}}$ with $a_0 \neq 0$, called the *A-sequence*, such that every element $d_{n+1,k+1}$ can be expressed as a linear combination, with coefficients in A, of the elements in the preceding row, starting from the preceding column:

$$d_{n+1,k+1} = a_0 d_{n,k} + a_1 d_{n,k+1} + a_2 d_{n,k+2} + \cdots$$

It can be proved that $h(z) = A(zh(z))$, $A(z)$ being the generating function for A. For example, for the Pascal triangle we have: $A(z) = 1 + z$ and the previous relation reduces to the well-known recurrence relation for binomial coefficients. The A-sequence doesn't characterize completely $(d(z), h(z))$ because $d(z)$ is independent of $A(z)$. But it can be proved that there exists a unique sequence $Z = \{z_0, z_1, z_2, \ldots\}$, such that every element in column 0 can be expressed as a linear combination of all the elements of the preceding row:

$$d_{n+1,0} = z_0 d_{n,0} + z_1 d_{n,1} + z_2 d_{n,2} + \cdots$$

This property has been recently studied in [32], where it is proved that $d(z) = d(0)/(1 - zZ(zh(z)))$, $Z(z)$ being the generating function for Z. Thus the triple $(d(0), Z(z), A(z))$ characterizes every proper Riordan array.

1.2. **Context-free grammars and pushdown automata.** For people having an analytical affinity, we want to sketch here what is a classsical viewpoint of people from language theory on these problem of "lattice paths".

It is well known that Dyck paths (excursions with jumps $(-1, +1)$), Motzkin paths (excursions with jumps $(-1, 0, +1)$), and Łukasiewicz walks (excursions with jumps -1 and a finite set of positive jumps) can be generated by context-free grammars, and thus have an algebraic generating function.

For example, the classical (non-ambiguous) decomposition of a Dyck path into "either a empty walk, either a walk beginning by $+1$ followed by a Dyck path, itself followed by -1 jump and then by a last (eventually empty) Dyck path" gives the grammar $\mathcal{D} = \epsilon + 1\mathcal{D}\bar{1}\mathcal{D}$. In the realm of generating functions, one substitutes each letter by a z (so that the number of words of length n is given by the coefficient of z^n), and this grammar gives the following functional equation $D(z) = 1 + z^2 D(z)^2$, that can be solved to obtain an explicit formula for the generating function $D(z)$:

$$D(z) = \frac{1 - \sqrt{1 - 4z^2}}{2z^2} = 1 + 2z^2 + 5z^4 + 14z^6 + 28z^8 + O(z^{10}).$$

It is also possible to derive a grammar for walks ending anywhere (that is, for the set $\mathcal{F}$ of prefixes of Dyck paths): $\mathcal{F} = \mathcal{D} + \mathcal{D}1\mathcal{F}$ (where the letter 1 represents an ascent in the path). This leads to $F(z, u) = D(z) + uzF(z, u)D(z)$ which gives $F(z, u) = \frac{D(z)}{1 - uzD(z)}$.

For a generalized Dyck language $\mathcal{D}$ (i.e., a language associated to excursions on $\mathbb{N}$ with a finite set $\mathcal{P}$ of jumps), the grammar is not so easy to obtain but all the known derivations [27, 36, 18, 2] lead to the same grammar:

Theorem 1. *The generalized Dyck languages are algebraic. They are generated by the following context-free grammar:*

$$\begin{cases} \mathcal{D} = \epsilon + \sum_{v(a)=0} a\mathcal{D} + \sum_{k=1}^{\min(c,d)} \mathcal{L}_k \mathcal{R}_k \,, \\ \mathcal{L}_i = \sum_{v(a)=i} a\mathcal{D} + \sum_{k=i+1}^{d} \mathcal{L}_k \mathcal{R}_{k-i} \,, \\ \mathcal{R}_j = \sum_{v(b)=-j} b\mathcal{D} + \sum_{k=1}^{c} \mathcal{L}_k \mathcal{R}_{k+j} \,, \end{cases}$$

where $v(a)$ is the size of the jump encoded by a and $c = -\min v(a)$ (resp. $d = \max v(a)$).

Note that this grammar is "strongly connected" (that is, from each non terminal symbol one can access to another non terminal symbol).

Corollary 1. *The set $\mathcal{F}$ of prefixes of a generalized Dyck language (equivalently the "meanders" of the walk) cannot be generated by a strongly connected context-free grammar.*

Proof. What is called the Drmota–Lalley–Woods in [22] implies that a strongly connected context-free grammar leads to asymptotics in $\frac{\rho^n}{\sqrt{\pi n^3}}$. As the fact to be strongly connected is independent from the drift of the walk, the results in Figure 2 (for meanders without drift) imply our assertion. □

From the grammar of Theorem 1, a Gröbner basis or resultant computation gives the algebraic equation satisfied by the generating function.

Example 1. *Transformation of the pushdown automata into a context-free grammar.*The relationship between context-free grammar and pushdown automata was studied by Chomsky [15] and Schützenberger [43]. A *pushdown automata* (PDA, for short) is a finite-state machine augmented with an external stack memory.

It is easy to define an PDA which recognizes a generalized Dyck language $\mathcal{D}$: This PDA is made up of a single state, the alphabet is the set of jumps (e.g., $\{\bar{2}, \bar{1}, 1, 2\}$) and the stack alphabet is $\{I\}$ (one counts in unary). Whenever one reads $\bar{2}$, one pops II from the stack; if one reads 1, one adds I in the stack and so on. From this (see e.g. [46, Section 8.3]), one gets a context-free grammar which generates $\mathcal{D}$. The converse transformation can be done via Greibach normal forms. □

1.3. **An related problem in probability theory: Brownian motion.** For people having a language theory affinity, we want to sketch here what is a classical viewpoint of people from probability theory on these problem of "random walks": consider the set of walks of length n, divide the lattice path representation of each walk by n in the x-direction and by $\sqrt{n}$ in the y-direction, put the uniform distribution on this set of walks. One has then a random variable B_n, when n tends to infinity, this random variable converges to the classical Brownian motion $(B_t)_{t\in[0,1]}$.

