*Discrete Mathematics and Theoretical Computer Science* DMTCS vol. **10:2**, 2008, 125–152

# *Culminating paths*

Mireille Bousquet-Mélou[1†] and Yann Ponty[2‡]

[1] *CNRS, LaBRI, Université Bordeaux 1, 351 cours de la Libération, 33405 Talence Cedex, France*
[2] *LRI, Bât 490 Université Paris-Sud, 91405 Orsay Cedex, France*

*received 7 June 2007, revised 30 April 2008, accepted 12 May 2008.*

Let $a$ and $b$ be two positive integers. A culminating path is a path of $\mathbb{Z}^2$ that starts from $(0, 0)$, consists of steps $(1, a)$ and $(1, -b)$, stays above the $x$-axis and ends at the highest ordinate it ever reaches. These paths were first encountered in bioinformatics, in the analysis of similarity search algorithms. They are also related to certain models of Lorentzian gravity in theoretical physics.

We first show that the language on a two letter alphabet that naturally encodes culminating paths is not context-free.

Then, we focus on the enumeration of culminating paths. A step by step approach, combined with the kernel method, provides a closed form expression for the generating function of culminating paths ending at a (generic) height $k$. In the case $a = b$, we derive from this expression the asymptotic behaviour of the number of culminating paths of length $n$. When $a > b$, we obtain the asymptotic behaviour by a simpler argument. When $a < b$, we only determine the exponential growth of the number of culminating paths.

Finally, we study the uniform random generation of culminating paths via various methods. The rejection approach, coupled with a symmetry argument, gives an algorithm that is linear when $a \geq b$, with no precomputation stage nor non-linear storage required. The choice of the best algorithm is not as clear when $a < b$. An elementary recursive approach yields a linear algorithm after a precomputation stage involving $O(n^3)$ arithmetic operations, but we also present some alternatives that may be more efficient in practice.

**Keywords:** Lattice paths – Enumeration – Random generation

## 1 Introduction

One-dimensional lattice walks on $\mathbb{Z}$ have been extensively studied over the past 50 years. These walks usually start from the point 0, and take their steps in a prescribed finite set $\mathcal{S} \subset \mathbb{Z}$. A large number of results are now known on the enumeration of sub-families of these walks, and can be obtained in a systematic way once the set $\mathcal{S}$ is given. This includes the enumeration of *bridges* (walks ending at 0), *meanders* (walks that always remain at a non-negative level), *excursions* (meanders ending at level 0), excursions of bounded height, and so on. In particular, the nature of the associated generating functions is

† MBM was partially supported by the French "Agence Nationale de la Recherche", project SADA ANR-05-BLAN-0372.

‡ YP was partially supported by the action "Aspects mathématiques et algorithmiques des réseaux biochimiques et évolutifs ($\pi$-vert)" of ACI Nouvelles Interfaces des Mathématiques, French Ministry of Research.

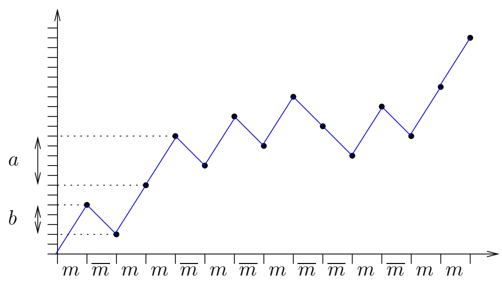

**Fig. 1:** A culminating path (for $a = 5$ and $b = 3$) and the corresponding word.

well understood: these series are always algebraic, and even rational for bounded walks [2, 5, 10, 8, 19, 26, 31, 32, 37]. These algebraicity properties actually reflect the fact that the languages on the alphabet $\mathcal{S}$ that naturally encode these families of walks are *context-free*, and even *regular* in the bounded case. In many papers, these one-dimensional walks are actually described as *directed* two-dimensional (2D) walks, upon replacing the starting point $0$ by $(0, 0)$ and every step $s$ by $(1, s)$. This explains why excursions are often called *generalized Dyck paths* (the authentic Dyck paths correspond to the case $\mathcal{S} = \{1, -1\}$). This two-dimensional setting allows for a further generalisation, with steps of the form $(i, j)$, with $i > 0$ and $j \in \mathbb{Z}$, but this does not affect the nature of the associated languages and generating functions. The uniform random generation of these walks has also been investigated, through a recursive approach [39, 24, 20] or using an anticipated rejection [6, 33].

This paper deals with a new class of walks which has recently occurred in two independent contexts, and seems to have a more complicated structure than the above mentioned classes: *culminating walks*. A 2D directed walk is said to be culminating if each step ends at a positive level, and the final step ends at the highest level ever reached by the walk (Figure 1). We focus here on the case where the steps are $(1, a)$ and $(1, -b)$, with $a$ and $b$ positive, hoping that this encapsulates all the possible typical behaviours.

In the case $a = b = 1$, culminating walks have recently been shown to be in bijection with certain *Lorentzian triangulations* [18], a class of combinatorial objects studied in theoretical physics as a model of discrete two-dimensional Lorentzian gravity. Using a transfer matrix approach, the authors derived the generating function for this case. We give two shorter proofs of their result. Also, while it is not clear how the method used in [18] could be extended to the general $(a, b)$-case, one of our approaches works for arbitrary values of $a$ and $b$.

The general $(a, b)$-case appears in bioinformatics in the study of the sensitivity of heuristic homology search algorithms, such as BLAST, FASTA or FLASH [1, 34, 11]. These algorithms aim at finding the most conserved regions (*similarities*) between two genomic sequences (DNA, RNA, proteins...) while allowing certain alterations in the entries of the sequences. In order to avoid the supposedly intrinsic quadratic complexity of the deterministic algorithms, these heuristic algorithms first consider identical regions of bounded size and extend them in both directions, updating the score with a bonus for a match or a penalty for an alteration, until the score drops below a certain threshold. The evolution of the score all the way through the final alignment turns out to be encoded by a culminating walk.

In [30], we first studied the probability of a culminating walk to contain certain patterns called *seeds*, as some recent algorithms make use of them to relax the mandatory conservation of small *anchoring* portions. Then, we proposed a variant of the recursive approach for the random generation of these walks.

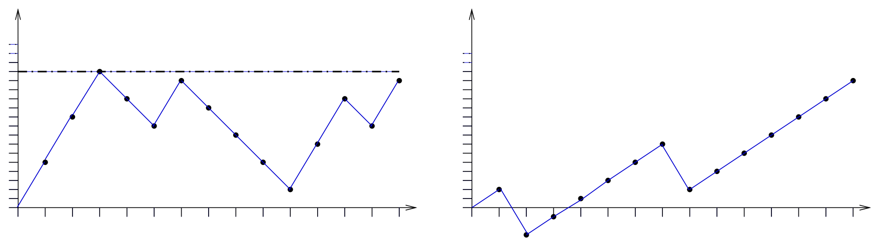

**Fig. 2:** Two walks that are not culminating, violating the final record condition (left) or the positivity condition (right).

Finally, we observed that the naive rejection-based algorithm, which consists in drawing uniformly at random up and down steps and rejecting the resulting walk if is not culminating, seemed to be linear (resp. exponential) when $a > b$ (resp. $a < b$). This observation, which is closely related to the asymptotic enumeration of culminating walks, is confirmed below in Section 6.2.

To conclude this introduction, let us fix the notation and summarize the contents of this paper. Let $a$ and $b$ be two positive integers. A walk (or path) of length $n$ is a sequence $(0, \eta_0), \ldots, (n, \eta_n)$ such that $\eta_0 = 0$ and $\eta_{i+1} - \eta_i \in \{a, -b\}$ for all $i$. The *height* of the walk is the largest of the $\eta_i$'s, while the *final height* is $\eta_n$. The walk is *culminating* if the two following conditions hold:

$$\forall i \in [1, n], \quad \eta_i > 0 \quad \text{(Positivity)},$$

$$\forall i \in [0, n-1], \quad \eta_i < \eta_n \quad \text{(Final record)}.$$

See Figures 1 and 2 for examples and counter-examples. We encode every walk by a word on the alphabet $\{m, \overline{m}\}$ in a standard way: each ascending step $(1, a)$ is replaced by a letter $m$ and each descending step $(1, -b)$ is replaced by a letter $\overline{m}$. We denote by $\{m, \overline{m}\}^*$ the set of words on the alphabet $\{m, \overline{m}\}$. From now on, we identify a path and the corresponding word. Since these objects are essentially one-dimensional, we will often use a 1D vocabulary, saying, for instance, that our paths take steps $+a$ and $-b$ (rather than $(1, a)$ and $(1, -b)$). We hope that this will not cause any confusion. Without loss of generality, we restrict our study to the case where $a$ and $b$ are coprime.

For any word $w$, we denote by $|w|_m$ (resp. $|w|_{\overline{m}}$) the number of occurrences of the letter $m$ (resp. $\overline{m}$) that it contains. We denote by $|w|$ the length of $w$. The function $\phi_{a,b} : \{m, \overline{m}\}^* \to \mathbb{N}$ maps a word to the final height of the corresponding walk. That is, $\phi_{a,b}(w) = a|w|_m - b|w|_{\overline{m}}$. The culmination properties can be translated into the following language-theoretic definition:

**Definition 1.1** *The language of* culminating words *is the set* $\mathcal{C}^{a,b} \subset \{m, \overline{m}\}^*$ *of words* $w$ *such that, for every non-empty prefix* $w'$ *of* $w$*:*

$$\phi_{a,b}(w') > 0 \quad \textit{(Positivity)},$$

*and, for every proper prefix* $w'$ *of* $w$*:*

$$\phi_{a,b}(w') < \phi_{a,b}(w) \quad \textit{(Final record)}.$$

The main result of Section 2 is that the language $\mathcal{C}^{a,b}$ is not context-free. In Section 3, we obtain a closed form expression for the generating function of culminating walks. This expression is complicated, but we believe this only reflects the complexity of this class of walks. This enumerative section is closely related to the recent work [10], devoted to a general study of excursions confined in a strip. In particular, *symmetric functions* play a slightly surprising role in the proof and statement of our results. We then derive in Section 4 the asymptotic number of culminating walks, in the case $a \geq b$. Our result implies that, asymptotically, a positive fraction of (general) $(a, b)$-walks are culminating if $a > b$. We prove that this fraction tends to 0 exponentially fast if $a < b$. More precisely, we determine the exponential growth of the number of culminating walks. This asymptotic section uses the results obtained in [5] on the exact and asymptotic enumeration of excursions and meanders. Finally, in Section 6, we present several algorithms for generating uniformly at random culminating walks of a given length. Our best algorithms are linear when $a \geq b$. When $a < b$, the choice of the best algorithm is not obvious. An elementary recursive approach yields a quasi-linear generating stage but requires the precomputation and storage of $O(n^3)$ numbers. We exploit in this section several generation schemes, like the recursive method [39, 24], the rejection method [14] and Boltzmann samplers [20]. Moreover, we address in Section 5 the random generation of positive walks, which is a preliminary step in some of our algorithms generating culminating walks. We have implemented our algorithms in Java, and we invite the reader to generate his/her own paths at the address `http://www.lri.fr/~ponty/walks`. Figure 3 shows random culminating paths of length 1000 generated with our software, for various values of $a$ and $b$.

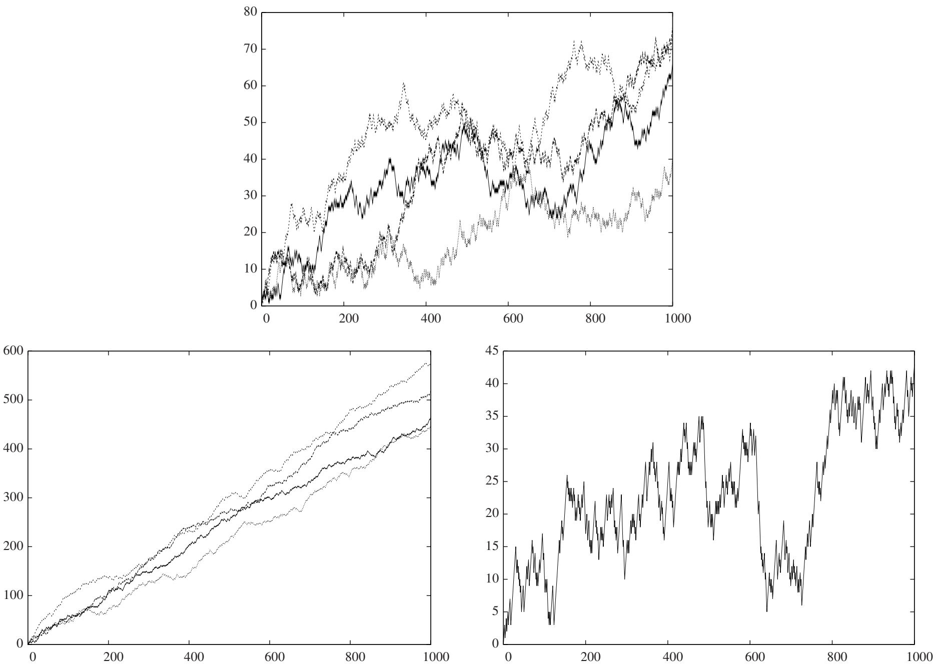


**Fig. 3:** Random culminating paths of size 1000, when $(a, b) = (1, 1)$, $(a, b) = (2, 1)$, $(a, b) = (1, 2)$. In the first two cases, four paths are displayed, while for the sake of clarity, only one path is shown in the third case.

# 2 Language theoretic properties

We denote by $\mathcal{C}^{a,b \Rightarrow k}$ the subset of $\mathcal{C}^{a,b}$ that consists of the walks (words) ending at height $k$. It will be easily seen that this language (for a fixed $k$) is regular. However, we shall prove that the full language $\mathcal{C}^{a,b}$ is not context-free. We refer to [27] for definitions on languages.

## 2.1 Culminating walks of bounded height

**Proposition 2.1** *For all $a, b, k \in \mathbb{N}$, the language $\mathcal{C}^{a,b \Rightarrow k}$ of culminating words ending at height $k$ is regular.*

**Proof:** The culminating paths of final height $k$ move inside a *bounded* space. This allows us to construct a (deterministic) finite-state automaton that recognizes these paths. The states of this automaton are the accessible heights (that is, $0, 1, \dots, k$), plus a *garbage* state $\perp$. The initial state is $0$, the final state is $k$, and the transition function $\delta$ is given, for $0 \leq q < k$, by:

$$\delta(q, m) \quad = \quad \begin{cases} q + a & \text{if } q \leq k - a, \\ \perp & \text{otherwise} \end{cases} \quad , \quad \delta(q, \overline{m}) \quad = \quad \begin{cases} q - b & \text{if } q > b, \\ \perp & \text{otherwise} \end{cases} \quad ,$$

while

$$\delta(k, \cdot) = \delta(\perp, \cdot) = \perp .$$

Clearly, this automaton sends any word attempting to walk below $0$ (resp. above $k$) in the garbage $\perp$, where it will stay forever and therefore be rejected. Moreover, it only accepts those words ending in the state $k$. Hence this automaton recognizes exactly $\mathcal{C}^{a,b \Rightarrow k}$. Since the state space is finite, $\mathcal{C}^{a,b \Rightarrow k}$ is a regular language. $\square$

## 2.2 Unbounded culminating walks

**Proposition 2.2** *For all $a, b \in \mathbb{N}$, the language $\mathcal{C}^{a,b}$ of culminating walks is not context-free.*

**Proof:** Recall that the intersection of a context-free language and a regular language is context-free [27]. Let $\mathcal{L}$ be the following regular language: $\mathcal{L} = m^*.\overline{m}^*.m^*$. It can be seen as the language of "zig-zag" paths. Let $\mathcal{K} = \mathcal{C}^{a,b} \cap \mathcal{L}$. It is easy to see that

$$\mathcal{K} = \{m^i.\overline{m}^j.m^k |\, i > 0,\, bj < ai \text{ and } bj < ak\}.$$

Assume that $\mathcal{C}^{a,b}$ is context-free. Then so is $\mathcal{K}$, and, by the pumping lemma for context-free languages [27, Theorem 4.7], there exists $n \in \mathbb{N}$ such that any word $w \in \mathcal{K}$ of length at least $n$ admits a factorisation $w = x.u.y.v.z$ satisfying the following properties:

$(i)$ $|u.v| \geq 1$,

$(ii)$ $|u.y.v| \leq n$,

$(iii)$ $\forall \ell \geq 0, w_\ell := x.u^\ell.y.v^\ell.z \in \mathcal{K}$.

| Where is the factor $u.y.v$? | $\ell$ | $w_\ell$ | Failing condition |
|---|---|---|---|
| $A$ | 0 | $m^{i-h}.\overline{m}^j.m^i$ | Pos.: $\phi(m^{i-h}.\overline{m}^j) = 1 - ah \le 0$ |
| $B$ | 2 | $m^i.\overline{m}^{j+h}.m^i$ | Pos.: $\phi(m^i.\overline{m}^{j+h}) = 1 - bh \le 0$ |
| $C$ | 0 | $m^i.\overline{m}^j.m^{i-h}$ | Fin. rec.: $\phi(w_\ell) = \phi(m^i) - ah \le \phi(m^i)$ |
| $A \cup B$ | | | |
| $\lvert u\rvert_{\overline{m}}.\lvert u\rvert_m + \lvert v\rvert_{\overline{m}}.\lvert v\rvert_m \neq 0$ | 2 | $m^p.\overline{m}^k.m^{k'}.\overline{m}^{p'}.m^i$ | $w_\ell \notin \mathcal{L}$ (Too many peaks) |
| $u = m^k,\ v = \overline{m}^{k'}$ | 2 | $m^{i+k}.\overline{m}^{k'+j}.m^i$ | Final record: $\phi(w_\ell) = \phi(m^{i+k}) + 1 - bk' \le \phi(m^{i+k})$ |
| $B \cup C$ | | | |
| $\lvert u\rvert_{\overline{m}}.\lvert u\rvert_m + \lvert v\rvert_{\overline{m}}.\lvert v\rvert_m \neq 0$ | 2 | $m^i.\overline{m}^p.m^k.\overline{m}^{k'}.m^{p'}$ | $w_\ell \notin \mathcal{L}$ (Too many valleys) |
| $u = \overline{m}^k,\ v = m^{k'}$ | 2 | $m^i.\overline{m}^{j+k}.m^{k'+i}$ | Pos.: $\phi(m^i.\overline{m}^{j+k}) = 1 - kb \le 0$ |

**Tab. 1:** Why the pumping lemma is not satisfied.

Since $a$ and $b$ are coprime, there exist $i > n$ and $j > n$ such that $ia - jb = 1$ (this is the Bachet-Bezout theorem). Hence the word $w = m^i\overline{m}^j m^i$ belongs to $\mathcal{K}$. In the rest of the proof, we will refer to the first sequence of ascending steps of $w$ as $A$, to the descending sequence as $B$ and to the second ascending sequence as $C$.

In Table 1, we consider all eligible factorisations of $w$ of the form $w = x.u.y.v.z$. Five cases arise, depending on which part of $w$ contains the factor $u.y.v$. Condition $(ii)$ implies that this factor cannot overlap simultaneously with the parts $A$ and $C$. Each of the cases $A \cup B$ and $B \cup C$ is further subdivided into two cases, depending on whether $u$ and $v$ are *monotone* or not.

For each factorisation, the table gives a value of $\ell$ for which the word $w_\ell$ does not belong to $\mathcal{K}$. This is justified in the rightmost column: either $w_\ell$ does not belong to the set $\mathcal{L}$ of zig-zag paths, or the positivity condition does not hold, or the last step of the walk is not a record.

Once all the possible factorisations have been investigated and found not to satisfy the pumping lemma, we conclude that the languages $\mathcal{K}$ and $\mathcal{C}^{a,b}$ are not context-free. □

# 3 Exact enumerative results

In this section, we give a closed form expression for the generating function of $(a, b)$-culminating walks. More precisely, we give an expression for the series counting culminating walks of height $k$, and then sum over $k$. This summation makes the series a bit difficult to handle, for instance to extract the asymptotic behaviour of the coefficients (Section 4). We believe that this complexity is inherent to the problem. In particular, we prove that the generating function of $(1, 1)$-culminating walks is not only transcendental, but also not *D-finite*. That is, it does not satisfy any linear differential equation with polynomial coefficients [37, Ch. 6].

## 3.1 Statement of the results and discussion

Let us first state our results in the (1,1)-case and then explain what form they take in the general $(a, b)$-case.

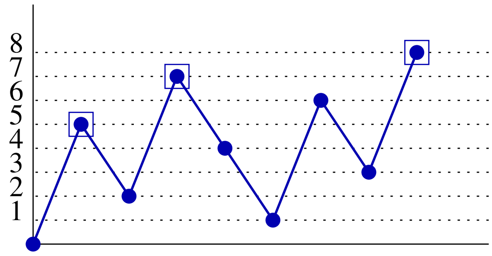


**Fig. 4:** When $a = 5$ and $b = 3$, there is no culminating walk of height $k$, for $k \in [\![1, 8]\!] \setminus \{5, 7, 8\}$. For $k = 5, 7, 8$, there is exactly one culminating walk.

**Proposition 3.1** *Let $a = b = 1$ and $k \geq 1$. The length generating function of culminating paths of height $k$ is*

$$C_k(t) = \frac{t^k}{F_{k-1}} = t\frac{U_1 - U_2}{U_1^k - U_2^k} = \frac{1-U^2}{1+U^2}\frac{U^k}{1-U^{2k}},$$

*where*

- *$F_k$ is the $k$th Fibonacci polynomial, defined by $F_0 = F_1 = 1$ and $F_k = F_{k-1} - t^2 F_{k-2}$ for $k \geq 2$,*
- *$U_1$ and $U_2$ are the two roots of the polynomial $u - t(1 + u^2)$:*

$$U_{1,2} = \frac{1 \mp \sqrt{1-4t^2}}{2t},$$

- *$U$ stands for any of the $U_i$'s.*

*The generating function of culminating walks,*

$$C(t) = \frac{1-U^2}{1+U^2}\sum_{k\geq 1}\frac{U^k}{1-U^{2k}}, \tag{1}$$

*is not D-finite.*

The above expression of $C(t)$ is equivalent to the case $x = y = 1$ of [18, Eq.(2.26)].

The first expression of $C_k$, in terms of the Fibonacci polynomials, is clearly rational. As explained in Section 2.1, the language of culminating walks of height $k$ is regular for all $a$ and $b$, so that the series $C_k$ will always be rational. Of course, $C_k$ is simply 0 when $k < a$. When $k = a$, there is only one culminating path, reduced to one up step, so that $C_k = t$. More generally, the following property, illustrated in Figure 4 and proved in Section 3.2.1, holds.