Conditioning the walk to end in 0 (resp. to be ≥ 0, to be ≥ 0 and to end in 0) gives a Brownian bridge (resp. a Brownian meander, Brownian excursion). They are in a sense canonical asymptotic representants of our discrete random walks. In probability theory, there is a huge literature giving the limit laws for most of the parameters of these Brownian objects. Some of these parameters (such as the theta distribution for the height [14, 10], the Airy distribution for the area [29]) still resist to a combinatorial approach (excepted for some simple case like Dyck paths, or more generally walks for which -1 is the only negative jump, this is what probabilists call left-continuous random walk). However, combinatorics is very useful as it provides at least two fundamental results not accessible to probability theory: exact enumeration and complete asymptotic expansions.

Indeed, probabilists cannot usually access, for some reasons intrinsic to their approach, to more than the main term in the following complete asymptotic expansion:

$$f_n = C\frac{A^n}{(2\pi n)^{3/2}}\left(1 + \sum_{i=1}^{\infty} \frac{a_i}{n^i}\right)$$

where C, A and the a_i's are algebraic constants which can be computed via analytic combinatorics.

Unfortunately, for combinatorialists, this kind of complete asymptotic expansion is only accessible when the model of random walks is very "regular" (both in time and space), whereas probability theory can deals with much more irregularities.

This article shows what can be done for some regular model of walks with different kind of jumps.

1.4. **Kernel method and formula for the generating function.** In fact, it is possible to say much more than what the context-free grammar approach offers: analytic combinatorics shows that the algebraic structure of the generating function depends entirely on the roots (converging in $z = 0$) of $1 - zP(u) = 0$, where $P(u)$ the characteristic polynomial associated to the walk:

$$P(u) = \sum_{k=-b}^{a} p_k u^k := \sum_{i \in \mathcal{P}} u^i \,. \qquad (1)$$

This approach was used in [1, 2, 4]), and allows to make explicit the generating function and the asymptotics of its coefficients in 7 steps:

1. find a recurrence [1] for $f_n(u)$;
2. translate this recurrence into a functional equation (via generating functions);
3. solve this functional equation (via the kernel method);
4. express the generating function as a product $A \cdot B$ where A is "simple";
5. prove that only the factor A matters asymptotically (topology of the branches);
6. prove that A has a square-root behavior (analysis of singularity);
7. deduce the asymptotics of the number of walks (transfer lemma).

For example, for $\mathcal{P} = \{-2, 0, 0, +1, +3\}$, i.e. $P(u) = u^{-2} + 2 + u + u^3$, one has:

1. $f_{n+1}(u) = \{u^{\geq 0}\}P(u)f_n(u) = P(u)f_n(u) - \{u^{-1}\}u^{-2}f_n(u) - \{u^{-2}\}u^{-2}f_n(u)$;
2. $(1 - zP(u))F(z,u) = 1 - zu^{-1}F_1(z) - zu^{-2}F_0(z)$;
3. We call $1 - zP(u)$ the kernel. It has 2 small branches $u_1(z)$ and $u_2(z)$ (~ 0 in $z \sim 0$). As the right hand side of the previous equation can be considered (rewritten) as a polynomial in u of degree 2, for which one knows the two roots (u_1 and u_2), one can factorize it. This leads to $u^c(1 - zP(u))F(z,u) = (u - u_1)(u - u_2)$.
4. The simple factor is $A = u - u_1(z)$ and we refer to [4] for the steps 5, 6, 7.

Figure 2 gives a summary of the most notable results for these one-dimensional walks. To answer to a question of Bousquet-Mélou, it is noteworthy that these results have some non trivial applications on two-dimensional walks: using her results from [11], and then using asymptotic results from [4], we can prove the non D-finiteness of some generating functions of walks on the slit plane.

Example 2. *Colored Motzkin paths.* In [37], the authors consider the following colored Motzkin walk: $[(0), \{(k) \rightsquigarrow (k-1)^\gamma (k)^\beta (k+1)^\alpha\}]$ and a variation, the "Shapiro-colored Motzkin paths" : $[\{(0), \{(0) \rightsquigarrow (0)^{\beta'}(1)^\alpha, (k) \rightsquigarrow (k-1)^\gamma (k)^\beta (k+1)^\alpha\}]$ for which the sets of colors available for the horizontal jump depends on whether one is at altitude 0 or not.

They proved that the Riordan array associated to these walks is $(1, A, Z)$ where $A = (\alpha, \beta, \gamma, 0, 0, \dots)$, $Z = (\beta, \gamma, 0, 0, \dots)$ for $F(z,u)$ (meanders), $Z_q = (\beta, 2\gamma, 0, 0, \dots)$ for $W(z,u)$ (walks) in the case of colored Motzkin walks; or $Z = (\beta', \gamma, 0, 0, \dots)$ for $F(z,u)$, $Z_q = (\beta', 2\gamma, 0, 0, \dots)$ for $W(z,u)$ in the case of Shapiro-colored Motzkin walks.

Theorem 5 for finding the rewriting rule associated to a Riordan array gives:
$[(r), \{(k) \rightsquigarrow (k-1)^\gamma (k)^\beta (k+1)^\alpha\}]$ for Motzkin meanders,
$[(r), \{(r+1) \rightsquigarrow (r)^{2\gamma} (r+1)^\beta (r+2)^\alpha, (k) \rightsquigarrow (k-1)^\gamma (k)^\beta (k+1)^\alpha\}]$ for Motzkin walks,
$[(r), \{(r) \rightsquigarrow (r-1)^\gamma (r)^{\beta'} (r+1)^\alpha, (k) \rightsquigarrow (k-1)^\gamma (k)^\beta (k+1)^\alpha\}]$ for Shapiro-colored Motzkin meanders,
$[(r), \{(r) \rightsquigarrow (r-1)^\gamma (r)^{\beta'} (r+1)^\alpha, (r+1) \rightsquigarrow (r)^\gamma (r+1)^{\beta+\gamma} (r+2)^\alpha, (k) \rightsquigarrow (k-1)^\gamma (k)^\beta (k+1)^\alpha\}]$ for Shapiro-colored Motzkin walks. (The label r root can be taken arbitrarily.)

More generally, for walks "almost homogeneous in space" (where one has a specific set $\mathcal{P}_m$ of jumps at the altitude m) it is possible to exhibit the generating function: the recurrence is of the type $f_{n+1}(u) = P_m(u)\{u^m\}f_n(u) + \{u^{\geq 0}\}P(u)f_n(u)$ and the kernel method allows to solve it.