**Property 3.2** *For $k \leq a + b$, there is at most one culminating path of height $k$.*

As soon as $k > a$, culminating walks of height $k$ have at least two steps. Deleting the first and last ones gives $C_k = t^2 W_k$, where $W_k$ counts walks (with steps $+a, -b$) going from $a$ to $k - a$ on the segment $[\![1, k-1]\!]$. General (and basic) results on the enumeration of walks on a digraph provide [36, Ch. 4]:

$$C_k = t^2 W_k = t^2\left((1 - tA_k)^{-1}\right)_{a,k-a} = t^2\frac{N_k}{D_k}, \tag{2}$$

where $A_k = (A_{i,j})_{1\le i,j\le k-1}$ is the adjacency matrix of our segment graph:

$$A_{i,j} = \begin{cases} 1 & \text{if } j = i + a \text{ or } j = i - b, \\ 0 & \text{otherwise,} \end{cases} \tag{3}$$

$D_k$ is the determinant of $(1 - tA_k)$ and $N_k/D_k$ is the entry $(a, k - a)$ of $(1 - tA_k)^{-1}$.

We note from Proposition 3.1 that, in the $(1, 1)$-case, both $N_k$ and $D_k$ are especially simple. Indeed, $N_k = t^{k-2}$, while $D_k = F_{k-1}$ satisfies a linear recurrence relation (with constant coefficients) of order 2. We will prove that, for all $a$ and $b$, both sequences $N_k$ and $D_k$ satisfy such a recurrence relation (of a larger order in general). The monomial form of $N_k$ will hold as soon as $a = 1$.

The second expression of $C_k$ given in Proposition 3.1 appears as a rational function of the roots of the polynomial $u - t(1 + u^2)$. Even though both series $U_1$ and $U_2$ are algebraic (and irrational), the fact that $C_k$ is symmetric in $U_1$ and $U_2$ explains why $C_k$ itself is rational. In general, we will write $C_k$ as a *symmetric rational function* of the $a+b$ roots of the polynomial $u^b - t(1+u^{a+b})$, denoted $U_1, \dots, U_{a+b}$.

The third expression of $C_k$ follows from the fact that $U_1U_2 = 1$. In general, $t = U^b/(1 + U^{a+b})$ for $U = U_i$, so that it will always be possible to write $C_k$ as a rational function of $U$. However, this expression will not be always as simple as above. The equivalence of the three expressions of Proposition 3.1 follows easily from the fact that

$$F_k = \frac{1 - U^{2k+2}}{(1 - U^2)(1 + U^2)^k}.$$

This can be proved by solving the recurrence relation satisfied by the $F_k$'s — or can be checked by induction on $k$.

Let us now state our generalisation of Proposition 3.1 to $(a, b)$-culminating walks. Our first expression of $C_k$, namely the rational form (2), involves the evaluation of two determinants of size (approximately) $k$. Our second expression of $C_k$ will be a *fixed* rational function of $U_1, \dots, U_{a+b}, U_1^k, \dots, U_{a+b}^k$, symmetric in the $U_i$, which involves two determinants of constant size $a + b$. The existence of such smaller determinantal forms for walks confined in a strip has already been recognized in [3, Ch. 1]. More recently, the case of excursions confined in a strip has been simplified and worked out in greater detail [10]. As in [10], our results will be expressed in terms of the *Schur functions* $s_\lambda$, which form one of the most important bases of symmetric functions in $n$ variables $x_1, \dots, x_n$: for any integer partition $\lambda$ with at most $n$ parts, $\lambda = (\lambda_1, \dots, \lambda_n)$ with $\lambda_1 \ge \lambda_2 \ge \cdots \ge \lambda_n \ge 0$,

$$s_\lambda(\mathcal{X}) = \frac{a_{\delta+\lambda}}{a_\delta}, \tag{4}$$

with $\mathcal{X} = (x_1, \dots, x_n)$, $\delta = (n-1, n-2, \dots, 1, 0)$ and $a_\mu = \det\left(x_i^{\mu_j}\right)_{1\le i,j\le n}$. We refer to [37, Ch. 7] for generalities on symmetric functions.

**Proposition 3.3** *Let $k > a$. With the above notation, the length generating function of $(a, b)$-culminating paths of height $k$ admits the following expressions:*

$$C_k(t) = t^2\left((1 - tA_k)^{-1}\right)_{a,k-a} = t^2\frac{N_k}{D_k} = t\frac{s_\mu(\mathcal{U})}{s_\lambda(\mathcal{U})},$$

*where $A_k$ is given by (3), the $(a+b)$-tuple $\mathcal{U} = (U_1, \dots, U_{a+b})$ is the collection of roots of the polynomial $u^b - t(1 + u^{a+b})$, and the partitions $\lambda$ and $\mu$ are given by $\lambda = (k-1)^a$ and $\mu = ((k-1)^{a-1}, a-1)$.*

*The determinant $D_k$ of $(1 - tA_k)$ and the relevant cofactor $N_k$ are respectively given by*

$$D_k = (-1)^{(a-1)(k-1)} t^{k-1} s_\lambda(\mathcal{U}) \quad \textit{and} \quad N_k = (-1)^{(a-1)(k-1)} t^{k-2} s_\mu(\mathcal{U}). \tag{5}$$

*Both sequences $N_k$ and $D_k$ satisfy a linear recurrence relation with coefficients in $\mathbb{Q}[t]$, respectively of order $\binom{a+b}{a}$ and $\binom{a+b}{a-1}$. These orders are optimal.*

Note that the expression of $C_k$ in terms of Schur functions still holds for $k = a$. Examples will be given below. For the moment, let us underline that the case $a = 1$ of this proposition takes a remarkably simple form, which will be given a combinatorial explanation in Section 3.2.3.

**Corollary 3.4** *When $a = 1$, the generating function of culminating walks of height $k \geq 1$ reads*

$$C_k(t) = \frac{t^k}{D_k} = \frac{t}{h_{k-1}(\mathcal{U})},$$

*where $h_i$ is the complete homogeneous symmetric function of degree $i$, $D_k = 1$ for $1 \leq k \leq b+1$ and $D_k = D_{k-1} - t^{b+1} D_{k-b-1}$ for $k > b+1$.*

**Examples.** Let us illustrate Proposition 3.3 by writing down explicitly the expression of $C_k$ for a few values of $a$ and $b$. We use the determinantal form (4) of Schur functions.

**Case** $a = b = 1$**.** Here $U_1$ and $U_2$ are the two roots of the polynomial $u - t(1 + u^2)$. The partition $\mu$ is empty, so that $s_\mu = 1$, while $\lambda = (k - 1)$. This gives

$$C_k = t \frac{\begin{vmatrix} U_1 & 1 \\ U_2 & 1 \end{vmatrix}}{\begin{vmatrix} U_1^k & 1 \\ U_2^k & 1 \end{vmatrix}} = t \frac{U_1 - U_2}{U_1^k - U_2^k},$$

as in Proposition 3.1. The recurrence relations satisfied by the polynomials $N_k$ and $D_k$ can always be worked out from their expressions (5), as will be explained in Section 3.2.2. In the case $a = b = 1$, one finds

$$C_k = t^2 N_k / D_k \quad \text{with} \quad N_k = t^{k-2} \quad \text{and} \quad D_k = D_{k-1} - t^2 D_{k-2},$$

with initial conditions $D_1 = D_2 = 1$.

**Case** $a = 1, b = 2$**.** Here $U_1, U_2, U_3$ are the three roots of the polynomial $u^2 - t(1 + u^3)$. Again, $\mu$ is empty and $\lambda = (k - 1)$ (this holds as soon as $a = 1$). One obtains

$$C_k = t \frac{\begin{vmatrix} U_1^2 & U_1 & 1 \\ U_2^2 & U_2 & 1 \\ U_3^2 & U_3 & 1 \end{vmatrix}}{\begin{vmatrix} U_1^{k+1} & U_1 & 1 \\ U_2^{k+1} & U_2 & 1 \\ U_3^{k+1} & U_3 & 1 \end{vmatrix}}.$$

The rational expression of $C_k$ reads

$$C_k = t^2 N_k / D_k \quad \text{with} \quad N_k = t^{k-2} \quad \text{and} \quad D_k = D_{k-1} - t^3 D_{k-3},$$

with initial conditions $D_1 = D_2 = D_3 = 1$. Note that this expression allows us to compute in a few seconds the number $c_n$ of culminating walks for $n$ up to 500.

**Case** $a = 2, b = 1$. Here $U_1, U_2, U_3$ are the three roots of the polynomial $u - t(1 + u^3)$. One has $\mu = (k-1, 1)$ and $\lambda = (k-1)^2$, which gives:

$$C_k = t\frac{\begin{vmatrix} U_1^{k+1} & U_1^2 & 1 \\ U_2^{k+1} & U_2^2 & 1 \\ U_3^{k+1} & U_3^2 & 1 \end{vmatrix}}{\begin{vmatrix} U_1^{k+1} & U_1^k & 1 \\ U_2^{k+1} & U_2^k & 1 \\ U_3^{k+1} & U_3^k & 1 \end{vmatrix}} = t\frac{\begin{vmatrix} \bar{U}_1^{k+1} & \bar{U}_1^{k-1} & 1 \\ \bar{U}_2^{k+1} & \bar{U}_2^{k-1} & 1 \\ \bar{U}_3^{k+1} & \bar{U}_3^{k-1} & 1 \end{vmatrix}}{\begin{vmatrix} \bar{U}_1^{k+1} & \bar{U}_1 & 1 \\ \bar{U}_2^{k+1} & \bar{U}_2 & 1 \\ \bar{U}_3^{k+1} & \bar{U}_3 & 1 \end{vmatrix}},$$

where $\bar{U}_i := 1/U_i$. Note that the series $\bar{U}_i$ are the roots of the polynomial $u^2 - t(1 + u^3)$, which occurs in the (symmetric) case $a = 1, b = 2$. It is actually clear from (2) that the denominator $D_k$ is unchanged when exchanging $a$ and $b$.

The rational expression of $C_k$ reads

$$C_k = t^2 N_k / D_k \quad \text{with} \quad N_k = tN_{k-2} + t^3 N_{k-3} \quad \text{and} \quad D_k = D_{k-1} - t^3 D_{k-3},$$

with initial conditions $N_1 = 0, N_2 = 1/t, N_3 = t$ and $D_1 = D_2 = D_3 = 1$.

## 3.2 Proofs

### 3.2.1 Proof of Property 3.2

Let us say that a path is *positive* if every step ends at a positive level. For instance, culminating walks are positive. For $n \geq 0$ there exists a unique positive walk of length $n$ and height at most $a + b$, denoted $w_n$. Indeed, given $h \in [\![0, a + b]\!]$, exactly one of the values $h + a, h - b$ lies in the interval $[\![1, a + b]\!]$. For the same reason, $w_i$ is a prefix of $w_j$ for $i \leq j$. Let $k \leq a + b$, and assume that there exist two distinct culminating walks of height $k$. These walks must be $w_i$ and $w_j$, for some $i$ and $j$, with, say, $i < j$. But then $w_i$ is a prefix of $w_j$, and ends at height $k$, which prevents $w_j$ from being culminating. □

### 3.2.2 Proof of Proposition 3.3

The expression of $C_k$ in terms of the adjacency matrix $A_k$ has been justified in Section 3.1. Let us now derive the Schur function expression of this series. We will give actually two proofs of this expression: the first one is based on the *kernel method* [8, 4, 3], and the second one on the *Jacobi-Trudi identity.* The first proof is completely elementary. The second one allows us to relate the polynomials $N_k$ and $D_k$ to the Schur functions $s_\lambda$ and $s_\mu$. This derivation is very close to what was done in [10] for excursions confined in a strip. Some of the results of [10] will actually be used to shorten some arguments.

**First proof via the kernel method:** Consider a culminating walk of height $k > a$. Such a walk has length at least 2. Delete its first and last steps: this gives a walk starting from level $a$, ending at level $k - a$, and confined between levels 1 and $k - 1$. Shifting this walk one step down, we obtain a non-negative walk starting from level $a - 1$ and ending at level $k - 1 - a$, of height at most $k - 2$. Let $G(t, u) \equiv G(u)$ denote the generating function of non-negative walks starting from $a - 1$, of height at most $k - 2$. In this series, the variable $t$ keeps track of the length while the variable $u$ records the final height. Write $G(u) = \sum_{h=0}^{k-2} u^h G_h$, where $G_h$ counts walks ending at height $h$. The above argument implies that the generating function of culminating walks of height $k$ is

$$C_k = t^2 G_{k-a-1}. \tag{6}$$

We can construct the walks counted by $G(u)$ step by step, starting from height $a-1$, and adding at each time a step $+a$ (unless the current height is $k-a-1$ or more) or $-b$ (unless the current height is $b-1$ or less). In terms of generating functions, this gives:

$$G(u) = u^{a-1} + t(u^a + u^{-b})G(u) - tu^{-b}\sum_{h=0}^{b-1} u^h G_h - tu^a \sum_{h=k-a-1}^{k-2} u^h G_h,$$

that is,

$$\left(u^b - t(1+u^{a+b})\right) G(u) = u^{a+b-1} - t\sum_{h=0}^{b-1} u^h G_h - tu^{a+b} \sum_{h=k-a-1}^{k-2} u^h G_h.$$

The *kernel* of this equation, that is, the polynomial $u^b - t(1+u^{a+b})$, has $a+b$ distinct roots, which are Puiseux series in $t$. We denote them $U_1, \dots, U_{a+b}$. Recall that $G(u)$ is a *polynomial* in $u$ (of degree $k-2$). Replacing $u$ by each of the $U_i$ gives a system of $a+b$ linear equations relating the unknown series $G_0, \dots, G_{b-1}$ and $G_{k-a-1}, \dots, G_{k-2}$. For $U = U_i$, with $1 \le i \le a+b$,

$$\sum_{h=0}^{b-1} U^h G_h + U^{a+b} \sum_{h=k-a-1}^{k-2} U^h G_h = U^{a+b-1}/t.$$

In matrix form, we have $\mathcal{M}\mathcal{G} = \mathcal{C}/t$, where $\mathcal{M}$ is the square matrix of size $a+b$ given by

$$\mathcal{M} = \begin{pmatrix} U_1^{a+b+k-2} & U_1^{a+b+k-3} & \cdots & U_1^{b+k-1} & U_1^{b-1} & U_1^{b-2} & \cdots & 1 \\ U_2^{a+b+k-2} & \cdots & & & & & \cdots & 1 \\ \vdots & & & & & & & \vdots \\ U_{a+b}^{a+b+k-2} & U_{a+b}^{a+b+k-3} & \cdots & U_{a+b}^{b+k-1} & U_{a+b}^{b-1} & U_{a+b}^{b-2} & \cdots & 1 \end{pmatrix}, \tag{7}$$

$\mathcal{G}$ is the column vector $(G_{k-2}, \dots, G_{k-a-1}, G_{b-1}, \dots, G_0)$, and $\mathcal{C}$ is the column vector $(U_1^{a+b-1}, \dots, U_{a+b}^{a+b-1})$. In view of the definition (4) of Schur functions,

$$\det(\mathcal{M}) = s_\lambda(\mathcal{U}),$$

with $\lambda = (k-1)^a$. It has been shown in [10] that the generating function of excursions (walks starting and ending at 0) confined in the strip of height $k-2$ is

$$\frac{(-1)^{a+1}}{t} \frac{s_{(k-2)^a}(\mathcal{U})}{s_{(k-1)^a}(\mathcal{U})},$$

and that, in particular, $s_\lambda(\mathcal{U}) \neq 0$. Hence $\mathcal{M}$ is invertible, and applying Cramer's rule to the above system gives

$$G_{k-a-1} = \frac{1}{t} \frac{s_\mu(\mathcal{U})}{s_\lambda(\mathcal{U})},$$

with $\lambda$ and $\mu$ defined as in the statement of the proposition. Combining this with (6) gives the desired Schur function form of $C_k$. $\square$

**A second proof via symmetric functions:** Let us now give an alternative proof of the Schur function expression of $C_k$. It will be based on the dual Jacobi-Trudi identity, which expresses Schur functions as a determinant in the elementary symmetric functions $e_i$ [37, Cor. 7.16.2]: for any partition $\nu$,

$$s_\nu = \det\left(e_{\nu'_j+i-j}\right)_{1\le i,j\le \nu_1}, \tag{8}$$

where $\nu'$ is the conjugate partition of $\nu$.

Let us consider the identity (2), with $D_k = \det(1-tA_k)$. It turns out that this determinant *is* of the form (8). Indeed, let us define $V_i = -U_i$, for $1 \le i \le a+b$. Then the only elementary symmetric functions of the $V_i$ that do not vanish are $e_0(\mathcal{V}) = 1, e_a(\mathcal{V}) = -1/t$ and $e_{a+b}(\mathcal{V}) = 1$ (with $\mathcal{V} = (V_1, \dots, V_{a+b})$). Let us apply (8) to $\nu = \lambda = (k-1)^a$, with variables $V_1, \dots, V_{a+b}$. Then $\nu' = a^{k-1}$ and one obtains

$$s_\lambda(\mathcal{V}) = (-t)^{-(k-1)} D_k = (-1)^{a(k-1)} s_\lambda(\mathcal{U}),$$

since $s_\lambda$ is homogeneous of degree $a(k-1)$. This gives the Schur function expression of $D_k$.

Now, by the general inversion formula for matrices, $N_k = (-1)^k \det((1-tA_k)^{k-a,a})$, where $(1-tA_k)^{k-a,a}$ is obtained by deleting row $k-a$ and column $a$ from $(1-tA_k)$. Let us apply (8) to $\nu = \mu = ((k-1)^{a-1}, a-1)$. Then $\nu' = a^{a-1}(a-1)^{k-a}$. The matrix $\left(e_{\nu'_j+i-j}\right)$ has size $k-1$, and its last column contains only one non-zero entry (equal to $e_0(\mathcal{V}) = 1$), in row $k-a$. After deleting this row and the last column, one obtains:

$$s_\mu(\mathcal{V}) = (-1)^{a-1}(-t)^{-(k-2)} \det((1-tA_k)^{k-a,a}) = (-1)^{a-1} t^{-(k-2)} N_k = (-1)^{k(a-1)} s_\mu(\mathcal{U}),$$

as $s_\mu$ is homogeneous of degree $k(a-1)$. This gives the desired expression of $N_k$. $\square$

**Linear recursions.** Finally, let us prove that the sequences of polynomials $N_k$ and $D_k$ satisfy a linear recurrence relation with coefficients in $\mathbb{Q}[t]$, the ring of polynomials in $t$. Equivalently, we prove that each of the generating functions

$$N(z,t) := \sum_{k\ge a} N_k z^k \quad \text{and} \quad D(z,t) := \sum_{k\ge a} D_k z^k$$

is actually a rational function in $z$ and $t$. The existence of a linear recursion then easily follows by the general theory of rational series [36, Ch. 4].

Given the expression (5) of $N_k$, what we have to do is to evaluate

$$N'(z; u_1, \dots, u_{a+b}) := \sum_{k\ge a} s_{(k-1)^{a-1}, a-1} z^k$$

where the symmetric functions involve the $a+b$ indeterminates $u_1, \dots, u_n$, with $n = a+b$. We use the definition (4) of Schur functions to write $s_{(k-1)^{a-1},a-1}$ as a ratio of determinants of size $n$. The determinant occurring at the denominator is the Vandermonde $V_n$ in the $u_i$'s, and is independent of $k$.

The determinant at the numerator is obtained from (7) by replacing the column containing $U_i^{b+k-1}$ by a column of $U_i^{a+b-1}$ (and then each $U_i$ by the indeterminate $u_i$). We expand it as a sum over permutations of length $n$, and obtain:

$$\begin{aligned} N'(z;u) &= \frac{1}{V_n}\sum_{k\geq a} z^k \sum_{\sigma\in\mathfrak{S}_n} \varepsilon(\sigma)\, \sigma\left(u_1^{n+k-2}\cdots u_{a-1}^{b+k}u_a^{a+b-1}u_{a+1}^{b-1}\cdots u_{n-1}^1 u_n^0\right) \\ &= \frac{1}{V_n}\sum_{\sigma\in\mathfrak{S}_n} \varepsilon(\sigma)\, \sigma\left(\frac{u_1^{n+a-2}\cdots u_{a-1}^{a+b}u_a^{a+b-1}u_{a+1}^{b-1}\cdots u_{n-1}^1 u_n^0}{1-zu_1\cdots u_{a-1}}\right), \end{aligned}$$

where $\sigma$ acts on functions of $u_1,\ldots,u_n$ by permuting the variables:

$$\sigma F(u_1,\ldots,u_n) = F(u_{\sigma(1)},\ldots,u_{\sigma(n)}).$$

Equivalently,

$$N'(z;u) = \frac{P(z;u)}{Q(z;u)}$$

where

$$Q(z;u) = \sum_{I\subset[\![n]\!],\ |I|=a-1}\left(1-z\prod_{i\in I}u_i\right)$$

and $P(z;u)$ is another polynomial in $z$ and the $u_i$, symmetric in the $u_i$'s. This symmetry property shows that replacing $u_i$ by $U_i$ transforms $N'(z;u)$ into a rational series in $z$ and $t$. The link between $N_k$ and $s_{(k-1)^{a-1},a-1}$ then gives

$$N(z,t) = \frac{(-1)^{a-1}P((-1)^{a-1}tz;\mathcal{U})}{t^2\, Q((-1)^{a-1}tz;\mathcal{U})},$$

another rational function of $z$ and $t$. A similar argument, given explicitly in [10], yields

$$D(z,t) = \frac{(-1)^{a-1}\tilde{P}((-1)^{a-1}tz;\mathcal{U})}{t\,\tilde{Q}((-1)^{a-1}tz;\mathcal{U})},$$

for two polynomials $\tilde{P}$ and $\tilde{Q}$ in $z$ and $u_1,\ldots,u_n$. More precisely,

$$\tilde{Q}(z;u) = \sum_{I\subset[\![n]\!],\ |I|=a}\left(1-z\prod_{i\in I}u_i\right).$$

By looking at the degree of $\tilde{Q}$ and $Q$, this establishes the existence of recurrence relations of order $\binom{a+b}{a-1}$ for $N_k$, and $\binom{a+b}{a}$ for $D_k$. If there were recursions of a smaller order, the polynomials $Q(z;\mathcal{U})$ or $\tilde{Q}(z;\mathcal{U})$ would factor. It has been shown in [10, Section 6] that $\tilde{Q}(z;\mathcal{U})$ is irreducible, and the same argument implies that $Q(z;\mathcal{U})$ is irreducible as well. $\square$

### 3.2.3 Two proofs of Corollary 3.4

Let us specialize Proposition 3.3 to the case $a = 1$. We observe that $\mu$ is the empty partition, so that $s_\mu = 1$, while $\lambda = (k-1)$, so that $s_\lambda = h_{k-1}$. The expressions (5) of $N_k$ and $D_k$ in terms of Schur functions give $N_k = t^{k-2}$ and $D_k = t^{k-1}h_{k-1}(\mathcal{U})$. Observe that $e_1(\mathcal{U}) = 1/t$ and $e_{b+1}(U) = (-1)^{b+1}$. The classical relation between elementary and complete symmetric functions [37, Eq. (7.13)] gives, for $k \geq 1$,

$$h_k(\mathcal{U}) = \frac{1}{t}\, h_{k-1}(\mathcal{U}) - h_{k-b-1}(\mathcal{U}),$$

with initial conditions $h_0 = 1$ and $h_i = 0$ for $i < 0$. This gives the desired recursion for $D_k$.