Note that this kernel method approach allows to mark a specific jump, to do it recursively would allow to predict the number of jumps of a given type in order to perform uniform random generation.

Another variation, the "generalized Motzkin paths" (which include the classical Dyck, Motzkin, Schroeder paths) studied by Sulanke in [47] are also accessible via the kernel method, upon a natural generalization: to consider dependencies of higher Markovian order (see [5]).

Finally, note that the colored Motzkin walks correspond to the cases where one has also a natural continued fraction representation of $E(z)$ (see some old stuff like [21]). □

[1] To this effect, the notation $\{u^{<0}\}P(u)$ proves convenient and stands for the sum of the monomial with a negative degree in a Laurent polynomial $P(u)$.

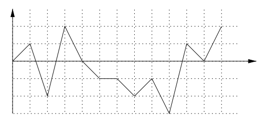
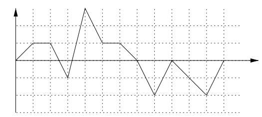
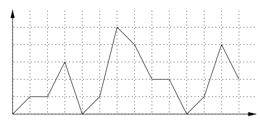
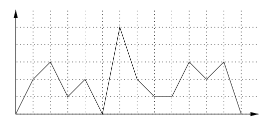
	ending anywhere	ending at 0
unconstrained (on $\mathbb{Z}$)	 walk/path ($\mathcal{W}$) $W(z) = \dfrac{1}{1 - zP(1)}$ $W_n = P(1)^n$	 bridge ($\mathcal{B}$) $B(z) = z \sum_{i=1}^{c} \dfrac{u_i'(z)}{u_i(z)}$ $B_n \sim \beta_0 \dfrac{P(\tau)^n}{\sqrt{2\pi n}}$
constrained (on $\mathbb{N}$)	 meander ($\mathcal{M}$) $M(z) = \dfrac{1}{1 - zP(1)} \prod_{i=1}^{c} (1 - u_i(z))$ $M_n \sim \mu_0 \dfrac{P(1)^n}{\sqrt{\pi n}}$ or $\mu_0^- \dfrac{P(\tau)^n}{2\sqrt{\pi n^3}}$ or $\mu_0^+ P(1)^n$	 excursion ($\mathcal{E}$) $E(z) = \dfrac{(-1)^{c-1}}{p_{-c} z} \prod_{i=1}^{c} u_i(z)$ $E_n \sim \epsilon_0 \dfrac{P(\tau)^n}{2\sqrt{\pi n^3}}$

FIGURE 2. The four types of paths: walks, bridges, meanders, and excursions. The corresponding generating functions and the asymptotics are also given (τ is such that $P'(\tau) = 0$. For meanders, there are 3 different asymptotics corresponding to the cases $P'(1) = 0$, $P'(1) < 0$, and $P'(1) > 0$. Details can be found in [4]).

EXAMPLE 3. *Simple families of walks: Łukasiewicz paths and tree codes.*
Consider generally a finite set $\mathcal{P}$ that contains -1 as single negative value. The corresponding paths are known as Łukasiewicz paths. They could also be called "simple families of walks", as they are related to the "simple families of trees" via the correspondence described below. Set $\phi(u) := uP(u)$, which is a polynomial (from Formula 1). There is only one small branch satisfying

$$u_1(z) = z\phi(u_1(z)), \tag{2}$$

and the GF of excursions is $\frac{1}{zp_{-1}} u_1(z)$. Łukasiewicz paths of type $\mathcal{P}$ encode trees whose node degrees are constrained to lie in $1+\mathcal{P}$, this by virtue of a well-known correspondence [28, Chap. 11]. (Traverse the tree in preorder and output a step of $d-1$ when a node of outdegree d is encountered.) In this way, it is seen that Equation (2) gives the GF of trees counted according to the number of their nodes, an otherwise classical result [31]. By Lagrange inversion, the number of trees comprised of n nodes is

$$T_n = \frac{1}{n} [w^{n-1}] \phi(w)^n,$$

where ϕ can be directly interpreted as the polynomial of the allowed node (out)degrees. □

2. WALKS ON $\mathbb{Z}$ WITH A INFINITE SET OF NEGATIVE JUMPS

Consider a sequence $(e_i(k))_{i \geq -a}$ (for a given integer $a > 0$) of polynomials assuming nonnegative integers values then the walk with an infinite set of jumps under consideration here are of the following kind:

$$[(0), \{(k) \rightsquigarrow (0)^{e_k(k)} (1)^{e_{k-1}(k)} (2)^{e_{k-2}(k)} \dots (k-1)^{e_1(k)} (k)^{e_0(k)} \dots (k+a)^{e_{-a}(k)}\}]. \tag{3}$$

where the exponent $e_i(k)$ is the multiplicity of the jumps $-i$ when one is at position k.

It may seem counter intuitive to write the $e_i(k)$'s in this direction; the reason is that if one takes for $e_i(k)$ an increasing sequence (e.g. $e_i(k) = i+1$) that one uses as exponents from left to right, then one gets a ordinary generating function with a radius of convergence 0 (as its coefficients $f_n(1)$ grow faster than $n!$, which corresponds to the rule $(k) \rightsquigarrow (k+1)^k$), whereas we show hereafter that the GF remains algebraic for a huge class of walks, if one writes the $e_i(k)$'s from right to left. The intuition is that one can put a huge multiplicity around (0), but not around (k).

If the sequence of polynomials $(e_i(k))_{i \geq -a}$ is ultimately $e_i(k) = 0$, then the situation covers the case of walks with a finite set of jumps studied in the previous section. If the sequence is ultimately $e_i(k) = 1$, then this covers the case of "factorial rules" which are of great interests for the generation of combinatorial objects [8] and for which it was proven in [3] that the associated generating function is algebraic.

We still note $f_{n,k}$ the number of walks on $\mathbb{N}$ of length n going from 0 to k and we want to find the bivariate generating function $F(z,u) = \sum_{n,k \geq 0} f_{n,k} u^k z^n$.

As in the previous section, these random walks on $\mathbb{N}$ can equivalently be seen as lattice paths, generating trees and also as Riordan arrays (when $a = 1$).