Let us now justify combinatorially the simplicity of $N_k$ and $D_k$. Recall that, for $k \geq 2$, one has $C_k = t^2 W_k$, where $W_k$ counts walks (with steps +1, $-b$) going from 1 to $k-1$ on the segment graph $[\![1, k-1]\!]$. The adjacency matrix of this graph is $A_k$. The combinatorial description(i) of the inverse of the matrix $(1 - tA_k)$ tells us that $D_k$ counts non-intersecting collections of elementary cycles on the segment $[\![1, k-1]\!]$, while $N_k$ counts configurations formed of a self-avoiding path $w$ going from 1 to $k-1$ together with a non-intersecting collection of elementary cycles that do not meet $w$. In the polynomials $N_k$ and $D_k$, each cycle of length $\ell$ is given a weight $(-t^\ell)$ while the path $w$ is simply weighted $t^\ell$ if it has length $\ell$. This gives directly $N_k = t^{k-2}$, as the only possible path $w$ is formed of $k-2$ up steps, and leaves no place to co-existing cycles. Now the only elementary cycles are formed of $b$ up steps and one down step $-b$. The recursion satisfied by $D_k$ is then obtained by discussing whether the point $k-1$ is contained in one such cycle.

Note that this proof can be rephrased in terms of *heaps of cycles* using Viennot's correspondence between walks on a graph and certain heaps [38]. The expression $N_k/D_k$ then appears as a specialization of the *inversion lemma* (also found in [38]). In particular, $D_k$ is the (alternating) generating function of *trivial* heaps of cycles. □

**Remark.** For general values of $a$ and $b$, the description of $D_k$ and $N_k$ in terms of cycles and paths on the graph $[\![1, k-1]\!]$ remains perfectly valid. But the structure of elementary cycles and self-avoiding paths becomes more complicated. See an example in Figure 5.

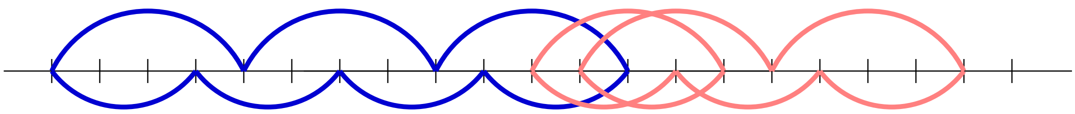

**Fig. 5:** Two non-intersecting elementary cycles (for $a = 4$ and $b = 3$).

### 3.2.4 Proof of Proposition 3.1

The expression of $C_k$ is just a specialization of Corollary 3.4 to the case $b = 1$. It remains to prove that the series $C(t)$ is not D-finite.

Let us first observe that $C(t)$ is D-finite if and only if the power series (in $u$) $B(u) := \sum_k u^k/(1-u^{2k})$ is D-finite. Indeed, one goes from $C(t)$ to $B(u)$, and vice-versa, by an algebraic substitution of the

(i) This description seems to have been around since, at least, the 80's [25, 38]. See [9, Thm. 2.1] for a modern formulation.

variable, as $U$ is an algebraic function of $t$ and $t = U/(1+U^2)$. It is known that D-finite series are preserved by algebraic substitutions [37, Thm. 6.4.10], so that we can now focus on the series $B(u)$.

This series has integer coefficients, and radius of convergence 1. Hence it is either rational, or admits the unit circle as a natural boundary [12]. As will be recalled later (10), the singular behaviour of $B(u)$ as $u$ approaches 1 involves a logarithm, which rules out the possibility of $B(u)$ being rational. Thus $B(u)$ has a natural boundary, and, in particular, infinitely many singularities. But D-finite series have only finitely many singularities, so that $B(u)$ is not D-finite. □

# 4 Asymptotic enumerative results

In this section we present some results on the asymptotic enumeration of culminating walks. Intuitively, three cases arise, depending on the *drift* of the walks, defined as the difference $a - b$. Indeed, an $n$-step random walk of positive drift is known to end at level $O(n)$ and is, intuitively, quite likely to be culminating. On the contrary, walks with a negative drift have a very small probability of staying positive. We first work out the intermediate case of a zero drift.

## 4.1 Walks with a null drift ($a = b = 1$)

When the drift is zero, the number of *positive* walks (walks in which every step ends at a positive level) of length $n$ is known to be asymptotically equivalent to $2^n/\sqrt{2\pi n}$. The average height, and the average final level of these walks both scale like $\sqrt{n}$. Hence we can expect the number of culminating walks to be of the order of $2^n/n$. This is confirmed by the following result.

**Proposition 4.1** *As $n \to \infty$, the number of $(1,1)$-culminating paths of length $n$ is asymptotically equivalent to $2^n/(4n)$.*

**Proof:** We start from the expression (1) of $C(t)$, with $U = U_1 = O(t)$, and apply the *singularity analysis* of [23]. Note that $U(t)$ is an odd function of $t$. Let us first study the even part of $C(t)$, which counts culminating paths of even length:

$$C_e(t) = \frac{1-U^2}{1+U^2} \sum_{k\geq 1} \frac{U^{2k}}{1-U^{4k}}.$$

Let $Z \equiv Z(x)$ be such that $U(t)^2 = Z(t^2)$. That is,

$$Z \equiv Z(x) = \frac{1-2x-\sqrt{1-4x}}{2x}.$$

The equation $U = t(1+U^2)$ gives $Z = x(1+Z)^2$. Moreover, we have $C_e(t) = D(t^2)$ where

$$D(x) = \frac{1-Z}{1+Z} \sum_{k\geq 1} \frac{Z^k}{1-Z^{2k}}.$$

We thus need to study the asymptotic behaviour of the coefficients of $D(x)$. We write

$$D(x) = S(Z(x)), \quad \text{with} \quad S(z) = \frac{1-z}{1+z} \sum_{k\geq 1} \frac{z^k}{1-z^{2k}}.$$

The series $Z(x)$ has radius of convergence $1/4$. It is analytic in the domain $\mathcal{D} = \mathbb{C} \setminus [1/4, +\infty)$, with exactly one singularity, at $x = 1/4$. One has $Z(0) = 0$, and $|Z(x)| < 1$ for all $x$ in $\mathcal{D}$. Indeed, assume $|Z(x)| \geq 1$ for some $x$ in $\mathcal{D}$. By continuity, $Z(x) = e^{i\theta}$ for some $x$ in $\mathcal{D}$. From the equation $x(1+Z)^2 = Z$, we conclude that $\theta \in (-\pi, \pi)$ (for $\theta = \pm\pi$, we would have $Z = -1 = 0$), and that $x = 1/(4\cos^2(\theta/2))$. But this contradicts the fact that $x \in \mathcal{D}$.

The series $S(z)$ has radius of convergence 1. Given that $|Z(x)| < 1$ in $\mathcal{D}$, this implies that $D(x) = S(Z(x))$ is analytic in the domain $\mathcal{D}$. It remains to understand how $D(x)$ behaves as $x$ approaches $1/4$ in $\mathcal{D}$.

Take $x = (1 - re^{i\theta})/4$, with $0 < r < 1$ and $|\theta| < \pi$. Then

$$Z(x) = 1 - 2\sqrt{1-4x} + O(1-4x) = 1 - 2\sqrt{r}e^{i\theta/2} + O(r).$$

In particular,

$$\arg(1 - Z(x)) = \theta/2 + O(\sqrt{r}).$$

Choose $\alpha \in (\pi/4, \pi/2)$. The above identity shows that there exists $\eta > 0$ and $\pi/2 < \phi < \pi$ such that, in the indented disk

$$\mathcal{I} = \{x : \ |1-4x| < \eta \text{ and } |\arg(1-4x)| < \phi\},$$

one has

$$|\arg(1 - Z(x))| < \alpha. \tag{9}$$

Now when $z \to 1$ in such a way that $|\arg(1-z)| < \alpha$,

$$\sum_{k\geq 1} \frac{z^k}{1-z^{2k}} \sim \frac{1}{2(1-z)} \log \frac{1}{1-z}, \quad \text{so that} \quad S(z) \sim \frac{1}{4} \log \frac{1}{1-z}. \tag{10}$$

This can be obtained using a Mellin transform or some already known results on the generating function of divisor sums [22].

Combining (9) and (10) shows that, as $x$ tends to $1/4$ in the indented disk $\mathcal{I}$,

$$D(x) = S(Z(x)) \sim \frac{1}{8} \log \frac{1}{1-4x}. \tag{11}$$

This allows us to apply the transfer theorems of [23]. Indeed, the series $D(x)$ is analytic in the following domain:

$$\Delta = \{x \neq 1/4 : |4x| < 1 + \eta \ \text{ and } \ |\arg(1-4x)| < \phi\},$$

with singular behaviour near $x = 1/4$ given by (11). From this we conclude that the coefficient of $x^n$ in $D(x)$ is asymptotically equivalent to $4^n/(8n)$. Going back to the series $C_e(t)$, this means that the number of culminating paths of (even) length $N = 2n$ is asymptotically equivalent to $2^N/(4N)$.

The study of the odd part of $C(t)$ is similar. $\square$

## 4.2 Walks with positive drift ($a > b$)

When the drift is positive, it is known that, asymptotically, a positive fraction of walks with steps $+a$, $-b$ is actually *positive* (every step ends at a positive level). More precisely, as $n \to \infty$, the number $p_n^{a,b}$ of positive walks of length $n$ satisfies

$$p_n^{a,b} \sim \kappa_{a,b}.2^n \tag{12}$$

for some positive constant $\kappa_{a,b}$. We will show that the culmination and final record conditions play similar filtering roles in the paths of $\{m, \overline{m}\}^*$, and prove the following result.

**Proposition 4.2** *For $a > b$, the number $c_n^{a,b}$ of culminating walks of length $n$ satisfies*

$$c_n^{a,b} = \kappa_{a,b}^2.2^n + O(\rho^n),$$

*where $\rho < 2$ and $\kappa_{a,b}$ is the constant involved in the asymptotics of positive walks.*

**Proof:** In what follows, we consider two families of paths that are close to the meanders and excursions defined in the introduction: the (already defined) *positive* walks, and certain *quasi-excursions*. The exact and asymptotic enumeration of meanders and excursions has been completely worked out in [5], and we will rely heavily on this paper. For instance, the estimate (12) follows from the results of [5] by noticing that a meander factors into an excursion followed by a positive walk. Let us call *quasi-excursion* a walk in which every step, except the final one, ends at a positive level. For instance, if $a = 3$ and $b = 2$, the word $m\overline{m}\overline{m}$ is a quasi-excursion. By removing the last step of such a walk, we see that quasi-excursions are in bijection with positive walks of final height 1, 2, ..., or $b$. We denote the number of quasi-excursions of length $n$ by $e_n^{a,b}$. Using the results of [5], it is easy to see that, when the drift is positive, quasi-excursions are exponentially rare among general walks. That is, there exists $\mu < 2$ such that for $n$ large enough,

$$e_n^{a,b} < \mu^n. \tag{13}$$

From now on, we drop the superscripts $a$ and $b$, writing for instance $c_n$ rather than $c_n^{a,b}$. For any word $w = w_1 \cdots w_k$, denote by $\overleftarrow{w}$ the mirror image of $w$, that is, $\overleftarrow{w} = w_k \cdots w_1$. Let $u$ be a culminating word of length $n$, and write $u = vw$, where the word $v$ (resp. $w$) has length $\lfloor n/2 \rfloor$ (resp. $\lceil n/2 \rceil$). Then both $v$ and $\overleftarrow{w}$ are positive walks, and this proves that

$$c_n \leq p_{\lfloor n/2 \rfloor} p_{\lceil n/2 \rceil}. \tag{14}$$

Conversely, let us bound the number of pairs $(v, w)$, where $v$ and $w$ are positive walks of respective lengths $\lfloor n/2 \rfloor$ and $\lceil n/2 \rceil$, such that the word $u = v\overleftarrow{w}$ is *not* culminating. This means that

- either $u$ factors as $v_1 w_1$, where $v_1$ is a quasi-excursion of length $i > \lfloor n/2 \rfloor$,
- or, symmetrically, $u$ factors as $v_2 \overleftarrow{w_2}$ where $w_2$ is a quasi-excursion of length $j > \lceil n/2 \rceil$.

This implies that

$$p_{\lfloor n/2 \rfloor} p_{\lceil n/2 \rceil} - c_n \leq 2 \sum_{i=\lfloor n/2 \rfloor}^{n} e_i 2^{n-i}.$$

In view of (13), we have, for $n$ large enough:

$$p_{\lfloor n/2\rfloor}p_{\lceil n/2\rceil} - c_n \leq 2\sum_{i=\lfloor n/2\rfloor}^{n} \mu^i 2^{n-i} \leq \frac{2}{1-\mu/2}\, 2^n(\mu/2)^{\lfloor n/2\rfloor} \leq \frac{4}{1-\mu/2}\,(2\mu)^{\lfloor n/2\rfloor}.$$

Combining this with (14) and the known asymptotics for the numbers $p_n$ gives the expected result. □

## 4.3 Walks with negative drift ($a < b$): exponential decay

When the drift is negative, it is known that positive walks are exponentially rare among general walks. Indeed, there exist constants $\kappa_{a,b} > 0$ and $\alpha_{a,b} \in (1,2)$, such that

$$p_n^{a,b} \sim \kappa_{a,b}\frac{\alpha_{a,b}^n}{n^{3/2}}.$$

More precisely,

$$\alpha_{a,b} = \frac{a+b}{\sqrt[a+b]{a^a b^b}} = \frac{1+q}{\sqrt[1+q]{q^q}} \equiv \alpha(q), \tag{15}$$

where $q = a/b < 1$. We show below that the constant $\alpha_{a,b}$ also governs the number of culminating walks of size $n$.

**Proposition 4.3** *For $a < b$, the number $c_n^{a,b}$ of culminating walks of length $n$ satisfies*

$$c_n^{a,b} = O\left(\frac{\alpha_{a,b}^n}{n^3}\right), \tag{16}$$

*where $\alpha_{a,b}$ is given above. Moreover,*

$$\lim_{n\to\infty}\left(c_n^{a,b}\right)^{1/n} = \alpha_{a,b}.$$

**Proof:** The inequality (14) still holds, and gives the upper bound (16) on the number of culminating paths.

Let us now prove that the growth constant of culminating walks is still $\alpha_{a,b}$ by constructing a large class of such walks. Let $\mathcal{E}_n$ be the set of excursions of length $n$ (from now on, we drop the superscripts $a$ and $b$). Such excursions only exist when $n$ is a multiple of $a+b$, and the number $e_n$ of such walks then satisfies

$$e_n \sim \kappa\alpha_{a,b}^n n^{-3/2}$$

for some positive constant $\kappa$. It is known that random $(a,b)$-excursions of length $n$ converge in law to the Brownian excursion, after normalising the length by $n$ and the height by $\kappa'\sqrt{n}$, for some constant $\kappa'$ depending on $a$ and $b$ [29]. This implies that the (normalized) height of a discrete excursion converges in law to the height of the Brownian excursion (described by a theta distribution). In particular, the probability $p_n$ that an excursion of $\mathcal{E}_n$ has height larger than $\sqrt{n}$ tends to a limit $p < 1$ as $n$ goes to infinity. Take an excursion of $\mathcal{E}_k$ of height less than $\sqrt{n}$, with

$$k = (a+b)\left\lfloor\frac{n-1-\sqrt{n}}{a+b}\right\rfloor$$

and append one up step at its left, and $n-k-1$ up steps at its right: this gives a culminating walk of length $n$, which proves that

$$c_n \geq e_k(1-p_k).$$

Taking $n$th roots gives the required lower bound on the growth of $c_n$. □

Hence there are exponentially few walks of size $n$ with steps $+a$, $-b$ that are culminating. It is likely that $c_n$ behaves like $\alpha_{a,b}^n n^{-3-\gamma}$, for some $\gamma \geq 0$ that remains to be determined. Note that the final height of an $n$-step meander is known to have a discrete limit law as $n \to \infty$ [5].

# 5 Random generation of positive walks

The random generation of positive walks will be a preliminary step in some of the algorithms we present in the next section for the generation of culminating walks. The main ideas underlying the generation are the same for both classes of walks, but the class of positive walks is simpler. We apply three different approaches to their random generation: *recursive methods* (two versions), *anticipated rejection*, and *Boltzmann sampling*. The choice of the best algorithm depends on the drift, as summarized in the top part of Table 2. We denote by $\mathcal{P}^{a,b}$ the language of positive walks, but the superscript $a, b$ will often be dropped.

## 5.1 Recursive step-by-step approach

The first approach we present is elementary: we construct positive walks step-by-step, choosing at each time an up or down step *with the right probability*. This is the basis of the recursive approach introduced in [39]. Here are the three ideas underlying the algorithm:

- Let $\mathcal{W}$ be a language, and let $\mathcal{W}_p$ denote the language of the prefixes of words of $\mathcal{W}$. Assume that for all $w \in \mathcal{W}_p$ such that $|w| \leq n$, we know the number $N_w(n)$ of words of $\mathcal{W}$ of length $n$ beginning with $w$ (we call these word *extensions* of $w$). Then it is possible to draw uniformly words of length $n$ in $\mathcal{W}$ as follows. One starts from the empty word, and adds steps incrementally. If at some point the prefix that is built is $w$, one adds the letter $x$ to $w$ with probability $N_{wx}(n)/N_w(n)$.
- When $\mathcal{W} = \mathcal{P}^{a,b}$, the number of extensions of length $n$ of a prefix $w \in \mathcal{W}_p$ depends only on two parameters:
  - the length difference $i = n - |w|$,
  - the final height of $w$, $j = \phi_{a,b}(w)$,
- Let $p_{i,j}$ be the number of extensions of length $n$ of such a prefix $w$. The numbers $p_{i,j}$ obey the following recurrence:

$$\begin{array}{rcll} p_{i,j} &=& p_{i-1,j+a} + \mathbb{1}_{j>b}\, p_{i-1,j-b} & \text{for } i \geq 1, \\ p_{0,j} &=& 1. & \end{array}$$

As the two parameters $i$ and $j$ are bounded by $n$ and $an$ respectively, the precomputation of the numbers $p_{i,j}$ takes $O(n^2)$ arithmetic operations and requires to store $O(n^2)$ numbers. Then, the generation of a random word of length $n$ can be performed in linear time. However, one should take into account the cost due to the size of the numbers in the precomputation stage. Indeed, the numbers $p_{i,j}$ are exponential in $n$, so that the actual time-space complexity for this stage may grow to $O(n^3)$. However, using a floating-point technique adapted from [16], it should be possible to take advantage of the numerical stability of the algorithm to reduce the space needed to $O(n^{2+\varepsilon})$.

This naive recursive approach is less efficient than the one presented below, which is based on context-free grammars. But it will be easily adapted to the generation of culminating walks, which cannot be generated via a grammar, as was proved in Section 2.

## 5.2 Recursive approach via context-free grammars

It is easy to see that the language $\mathcal{P}^{a,b} \equiv \mathcal{P}$ is recognized by a non-deterministic push-down automaton. This implies that $\mathcal{P}$ is context-free. The same holds for the language $\mathcal{D}^{a,b} \equiv \mathcal{D}$ of excursions. A non-ambiguous context-free grammar generating excursions is given explicitly in [19]. It suffices to add one equation to obtain a non-ambiguous grammar generating positive walks:

$$\begin{array}{rclcrcl} \mathcal{D} &=& \varepsilon + \sum_{k=1}^{a} \mathcal{L}_k \mathcal{R}_k, & \qquad & \mathcal{L}_i &=& \mathbb{1}_{i=a} m\, \mathcal{D} + \sum_{k=i+1}^{a} \mathcal{L}_k \mathcal{R}_{k-i}, \\ \mathcal{P} &=& \varepsilon + \sum_{i=1}^{a} \mathcal{L}_i \mathcal{P}, & & \mathcal{R}_j &=& \mathbb{1}_{j=b} \overline{m}\, \mathcal{D} + \sum_{k=j+1}^{b} \mathcal{L}_{k-j} \mathcal{R}_k. \end{array} \tag{17}$$

In this system, $\varepsilon$ is the empty word, $\mathcal{D}$ (resp. $\mathcal{P}$) is the language of excursions (resp. positive walks) while $\mathcal{L}_i$, $1 \le i \le a$ and $\mathcal{R}_j$, $1 \le j \le b$, are $a + b$ auxiliary languages defined in [19]. As above, $m$ and $\overline{m}$ are the up and down letters in our alphabet.

From this grammar, we can apply the recursive approach of [24] for the uniform generation of *decomposable* objects, implemented in the `combstruct` package of Maple or in the stand-alone software `GenRGenS` [35]. The generation of positive walks of size $n$ begins with the precomputation of $O(n)$ large numbers. These numbers count words of length $r$, for all $r \le n$, in each of the languages involved in the grammar. The fastest way to get them is to convert the algebraic system (17) into a system of linear differential equations, which, in turn, yields a system of linear recurrence relations (with polynomial coefficients) defining the requested numbers. This step requires a linear number of arithmetic operations. But one has to multiply numbers whose size (number of digits) is $O(n)$, which may result, in practice, in a quadratic time-complexity for the precomputation stage. Then, the generation of a random positive walk can be performed in time $O(n \log n)$.

Note that a careful implementation [15] of the floating point approach of [16] using an arbitrary-precision floating-point computation library yields a $O(n^{1+\varepsilon})$ complexity after a $O(n^{1+\varepsilon})$ precomputation.

## 5.3 Anticipated rejection

The principle of this approach is to start with an empty walk, and then add successive up and down steps by flipping an unbiased coin until the walk reaches the desired length $n$, or a non-positive ordinate. In this case, the walk is rejected and the procedure starts from the beginning. Of course, no precomputation nor non-linear storage is required. This principle was applied to meanders, in the case $a = b = 1$, in [6], as a first step towards the uniform random generation of *directed animals*. The analysis of this algorithm yielded a linear time-complexity, later generalized in [7] to the case of coloured walks, in which up, down, and level steps come respectively in $p$, $q$ and $r$ different colours. There, it was shown that the time-complexity is linear when $p \ge q$, but exponential when $p < q$.

Unsurprisingly, we obtain similar results for the general $(a, b)$-case.

**Proposition 5.1** *The anticipated rejection scheme applied to the uniform random generation of* $(a,b)$*-positive walks has a linear time-complexity when* $a \geq b$ *and an exponential complexity in* $\Theta((2/\alpha_{a,b})^n n\sqrt{n})$ *when* $a < b$*, with* $\alpha_{a,b} = \frac{a+b}{\sqrt[a+b]{a^a b^b}} < 2$.

**Proof:** We first note that the language $\mathcal{P}$ of positive walks is a *left-factor language*. That is, it is stable by taking prefixes, and every word of $\mathcal{P}$ is the proper prefix of another word of $\mathcal{P}$. It has been proved in [14] that the average complexity $f_{\mathcal{L}}(n)$ of the anticipated rejection scheme for a left-factor language $\mathcal{L}$ on a $k$-letter alphabet is

$$f_{\mathcal{L}}(n) = \frac{[z^n]\frac{z}{1-z}L(z/k)}{[z^n]L(z/k)}$$

where $L(z)$ is the length generating function of the words of $\mathcal{L}$.

We now exploit the results of [5], giving the singular behaviour of the series $M(z)$ and $E(z)$ that count respectively meanders and excursions. As a meander factors uniquely as an excursion followed by a positive walk, we can derive from [5] the singular behaviour of the series $P(z) = \sum p_n z^n$ that counts positive walks. This series is always algebraic, so that singularity analysis applies.