However, as the alphabet (the jumps) is now infinite, we want to emphasize the fact that there is no more a language theory argument (related to pushdown automata, or equivalently to context-free grammars) proving that the generating functions $F(z,u)$ of these walks are algebraic. The theory of context-free grammars with an infinite set of rules and an infinite alphabet remains to be done...

It is also noteworthy that people from probability theory (who are used to see a Brownian motion behind each random walk!) will not have a direct and simple intuition of the limit laws for this class of walks, as the renormalization to the Brownian (which is well known when there is a finite set of jumps) is not so clear in the case of an infinite set of jumps.

Fortunately, combinatorics allow us to exhibit the generating function of such a walk and complex analysis allows to give the asymptotics of its coefficients.

Finally, note that the study of walks on $\mathbb{Z}$ with a infinite set of *positive* jumps is equivalent to the problem under consideration here, by reversion of the time axis.

2.1. **Rationality and algebraicity of classes of rewriting rules.**

Theorem 2. *For a constant $B \geq 0$, the rule*

$$[(r), \{(k) \rightsquigarrow (0)^{e_k(k)} \dots (B)^{e_{k-B}(k)} (k)^{e_0} (k+1)^{e_{-1}} \dots (k+a)^{e_{-a}}\}]$$

(where $e_k(k), \dots, e_{k-B}(k)$ are polynomial in k, $e_i(k) = 0$ for $0 < i < k-B$ and $e_i(k) = e_i$, some fixed constants, for $i \leq 0$) has a rational generating function $F(z,u)$.

Sketch of proof. We illustrate the general case by the following example:

$$[(0), \{(k) \rightsquigarrow (0)^{k^2} (2)^{3k-1} (3)(k)(k+1)^2 \dots (k+3)^5\}] .$$

As the multiplicity $3k-1$ is obtained by $\partial_u f_n(u^3)/u$ (which gives $3f_n'(1) - f_n(1)$ once instantiated in $u = 1$), one has the following recurrence:

$$f_{n+1}(u) = (1 + 2u + 5u^3) f_n(u) + u^0 (f_n''(1) + f_n'(1)) + u^2 (3f_n'(1) - f_n(1)) + u^3 f_n(1) .$$

This leads to a functional equation

$$(1 - zP(u))F(z,u) = 1 + z(u^3 - 1)F(z,1) + z(3u^2 - 1)F'(z,1) + zF''(z,1)$$

taking the first 2 derivatives and instantiating in $u = 1$ gives a rational system of full rank, hence $F(z,u)$ is rational: $F(z,u) = \frac{u^3(22z^2 - 112z^3 - z) + u^2(480z^3 - 60z^2) + 528z^3 + -250z^2 + 31z - 1}{(1 - zP(u))(872z^3 - 212z^2 + 30z - 1)}$. □

Theorem 3. *For a constant $B \geq 0$, the rule*

$$[(r), \{(k) \rightsquigarrow (0)^{e_k(k)} \dots (B)^{e_{k-B}(k)} (B+1) \dots (k-b-1)(k-b)^{e_b} \dots (k+a)^{e_{-a}}\}]$$

(where $e_k(k), \dots, e_{k-B}(k)$ are polynomial in k, $e_i(k) = 1$ for $b < i < k-B$ and $e_i(k) = e_i$, some fixed constants, for $i \leq b$) has an algebraic generating function $F(z,u)$.

Sketch of proof. When $B = 0$, $r = 0$ and $e_k(k) = 1$, it corresponds to the theorem giving the algebraicity of "factorial rules" in [3]. We illustrate the general case by the following example:

$$[(0), \{(k) \rightsquigarrow (0)^{k^2}(2)^{3k^5-2}(6)(7)\dots(k-5)(k-4)^2(k-2)^3(k)(k+3)^2(k+23)\}]$$

The recurrence is

$$f_{n+1}(u) = (u^{-4} + 3u^{-2} + 1 + 2u^3 + u^{23})f_n(u) - \{u^{<0}\}P(u)f_n(u) + \sum_{i=0}^{5} t_i(u)\partial_u^i f_n(1)\,,$$

this gives the functional equation

$$(1 - zP(u))F(z,u) = 1 - z\sum_{k=0}^{c-1} r_k(u)F_k(z) + z\sum_{i=0}^{5} t_i(u)\partial_u^i F(z,1)\,.$$

The 4 small roots of the kernel and the 5 equations obtained by taking the first 5 derivatives and setting $u = 1$ gives a system of full rank with 9 equations with 9 unknown univariate generating functions, which are thus all algebraic, and then one has a formula for $F(z,u)$, involving the u_i, which implies its algebraicity. □

The "kernel method" has been part of the folklore of combinatorialists for some time and is related to the what is known as "the quadratic method" in enumeration of planar maps [13]. Earlier references (see [25] Ex. 2.2.1.11 for Dyck paths, [39, Sec. 15.4] for a pebbling game, [20] for a queuing theory application) deals with the case of a functional equation of the form

$$K(z,u)F(z,u) = A(z,u) + B(z,u)G(z)$$

(with F, G the unknown functions), when there is only one small branch, u_1, such that $K(z,u_1(z)) = 0$. In that case, a single substitution does the job, and $G(z) = -A(z,u_1)/B(z,u_1)$.

The kernel method in its more general version was developed in a few unpublished works by Flajolet and Banderier [1, 2] and also in their study of lattice paths [4]. Sometimes, it is not necessary to solve the system obtained by plugging the small branches (this observation is due to Bousquet-Mélou and first appeared in [12] and also in [3]); this is unfortunately not the case for the Theorem 3 in its full generality.