- For $a \geq b$, the series $P(z/2)$ has an algebraic singularity at $z = 1$ in $(1-z)^{-\nu}$ (with $\nu = 1$ if $a > b$, and $\nu = 1/2$ if $a = b$). Thus $P(z/2)/(1-z)$ has a singular behaviour in $(1-z)^{-\nu-1}$. A singularity analysis gives $f_{\mathcal{P}}(n) \sim n/\nu$.
- For $a < b$, the series $P(z/2)$ has a square-root singularity at $2/\alpha_{a,b} > 1$, but $P(z/2)/(1-z)$ has a smaller radius of convergence $z_c = 1$, with a simple pole at this point. This gives

$$f_{\mathcal{P}}(n) \sim \frac{2^n\, P(1/2)}{p_n} \sim \kappa \left(\frac{2}{\alpha_{a,b}}\right)^n n\sqrt{n}$$

for some constant $\kappa$.

□

## 5.4 Boltzmann sampling

A Boltzmann generator [20] generates every object in the class $\mathcal{C}$ with a probability proportional to $x^n$, where $n$ is the size of the object. More precisely, for every object $w$ (a walk, in our context):

$$\mathbb{P}(w) = \frac{x^{|w|}}{C(x)}$$

where $C(x)$ is the generating function of the objects of $\mathcal{C}$. Of course, this results in a relaxation of the size constraint, since objects of all sizes can be generated. But, by tuning carefully the parameter $x$ (which has to be smaller than or equal to the radius of convergence of $C(x)$), and rejecting the too large and too small objects, one can often achieve an approximate-size random sampling, with a *tolerance* $\varepsilon$, in linear time. This means that after a linear number of real-arithmetic operations, and a number of attempts that is constant on average, the algorithm will produce an object of size $|w| \in [(1-\varepsilon)n, (1+\varepsilon)n]$, which is uniform among the objects of the same size.

In particular, the grammar (17) shows that the class of positive walks is *specifiable* in the sense of [20]. The analysis of the generating functions of meanders and excursions performed in [5] shows that the series $P(z)$ counting positive walks is always analytic in a $\Delta$-domain, with a dominant singularity in $(1-\mu t)^{-\nu}$, where $\nu = 1$ if $a > b$, $\nu = 1/2$ if $a = b$ and $\nu = -1/2$ if $a < b$. In the first two cases, Theorem 6.3 of [20] gives an approximate sampling in linear time (and an exact sampling in quadratic time). In the third case, the standard deviation of the objects produced by a standard Boltzmann sampler is much larger than their mean, which makes rejection costly. However, we can generate instead *pointed* positive walks, that is, positive walks with a distinguished step, and forget the pointing: as guaranteed by Theorem 6.5 of [20], this gives again an approximate sampling in linear time.

To conclude, the uniform random generation of $(a, b)$-positive walks of size $n$ can be performed in linear time when $a \geq b$ by an anticipated rejection, and this strategy does not require any precomputations nor storage. When $a < b$, our best algorithm for exact sampling remains the recursive approach based on the grammar (17). It runs in $O(n^{1+\varepsilon})$ after a $O(n^{1+\varepsilon})$ precomputation. However, one can achieve, in linear time and space, an approximate-size sampling using a Boltzmann generator.

# 6 Random generation of culminating walks

## 6.1 Recursive step-by-step approach

This elementary procedure, introduced in [30], generates culminating walks step by step, choosing every new step with the right probability. This is again an instance of Wilf's recursive method. The arguments given in Section 5.1 for positive walks should now be replaced by the following ones:

- For $\mathcal{W} = \mathcal{C}^{a,b}$, the number of extensions of length $n$ of a prefix $w \in \mathcal{W}_p$ depends only on three parameters:
  - the length difference $i = n - |w|$,
  - the final height $j = \phi_{a,b}(w)$,
  - the maximal height $h$ reached by $w$.
- Let $c_{i,j,h}$ be the number of extensions of length $n$ of such a prefix $w$. The numbers $c_{i,j,h}$ obey the following recurrence:

$$\begin{aligned} c_{i,j,h} &= c_{i-1,j+a,\max(h,j+a)} + \mathbb{1}_{j>b}\, c_{i-1,j-b,h} \quad \text{for } i > 1, \\ c_{1,j,h} &= \mathbb{1}_{j+a>h}. \end{aligned}$$

As the parameters $i$, $j$ and $h$ are bounded by $n$, $an$ and $an$ respectively, the precomputation of the numbers $c(i, j, h)$ takes $O(n^3)$ arithmetic operations and requires to store $O(n^3)$ numbers. Then, the generation of a random word of length $n$ can be performed in linear time. But again, the numbers $c_{i,j,h}$ are exponential in $n$, so that the actual time-space complexity of the precomputation stage may grow to $O(n^4)$.

The above procedure is easily adapted to generate culminating walks ending at a prescribed height $k$. The number $c^{(k)}_{i,j}$ of $i$-step extensions of a prefix ending at height $j$ is given by

$$\begin{aligned} c^{(k)}_{i,j} &= \mathbb{1}_{j+a<k}\, c^{(k)}_{i-1,j+a} + \mathbb{1}_{j>b}\, c^{(k)}_{i-1,j-b} \quad \text{for } i > 1, \\ c^{(k)}_{1,j} &= \mathbb{1}_{j+a=k}. \end{aligned}$$

Now $j$ is bounded by $k$, so that we only have to compute a table of $O(kn)$ numbers, in $O(kn)$ arithmetic operations. The actual time-space complexity is likely to grow to $O(kn^2)$ due to the handling of large numbers.

However, whether the height of the walk is fixed or not, one should be able to limit the computational overhead due to the size of these numbers to $O(n^\varepsilon)$, using a floating-point technique adapted from [16].

## 6.2 Rejection methods

We presented in Section 5.3 an example of the *anticipated* rejection approach. The more general *rejection principle* has been applied successfully to various problems [17, 6, 20]. The principle of a rejection algorithm for words in $\mathcal{W}$ is to draw objects uniformly in a superset $\mathcal{V} \supset \mathcal{W}$ until an object of $\mathcal{W}$ is found. The average-case complexity of a such a technique is then $\zeta(n)v_n/w_n$, where $\zeta(n)$ is the cost for the generation of a word of size $n$ in $\mathcal{V}$, and $w_n$ and $v_n$ respectively denote the number of words of length $n$ in $\mathcal{W}$ and $\mathcal{V}$.

The aim is to find a superset $\mathcal{V}$ satisfying the following (sometimes conflicting) requirements:

– the words of $\mathcal{V}$ can be generated quickly, so that $\zeta(n)$ is small,

– the set $\mathcal{V}$ is not too large, so that the ratio $v_n/w_n$ is small.

Moreover, testing whether a word of $\mathcal{V}$ actually belongs to $\mathcal{W}$ should be doable in linear time. This is obviously the case when $\mathcal{W} = \mathcal{C}^{a,b}$.

We investigate below two possibilities for the superset $\mathcal{V}$, while fixing $\mathcal{W} = \mathcal{C}^{a,b}$.

### 6.2.1 Drawing from positive walks

Here, we take for $\mathcal{V}$ the set of positive walks. Their random generation has been discussed in Section 5, and we refer to the last lines of this section for our conclusions on this question.

– When $a < b$, the number $v_n$ of positive walks of length $n$ grows like $\alpha_{a,b}^n n^{-3/2}$ (up to a multiplicative constant). If $c_n^{a,b}$ grows like $\alpha_{a,b}^n n^{-3-\gamma}$ for $\gamma \geq 0$ (see Proposition 4.3), the cost will be $O(n^{\gamma+5/2+\varepsilon})$, with a preprocessing stage of $O(n^{1+\varepsilon})$. However, approximate-size sampling can be performed in time $O(n^{\gamma+5/2})$, with no preprocessing stage. It suffices to reject among the set of positive walks generated by a Boltzmann algorithm.

– If $a = b$, then $v_n$ grows like $2^n n^{-1/2}$, while $c_n \sim 2^n/n$ (Proposition 4.1). Hence the cost here is $O(n^{3/2})$.

– Finally, for $a > b$, the number of culminating walks grows like $2^n$ (Proposition 4.2). This shows that the algorithm is linear.

**Remark.** For $a > b$, culminating walks are so numerous that we can even perform the rejection in the set of *general* $(a, b)$-walks, and still obtain a linear complexity, as discussed in the introduction. However, it seems natural to perform an anticipated rejection, rejecting walks as soon as they stop being positive: but this amounts to performing rejection in the set of positive walks, obtained themselves via an anticipated rejection from general walks.

### 6.2.2 Drawing from hybrid walks

We begin with a simple, yet crucial, observation:

> *Let* $\overleftarrow{w}$ *denote the mirror image of the word* $w$. *Then* if $w \in \mathcal{C}^{a,b}$, so is $\overleftarrow{w}$.

Graphically, taking the mirror image amounts to a central symmetry on walks. This remark implies that, on average, the mid-point of a culminating walk lies at a height which is half the final height. This suggests another possible superset of $\mathcal{C}^{a,b}$ from which we may draw, namely the language $\mathcal{H}^{a,b}$ of *hybrid walks*, defined by

$$\mathcal{H} \equiv \mathcal{H}^{a,b} := \bigcup_{n \geq 0} \mathcal{P}_{\lfloor n/2 \rfloor} \overleftarrow{\mathcal{P}_{\lceil n/2 \rceil}},$$

where $\mathcal{P}$ is the language of positive walks, and $\overleftarrow{\mathcal{P}}$ the language of mirror images of positive walks. As already observed in Section 4, $\mathcal{C}^{a,b} \subset \mathcal{H}^{a,b}$.

The intuition behind the choice of the superset $\mathcal{H}^{a,b}$ is that a path that violates the positivity (resp. final record) condition is likely to do so at its beginning (resp. ending). Thus, ensuring positivity on the first half of the walk, and the final record condition on the second half, should yield a lower rejection probability than ensuring positivity everywhere, as we did when drawing from positive walks.

How can one generate hybrid walks uniformly at random? As a hybrid walk of length $n$ is the (non-ambiguous) concatenation of a positive walk of size $\lfloor n/2 \rfloor$ and of the mirror image of another positive walk, of size $\lceil n/2 \rceil$, it is sufficient to draw positive walks uniformly at random. The cost of the generation of a hybrid walk of length $n$ will be twice the cost of the generation of a positive walk of length (approximately) $n/2$. We refer again to the end of Section 5 for our conclusions on this cost. We do not use below the Boltzmann sampling for positive walks, since gluing two positive walks of approximate size $n/2$ does not give the same probability to all hybrid walks of a given size.

Let us now discuss the efficiency of the rejection approach based on the language $\mathcal{H}$.

- When $a < b$, we have $|\mathcal{H}_n| = \Theta(\alpha_{a,b}^n / n^3)$, while $m_n = \Theta(\alpha_{a,b}^n / n^{3/2})$, so that we gain an order $O(n^{3/2})$ in complexity (comparing with the rejection of positive walks). This leads to a cost $O(n^{\gamma+1+\varepsilon})$ if $c_n$ scales like $\alpha_{a,b}^n n^{-3-\gamma}$, with a $O(n^{1+\varepsilon})$ precomputation.
- When $a = b = 1$, $|\mathcal{H}_n| = \Theta(2^n/n)$, while $m_n = \Theta(2^n/\sqrt{n})$, so that the gain is of order $\sqrt{n}$. Consequently, the complexity of the rejection algorithm based on $\mathcal{H}$ is *linear*. No precomputation nor storage is required.
- For $a > b$, we have $|\mathcal{H}_n| = \Theta(2^n)$, and similarly $m_n = \Theta(2^n)$. So the complexity gain (compared with the approach that generates positive walks) can only be $\Theta(1)$. The algorithm is still linear.