Example 4. *Tennis ball problem.* Let $s \geq 2$ be an integer and consider the following problem known as *the s-tennis ball problem.* At the first turn one is given balls numbered one through s. One throws one of them out of the window onto the lawn. At the second turn balls numbered $s+1$ through $2s$ are brought in and now one throws out on the lawn any of the $2s-1$ remained. Then balls $2s+1$ through $3s$ are brought in and one throws out one of the $3s-2$ available balls. The game continues for n turns. At this point, one picks up the n balls in the lawn and consider the ordered sequence $B = (b_1, b_2, \dots, b_n)$ with $b_1 < b_2 < \cdots < b_n$. This sequence will be called a *tennis ball s-sequence* and the first question is: how many tennis ball s-sequences of length n exist? The second question is: what is the sum of all the balls in all the possible s-sequences of length n ? Obviously, if we answer to both these questions, we also know the average sum of the balls in an s-sequence of length n. The problem with $s = 1$ has been solved in [30] while the general case $s \geq 1$ has been studied in [33] from a generating function viewpoint. In fact, the authors consider an infinite tree with root 0 and with s children. Each $(n+1)$-length path in this tree corresponds to an s-sequence of length n. This infinite tree is isomorphic to the generating tree with specification $[(1), \{(k) \rightsquigarrow (1)\dots(k+s-2)(k+s-1)\}]$.

By using this result the authors find that the number of tennis ball s-sequences of length n are counted by T_{n+1}, where $T_n = \frac{1}{1+(s-1)n}\binom{sn}{n}$ (the number of s-ary trees with n-nodes) and the cumulative sum of all the balls thrown onto the lawn in n turn is

$$\Sigma_n = \frac{1}{2}(sn^2 + (3s-1)n + 2s)T_{n+1} - \frac{1}{2}\sum_{k=0}^{n+1}\binom{sk}{k}\binom{s(n+1-k)}{n+1-k}.$$

More generally, we intend to develop in a forthcoming work the combinatorial and asymptotic analysis of the area under factorial walks (and of a related parameter: the internal weighted path length of the associated generating tree). □

2.2. **Asymptotics.**

Theorem 4. *The number of walks of length n for the "factorial" rule*

$$[(0),\{(k) \rightsquigarrow (0)(1)\dots(k-b-1)(k-b)^{e_b}\dots(k)^{e_0}\dots(k+a)^{e_{-a}}\}]$$

(where $e_i(k)=1$ for $b<i\leq k$ and $e_i(k)=e_i$, some fixed constants, for $i\leq b$) has the following asymptotics $A\frac{\rho^{-n}}{\sqrt{2\pi n^3}}$, where A and ρ are algebraic constants depending on the finite set of jumps $\mathcal{P}$.

Sketch of proof. The approach consists of the same seven steps as described in the previous section, even if there are some complications due to the fact that the kernel is now of the kind $1-z\phi(u)$ where $\phi(u)$ is not unimodal, but one can however establish than u_0 now dominates and has a square-root behavior. □

This theorem is the key for most of the statistical properties of this class of walks. We will give the limit laws (local time, final altitude, number of factors...) in the full version of this article.

2.3. **Constant exponents.** Consider now the case where, for each i, the exponent $e_i(k)$ of the rule (3) is a constant (that is, the polynomial in $e_i(k)$ does not depend on k, so one simply writes e_i). How far can we relate the behavior of the walk

$$[(0),(k) \rightsquigarrow (0)^{e_k}(1)^{e_{k-1}}\dots(k-2)^{e_2}(k-1)^{e_1}(k)^{e_0}(k+1)^{e_{-1}}\dots(k+a)^{e_{-a}}] \tag{4}$$

to the generating function of the exponents $E(z)=\sum_{i\geq -a} e_i z^i$?

Conjecture 1. *If the generating function of the exponents $E(z)$ is algebraic then the bivariate generating function of the walk $F(z,u)$ is algebraic.*

N.B.: When $a=1$, the Riordan arrays approach prove that $F(z,u)$ is indeed algebraic when $E(z)$ is algebraic, thanks to the following theorem [35]:

Theorem 5. *Let $r\in\mathbb{N}$, $a_j\in\mathbb{N}$, $a_0\neq 0$, $b_k\in\mathbb{Z}$, and $b_{r+j}+a_{j+1}\geq 0$, $\forall j\geq 0$ and $k\geq r$, and let*

$$[(r),\{(k) \rightsquigarrow (r)^{b_k}\prod_{j=0}^{k+1-r}(k+1-j)^{a_j}\}] \tag{5}$$

be a generating tree specification. Then, the matrix associated to (5) is a proper Riordan array D defined by the triple $(d(0),A,Z)$, such that

$$d(0)=1,\quad A=(a_0,a_1,a_2,\dots),\quad Z=(b_r+a_1,b_{r+1}+a_2,b_{r+2}+a_3,\dots).$$

Vice versa, if D is a proper Riordan array defined by $(d(0),A,Z)$ with $d(0)=1$ and $a_j,z_j\in\mathbb{N}$, then D is the matrix associated to the generating tree specification (5) with $b_{r+j}=z_j-a_{j+1}$.

Example 5. *Some simple exponents.*

When e_i for $i\geq 0$ is the number of t-ary trees with i nodes (and $a=1$, $e_{-1}=1$), $F(z,u)$ satisfies a algebraic equation of degree t. For the $t=2$ case (Catalan numbers), the generating function of excursions $F(z,0)=\frac{3+2z-\sqrt{1-4z-z^2}}{2(1+2z+z^2)}$ has some coefficients which are related to the sequence #660 in the Encyclopedia of Combinatorial Structures (see http://algo.inria.fr/encyclopedia/).

When $e_i=i+c$ (and $a=1$, $e_{-1}=1$) $F(z,u)$ satisfies an algebraic equation of degree 3: $\big((1-2u)z^2+(c-(c+1)+2u^2)\big)F^3+\big((u-2)z+(-c-2+4u-2u^2)z^2\big)F^2+\big(1+(2-2u)z\big)F=1.$
□

Example 6. A new rewriting rule for $(4,2)$-tennis ball problem.