# 7 Conclusion and perspectives

We have studied culminating paths, from the point of view of formal languages, enumerative combinatorics and random generation. Our best results in terms of random generation are summarized in Table 2.

An important question that is left open is to determine the asymptotic growth of the number of culminating walks when the drift is negative ($a < b$). One possible approach would be to exploit the closed form expression of Proposition 3.3, in the spirit of Proposition 4.1 and [5]. The result might

| Paths | Method | ♯ Attempts | Precomp. | Cost |
|---|---|---|---|---|
| $\mathcal{P}^{a,b}$ $a < b$ | Recursive method, Section 5.2: standard implementation or floating-point implementation. Approximate-size Boltzmann, Section 5.4 | $O(1)$ | $O(n^2)$ $O(n^{1+\varepsilon})$ $\emptyset$ | $O(n \log n)$ $O(n^{1+\varepsilon})$ $O(n)$ |
| $a \geq b$ | Anticipated rejection, Section 5.3 | $O(\sqrt{n})$ $(a = b)$ $O(1)$ $(a > b)$ | $\emptyset$ | $O(n)$ |
| $\mathcal{C}^{a,b \Rightarrow k}$ | Recursive method, Section 6.1 | | $O(kn^{1+\varepsilon})$ | $O(n)$ |
| $\mathcal{C}^{a,b}$ $a < b$ | Recursive method, [30] and Section 6.1 or rejection from hybrid walks, Section 6.2.2 | $O(n^\gamma)$ | $O(n^{3+\varepsilon})$ $O(n^{1+\varepsilon})$ | $O(n)$ $O(n^{1+\gamma+\varepsilon})$ |
| $a = b$ | Rejection from hybrid walks, Section 6.2.2 | $O(1)$ | $\emptyset$ | $O(n)$ |
| $a > b$ | Rejection from positive walks or hybrid walks, Sections 6.2.1 and 6.2.2 | $O(1)$ | $\emptyset$ | $O(n)$ |

**Tab. 2:** The complexity of random generation of positive and culminating paths. The cost is that of one random drawing, once the precomputations have been performed. It is assumed that $c_n \sim \alpha_{a,b}^n n^{-3-\gamma}$ if $a < b$.

have interesting consequences regarding the random generation of culminating walks. In particular, if $c_n^{a,b} = \Theta((m_{n/2}^{a,b})^2\, n^{-\gamma}) = \Theta(\alpha_{a,b}^n n^{-3-\gamma})$, with $\gamma < 2$, the generation algorithm based on hybrid walks would be faster than the recursive algorithm, at least for generating few paths. However, our numerical data suggest that the ratio $c_n^{a,b}/(m_{n/2}^{a,b})^2$ decreases at least as fast as $n^{-2}$.

It would also be interesting to study how the height is distributed on random culminating walks of length $n$. Such a study may provide better algorithms for random generation, especially in the $a < b$ case, where the height is expected to be small. How does the average height scale with $n$? Is there a limiting distribution for some normalized height? This is related to a more ambitious question: is there a limiting process for culminating walks, in the same way discrete excursions converge to the Brownian excursion [29], or discrete meanders to the Brownian meander [28]? In the case $a = b = 1$, a candidate for the limit process could be the meander conditioned (with care) to reach its maximum at time 1. Note that the joint law of the maximum and final position of a meander is known [21], and related to the law of the maximum and minimum of a Brownian bridge, both in the continuous and discrete cases [13]. The case where the maximum coincides with the final position (an event of zero probability in the continuous case) is closely related to our culminating walks.

Future extensions of the present work may also include the study of culminating walks with more than two types of steps, in order to model different kinds of matches and mismatches, and thus capture the whole scoring scheme of the FLASH algorithm. For instance, it is usually considered less drastic to replace a purine base by another purine base (A↔G) rather than a pyrimidine one in DNA. It is thus natural to penalize differently different mismatches. This could be modelled by introducing down steps of different heights.

Lastly, a natural, biologically relevant perspective would be to address the *non-uniform* generation of culminating paths. Indeed, the matches and mismatches may not be uniform over a biological sequence, and be subject to local correlations. This is classically modelled by a Markov chain (further conditioned to yield culminating paths). Our algorithms could in principle be adapted to this more general context, but their analysis would need to be carefully worked out. In particular, the drift of random walks would depend on the chain and differ in general from $a - b$. We naturally expect the efficiency of our algorithms to depend of the model, culminating walks with positive drift being much easier to generate than those with a negative drift.

## Acknowledgements

We are grateful to Jean-François Marckert for pointing out the reference [29], which shortened significantly the proof of Proposition 4.3. We also thank Jean Bertoin, Philippe Chassaing, Jean-François Le Gall and Svante Janson for discussions on the possible limiting process of culminating paths.

## References

[1] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman. Basic local alignment search tool. *J. Molecular Biology*, 215(3):403–410, 1990.

[2] A. Ayyer and D. Zeilberger. Two dimensional directed lattice walks with boundaries. ArXiv cond-mat/0701674.

[3] C. Banderier. *Combinatoire analytique des chemins et des cartes*. PhD thesis, Université Paris 6, 2001.

[4] C. Banderier, M. Bousquet-Mélou, A. Denise, P. Flajolet, D. Gardy, and D. Gouyou-Beauchamps. Generating functions for generating trees. *Discrete Math.*, 246(1-3):29–55, 2002.

[5] C. Banderier and P. Flajolet. Basic analytic combinatorics of directed lattice paths. *Theoret. Comput. Sci.*, 281(1-2):37–80, 2002.

[6] E. Barcucci, R. Pinzani, and R. Sprugnoli. The random generation of underdiagonal walks. In P. Leroux and C. Reutenauer, editors, *Proceedings of 4th Conference on Formal Power Series and Algebraic Combinatorics (FPSAC'92)*. Université du Québec à Montréal, 1992.

[7] E. Barcucci, R. Pinzani, and R. Sprugnoli. The random generation of directed animals. *Theoret. Comput. Sci.*, 127(2):333–350, 1994.

[8] M. Bousquet-Mélou and M. Petkovšek. Linear recurrences with constant coefficients: the multivariate case. *Discrete Math.*, 225(1-3):51–75, 2000.

[9] M. Bousquet-Mélou. Rational and algebraic series in combinatorial enumeration. In *Proceedings of the International Congress of Mathematicians*, pages 789–826, Madrid, 2006. European Mathematical Society Publishing House.

[10] M. Bousquet-Mélou. Discrete excursions. *Séminaire Lotharingien de Combinatoire*, 57, 2008. (electronic) Article B57d, 23 pp. ArXiv math.CO/0701171.

[11] A. Califano and I. Rigoutsos. Flash: A fast look-up algorithm for string homology. In *Proceedings of the 1st International Conference on Intelligent Systems for Molecular Biology*, pages 56–64. AAAI Press, 1993.

[12] F. Carlson. Über Potenzreihen mit ganzzahligen Koeffizienten. *Math. Z.*, 9(1-2):1–13, 1921.

[13] E. Csáki and S. G. Mohanty. Excursion and meander in random walk. *Canad. J. Statist.*, 9(1):57–70, 1981.

[14] A. Denise. Génération aléatoire et uniforme de mots. *Discrete Math.*, 153:69–84, 1996.

[15] A. Denise, I. Dutour, and P. Zimmermann. CS: a MuPAD package for counting and randomly generating combinatorial structures. In *Proceedings of 10th Conference on Formal Power Series and Algebraic Combinatorics (FPSAC'98)*, pages 195–204, 1998. Also published in MathPAD 8 (1) 1998.

[16] A. Denise and P. Zimmermann. Uniform random generation of decomposable structures using floating-point arithmetic. *Theoret. Comput. Sci.*, 218:233–248, 1999.

[17] L. Devroye. *Non-Uniform Random Variate Generation*. Springer-Verlag, New York, 1986.

[18] P. Di Francesco, E. Guitter, and C. Kristjansen. Integrable 2D Lorentzian gravity and random walks. *Nuclear Phys. B*, 567(3):515–553, 2000.

[19] P. Duchon. On the enumeration and generation of generalized Dyck words. *Discrete Math.*, 225(1-3):121–135, 2000.

[20] P. Duchon, P. Flajolet, G. Louchard, and G. Schaeffer. Boltzmann samplers for the random generation of combinatorial structures. *Combin. Probab. Comput.*, 13(4-5):577–625, 2004.

[21] R. T. Durrett and D. L. Iglehart. Functionals of Brownian meander and Brownian excursion. *Ann. Probability*, 5(1):130–135, 1977.

[22] P. Flajolet, X. Gourdon, and P. Dumas. Mellin transforms and asymptotics: harmonic sums. *Theoret. Comput. Sci.*, 144(1-2):3–58, 1995.

[23] P. Flajolet and A. Odlyzko. Singularity analysis of generating functions. *SIAM J. Discrete Math.*, 3(2):216–240, 1990.

[24] P. Flajolet, P. Zimmerman, and B. Van Cutsem. A calculus for the random generation of labelled combinatorial structures. *Theoret. Comput. Sci.*, 132(1-2):1–35, 1994.

[25] D. Foata. A noncommutative version of the matrix inversion formula. *Adv. in Math.*, 31(3):330–349, 1979.

[26] I. M. Gessel. A factorization for formal Laurent series and lattice path enumeration. *J. Combin. Theory Ser. A*, 28(3):321–337, 1980.

[27] J. E. Hopcroft and J. D. Ullman. *Formal languages and their relation to automata*. Addison-Wesley, 1969.

[28] D. L. Iglehart. Functional central limit theorems for random walks conditioned to stay positive. *Ann. Probability*, 2:608–619, 1974.

[29] W. D. Kaigh. An invariance principle for random walk conditioned by a late return to zero. *Ann. Probability*, 4(1):115–121, 1976.

[30] G. Kucherov, L. Noe, and Y. Ponty. Estimating seed sensibility on homogenous alignments. In IEEE, editor, *Proceedings of Fourth IEEE Symposium on Bioinformatics and Bioengineering (BIBE04)*, page 387, 2004.

[31] J. Labelle. Langages de Dyck généralisés. *Ann. Sci. Math. Québec*, 17(1):53–64, 1993.

[32] J. Labelle and Y. N. Yeh. Generalized Dyck paths. *Discrete Math.*, 82(1):1–6, 1990.

[33] G. Louchard. Asymptotic properties of some underdiagonal walks generation algorithms. *Theoret. Comput. Sci.*, 218(2):249–262, 1999.

[34] W. R. Pearson and D. J. Lipman. Improved tools for biological sequence comparison. *Proceedings of the National Academy of Sciences of the USA*, 85:2444–2448, 1988.

[35] Y. Ponty, M. Termier, and A. Denise. GenRGenS: Software for generating random genomic sequences and structures. *Bioinformatics*, 22(12):1534–1535, 2006.

[36] R. P. Stanley. *Enumerative combinatorics. Vol. 1*, volume 49 of *Cambridge Studies in Advanced Mathematics*. Cambridge University Press, Cambridge, 1997.

[37] R. P. Stanley. *Enumerative combinatorics. Vol. 2*, volume 62 of *Cambridge Studies in Advanced Mathematics*. Cambridge University Press, Cambridge, 1999.

[38] G. X. Viennot. Heaps of pieces. I. Basic definitions and combinatorial lemmas. In *Combinatoire énumérative (Montréal, Québec, 1985)*, volume 1234 of *Lecture Notes in Math.*, pages 321–350. Springer, Berlin, 1986.

[39] H. S. Wilf. A unified setting for sequencing, ranking, and selection algorithms for combinatorial objects. *Advances in Math.*, 24(3):281–291, 1977.