The problem of balls on the lawn admits many other variants. For example, one could be supplied with s balls at each turn but now throw out t balls at a time with $t<s$. The general (s,t) case is an open problem while the $(4,2)$ case has been treated in [33], where the authors study the problem by introducing a bilabeled generating tree technique. Anyway, recently Merlini and Sprugnoli found that the problem can be expressed by the rule (4) with $e_i=i+3$ and $a=2$, namely:

$$[(0),\{(k) \rightsquigarrow (0)^{k+3}(1)^{k+2}(2)^{k+1}\dots(k+2)\}] \tag{6}$$

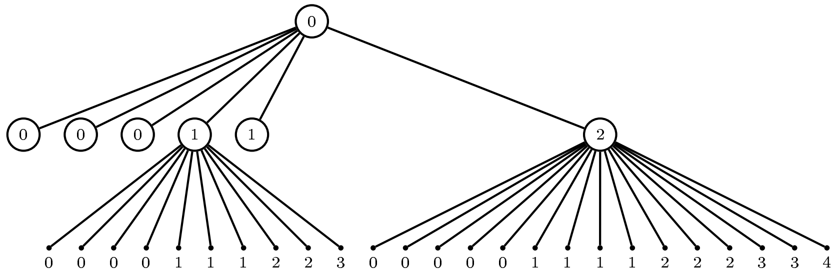


FIGURE 3. The partial generating tree for the specification (6)

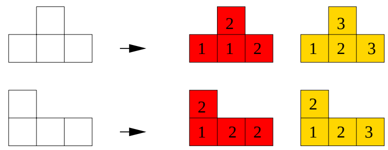


FIGURE 4. The schedules corresponding to two particular 1-histograms.

In fact, if we don't care of the order of the balls thrown away, so that the configuration $(1,4)$, $(5,8)$, $(2,10)$ is considered to be the same as $(1,2)$, $(4,5)$, $(8,10)$, it can be proved that the number of $(4,2)$-sequences of length $2n$ in which the last-but-one element is $2n+k-1$ corresponds to the number of nodes with label k at level n in the generating tree of Figure 3 (for example, the possible sequences of length 2 are $(1,2)$, $(1,3)$, $(1,4)$, $(2,3)$, $(2,4)$ and $(3,4)$). □

EXAMPLE 7. *Printers.*
In [34] the authors present a combinatorial model for studying the characteristics of job scheduling in a slow device, for example a printer in a local network. The policy usually adopted by spooling systems is called *First Come First Served* (FCFS) and can be realized by queuing the processes according to their arrival time and by using a FIFO algorithm. A job (printing a file) consists in a finite number of *actions* (printing-out a single page). Each action takes constant time to be performed (a *time slot*). If we fix n time slots, and suppose that at the end of the period the queue becomes empty, while it was never empty before, the successive states of the jobs queue can be described by a combinatorial structure called *labeled 1-histograms*. A 1-*histogram* of length n is a histogram whose last column only contains 1 cell and, whenever a column is composed by k cells, then the next column contains at least $k-1$ cells. It is at all obvious that a 1-histogram corresponds to a path in the generating tree produced by the specification $[(1),(k) \rightsquigarrow (1)\dots(k+1)]$. A *labeled* 1-*histograms* of length n is a 1-histogram in which we label each cell according to some rules (see [34] for the details). Figure 4 illustrates the possible schedules for two particular 1-histograms of length 3: the first one, for example, corresponds to i) a first job which consists in printing two pages and a second job, which starts at time slot 2, and corresponds to printing a page at time slot 3, and ii) three different jobs which consists in printing a single page, the first at time slot 1, the second at time slot 2 and the third at time slot 3, after queuing at time slot 2. It can be proved that the number of schedules of length n with k jobs request at the first time slot corresponds to the number of nodes at level n having label $k+1$ in the generating tree with specification (7).

$$[(1),\{(k) \rightsquigarrow (1)(1)\dots(k)(k)(k+1)\}] \tag{7}$$

This gives that the number S_n of possible schedules corresponds to the n^{th} small Schröder number, that is, the generating function for S_n is $(1-3z-\sqrt{1-6z+z^2})/(4z)$.

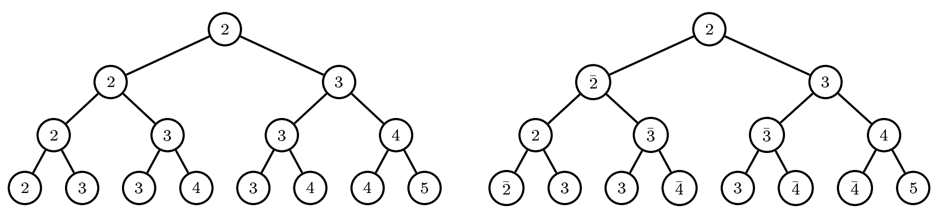
More generally, consider the rewriting rule (4) when the e_i's are ultimately constants (say, equal to C). This gives the recurrence $f_{n+1}(u) = C\frac{f_n(u)-f_n(1)}{u-1} + P(u)f_n(u)$. So $F(z,u) = \frac{\prod_{i=0}^{b} u-u_i(z)}{K(z,u)}$, where the u_i's are the $b+1$ small roots of the kernel $K(u,z) = u^b(1-u)(1-zP(u)+\frac{z}{u-1})$. This gives some nice classical closed form formulae for small values of a and b, e.g., the bivariate generating function for $b = 0$, $a = 1$, $e_{-1} = 1$ satisfies $(z + (C + u)z^2)F(z, u)^2 - (1 + (1 - u)z)F(z, u) - 1 = 0$. □

EXAMPLE 8. *Single rewriting.* It has been explained before that to put a multiplicity k on the jumps around k gives an ordinary generating function of radius of convergence 0. Let's now consider a new model of generating tree for which the generating function of such walks remains algebraic. At each level, for each label, one chooses one of the node with label k and one rewrites only this node (amongst the nodes of same label). There is also a lot of closed form formula for this model. □

3. Marked, signed, colored generating trees

In [35, 38], the authors introduced a variant of generating trees based on the concept of *marked label*: a marked label is any positive integer, marked by a bar, for which appropriate rules are given in the specification. The idea is that marked labels kill or annihilate the non-marked labels with the same label, i.e. the count relative to an integer j is the difference between the number of non-marked and marked labels j at a given level. The concept of matrix associated to a generating tree can be extended by computing the difference $d_{n,k}$ between the number of nodes at level n with label $k + r$ and the number of nodes with label $\overline{k + r}$. A simple example is given by the following generating tree of Figure 5. This tree can be seen as the *inverse* of the generating tree having specification 5 since the corresponding matrices, shown in Figure5, are the Pascal triangle and its inverse.

The concept of marked labels has also been the same concept has been used in [17] in relation with the definition of signed *ECO-systems*. A different approach can be found in [40] where the authors introduced *colored rules* which correspond to generating trees in which nodes with the same label can have various colors; what makes this approach different is the fact that these labels don't annihilate each other and the count relative to a level in the tree is exactly the number of nodes at that level.



n/k	0	1	2	3	4
0	1				
1	1	1			
2	1	2	1		
3	1	3	3	1	
4	1	4	6	4	1

n/k	0	1	2	3	4
0	1				
1	-1	1			
2	1	-2	1		
3	-1	3	-3	1	
4	1	-4	6	-4	1

FIGURE 5. The Pascal generating tree: specification $[(2), \{(k) \rightsquigarrow (k)(k+1)\}]$. The *inverse* of Pascal generating tree: $[(2), \{(k) \rightsquigarrow (\overline{k})(k+1), (\overline{k}) \rightsquigarrow (k)(\overline{k+1})\}]$. The Pascal triangle and its inverse.

EXAMPLE 9. *Directed animals.* Directed animals have been topic of interest in statistical mechanics and in enumerative combinatorics (see, for example [6, 23]). A *directed animal* is a finite subset, X, of the lattice $\mathbb{N} \times \mathbb{N}$ such that X contains the origin and any element of X is reachable from the origin via one or more paths consisting of unit north and east steps, and each point along the path is also in X. In particular, if $d_{n,k}$ is the number of directed animals with n points of which k are on the x-axis, then it can be proved that

$$d_{n+1,k+1} = d_{n,k} + d_{n-1,k} + d_{n-1,k+1} + d_{n-1,k+2} + \dots$$

In the previous sections we illustrated that, what seems essential in a Riordan array is the fact that the elements in a given row linearly depend on the elements of the row above it, starting from the previous element. But this dependence can be made much looser, as has been proved in [32]. By using this result, we can prove that the matrix corresponding to $d_{n,k}$ is indeed a Riordan array, such that the generating function for the A-sequence is

$$A(z) = \sum_{k\geq 0} a_k z^k = \frac{1}{2}\left(1+\sqrt{\frac{1+3z}{1-z}}\right) = 1+z+z^3-z^4+3z^5-6z^6+O(z^7)$$

and the Z-sequence is always 0. From a generating function viewpoint, the presence of negative coefficients in the A-sequence, corresponds to consider marked labels in the tree. The marked generating tree corresponding to directed animals can be defined as:

$$[(1), \{(k) \rightsquigarrow (2)^{a_{k-1}} \dots (k)^{a_1}(k+1)^{a_0}, (\overline{k}) \rightsquigarrow (\overline{2})^{a_{k-1}} \dots (\overline{k})^{a_1}(\overline{k+1})^{a_0}\}]\,. \tag{8}$$

Whenever a coefficient has a negative value, one changes the sign and mark (or unmark) the corresponding labels. □

Acknowledgements. This work was partially supported by the Future and Emerging Technologies programme of the EU under contract number IST-1999-14186 (ALCOM-FT), by the INRIA postdoctoral programme and by the Max-Planck Institut. The first author also benefited of pasta and of an invitation in Florence in June and November 2001.

References

[1] Cyril Banderier. Combinatoire analytique : application aux marches aléatoires. *Mémoire de DEA, Université Paris VII*, 1998.
[2] Cyril Banderier. *Combinatoire analytique des chemins et des cartes.* PhD thesis, Université de Paris 6, 2001.
[3] Cyril Banderier, Mireille Bousquet-Mélou, Alain Denise, Philippe Flajolet, Danièle Gardy, and Dominique Gouyou-Beauchamps. Generating functions for generating trees. *To appear in Discrete Mathematics*, 2000. extended version of the article from the Proceedings of FPSAC'99.
[4] Cyril Banderier and Philippe Flajolet. Basic analytic combinatorics of directed lattice paths. *Theoretical Computer Science*, June 2001. Submitted, 37 pages.
[5] Cyril Banderier and Simone Rinaldi. Rewriting rules of higher markovian order. *In preparation*, 2001.
[6] E. Barcucci, R. Pinzani, and R. Sprugnoli. The random generation of directed animals. *Theoret. Comput. Sci.*, 127(2):333–350, 1994.
[7] Elena Barcucci, Alberto Del Lungo, Elisa Pergola, and Renzo Pinzani. A methodology for plane tree enumeration. *Discrete Mathematics*, 180(1-3):45–64, 1998.
[8] Elena Barcucci, Alberto Del Lungo, Elisa Pergola, and Renzo Pinzani. From Motzkin to Catalan permutations. *Discrete Mathematics*, 217(1-3):33–49, 2000. Formal power series and algebraic combinatorics (Vienna, 1997).
[9] Elena Barcucci, Alberto Del Lungo, Elisa Pergola, and Renzo Pinzani. Permutations avoiding an increasing number of length-increasing forbidden subsequences. *Discrete Mathematics & Theoretical Computer Science. DMTCS. An Electronic Journal*, 4(1):31–44 (electronic), 2000.
[10] Philippe Biane, Jim Pitman, and Marc Yor. Probability laws related to the Jacobi theta and Riemann zeta functions, and Brownian excursions. *Bull. Amer. Math. Soc. (N.S.)*, 38(4):435–465 (electronic), 2001.
[11] Mireille Bousquet-Mélou. Walks on the slit plane: other approaches. *Advances in Applied Mathematics*, 27:243–288, 2001.
[12] Mireille Bousquet-Mélou and Marko Petkovšek. Linear recurrences with constant coefficients: the multivariate case. *Discrete Mathematics*, 225(1-3):51–75, 2000.
[13] W. G. Brown and W. T. Tutte. On the enumeration of rooted non-separable planar maps. *Canad. J. Math.*, 16:572–577, 1964.
[14] P. Chassaing, J. F. Marckert, and M. Yor. The height and width of simple trees. In *Mathematics and computer science (Versailles, 2000)*, pages 17–30. Birkhäuser, Basel, 2000.
[15] N. Chomsky. Context-free grammar and pushdown storage. *Quarterly Progress Report, M.I.T. Research Laboratory in Electronics*, 65:187–194, 1962.
[16] F. R. K. Chung, R. L. Graham, V. E. Hoggatt, and M. Kleiman. The number of Baxter permutations. *Journal of Combinatorial Theory, Series A*, 24:382–394, 1978.
[17] Sylvie Corteel. Séries génératrices exponentielles pour les eco-systèmes signés. In *FPSAC'00*. Springer, June 2000.
[18] Philippe Duchon. On the enumeration and generation of generalized Dyck words. *Discrete Mathematics*, 225(1-3):121–135, 2000. Formal power series and algebraic combinatorics (Toronto, 1998).
[19] S. Dulucq, S. Gire, and J. West. Permutations with forbidden subsequences and nonseparable planar maps. *Discrete Mathematics*, 153(1-3), 1996. Proceedings of the 5th Conference on Formal Power Series and Algebraic Combinatorics (Florence, 1993).
[20] Guy Fayolle and Roudolf Iasnogorodski. Two coupled processors: the reduction to a Riemann-Hilbert problem. *Z. Wahrsch. Verw. Gebiete*, 47(3):325–351, 1979.
[21] P. Flajolet. Combinatorial aspects of continued fractions. *Discrete Math.*, 32(2):125–161, 1980.

[22] Philippe Flajolet and Robert Sedgewick. *Analytic Combinatorics.* Book in preparation, 2001.
[23] D. Gouyou-Beauchamps and G. Viennot. Equivalence of a two-dimensional animal problem to a one-dimensional problem. *Advances in Applied Mathematics*, 9:334–357, 1988.
[24] O. Guibert and E. Pergola. Enumeration of vexillary involutions which are equal to their mirror/complement. *Discrete Math.*, 224(1-3):281–287, 2000.
[25] Donald Ervin Knuth. *The Art of Computer Programming*, volume 1. Addison-Wesley, third edition, 1997.
[26] Darla Kremer. Permutations with forbidden subsequences and a generalized Schröder number. *Discrete Mathematics*, 218(1-3):121–130, 2000.
[27] Jacques Labelle and Yeong-Nan Yeh. Generalized Dyck paths. *Discrete Mathematics*, 82:1–6, 1990.
[28] M. Lothaire. *Combinatorics on words.* Cambridge University Press, Cambridge, 1997. With a foreword by Roger Lyndon and a preface by Dominique Perrin, Corrected reprint of the 1983 original, with a new preface by Perrin.
[29] Guy Louchard. The Brownian excursion area: a numerical analysis. *Computers & Mathematics with Applications. An International Journal*, 10(6):413–417 (1985), 1984.
[30] C. L. Mallows and L. W. Shapiro. Balls on the lawn. *Journal of Integer Sequences*, 2, 1999.
[31] A. Meir and J. W. Moon. On the altitude of nodes in random trees. *anadian Journal of Mathematics*, 30:997–1015, 1978.
[32] D. Merlini, D. G. Rogers, R. Sprugnoli, and M. C. Verri. On some alternative characterizations of Riordan arrays. *Canadian Journal of Mathematics*, 49(2):301–320, 1997.
[33] D. Merlini, D. G. Rogers, R. Sprugnoli, and M. C. Verri. The tennis ball problem, 2000. Submitted for publication.
[34] D. Merlini, R. Sprugnoli, and M. C. Verri. Waiting patterns for a printer. In *FUN with algorithms 2, E. Lodi, L. Pagli, N. Santoro, Editors*, Carleton Scientific, pages 183–198, 2001.
[35] D. Merlini and M. C. Verri. Generating trees and proper Riordan Arrays. *Discrete Mathematics*, 218:167–183, 2000.
[36] Donatella Merlini, Douglas G. Rogers, Renzo Sprugnoli, and Maria Cecilia Verri. Underdiagonal lattice paths with unrestricted steps. *Discrete Applied Mathematics. Combinatorial Algorithms, Optimization and Computer Science*, 91(1-3):197–213, 1999.
[37] Donatella Merlini, Renzo Sprugnoli, and Maria Cecilia Verri. Algebraic and combinatorial properties of simple, coloured walks. In *Trees in algebra and programming—CAAP '94 (Edinburgh, 1994)*, pages 218–233. Springer, Berlin, 1994.
[38] Donatella Merlini, Renzo Sprugnoli, and Maria Cecilia Verri. An algebra for generating trees. In *Mathematics and computer science (Versailles, 2000)*, pages 127–139, Basel, 2000. Birkhäuser.
[39] A. M. Odlyzko. *Asymptotic enumeration methods*, volume II. Elsevier, 1995. In Handbook of Combinatorics, (R. Graham, M. Grötschel, and L. Lovász, eds.).
[40] Elisa Pergola, Renzo Pinzani, and Simone Rinaldi. Towards an algebra of succession rules. In *Mathematics and computer science (Versailles, 2000)*, pages 141–152, Basel, 2000. Birkhäuser.
[41] Elisa Pergola and Robert A. Sulanke. Schröder triangles, paths, and parallelogram polyominoes. *J. Integer Seq.*, 1:Article 98.1.7 (9 HTML documents), 1998.
[42] D. G. Rogers. Pascal triangles, Catalan numbers and renewal arrays. *Discrete Mathematics*, 22:301–310, 1978.
[43] M. P. Schützenberger. Context-free language and pushdown automata. *Information and Control*, 6:246–264, 1963.
[44] L. W. Shapiro, S. Getu, W.-J. Woan, and L. Woodson. The Riordan group. *Discrete Applied Mathematics*, 34:229–239, 1991.
[45] R. Sprugnoli. Riordan arrays and combinatorial sums. *Discrete Mathematics*, 132:267–290, 1994.
[46] T. A. Sudkamp. *Languages and machines.* Addison-Wesley, 1997.
[47] Robert A. Sulanke. Moments of generalized Motzkin paths. *Journal of Integer Sequences*, 3:Article 00.1.1, 1 HTML document (electronic), 2000.
[48] J. West. Generating trees and the Catalan and Schröder numbers. *Discrete Mathematics*, 146:247–262, 1995.
[49] J. West. Generating trees and forbidden subsequences. *Discrete Mathematics*, 157:363–374, 1996.

E-mail address: `Cyril.Banderier@inria.fr`, `http://algo.inria.fr/banderier`

INRIA-Rocquencourt, 78150 Le Chesnay (France) & Max-Planck-Institut, 66123 Saarbrücken (Germany)

E-mail address: `merlini@dsi.unif.it`, `http://www.dsi.unifi.it/~merlini/`

Dipartimento di Sistemi e Informatica, Universit degli Studi di Firenze, Via Lombroso 6/17, 50134 Firenze (Italy)