DIPLOMARBEIT

The Vectorial Kernel Method and Applications

ausgeführt am

Institut für
Analysis und Scientific Computing
TU Wien

unter der Anleitung von

Ao. Univ. Prof. Dr. Bernhard Gittenberger

durch

Katharina Schindegger
Matrikelnummer: 11918467
Schmalzgasse 5
6835 Zwischenwasser

Wien, am 09.05.2026

Verfasserin

Betreuer

Kurzfassung

Die Aufzählung von Gitterpfaden ist ein klassisches Gebiet der Kombinatorik mit engen Verbindungen zur Wahrscheinlichkeitstheorie und zur Informatik. Objekte wie Dyck-Pfade, Mäander und Exkursionen modellieren zufällige Irrfahrten mit Nichtnegativitätsbedingungen und treten in der Analyse von Algorithmen und der Warteschlangentheorie auf Eine zentrale Frage in diesem Bereich ist, wie viele Gitterpfade einer gegebenen Länge ein vorgeschriebenes lokales Muster aus aufeinanderfolgenden Schritten vermeiden oder wie oft ein solches Muster in einem typischen Pfad vorkommt.

Diese Arbeit behandelt die vectorial kernel method, ein leistungsfähiges algebraisches Werkzeug zur systematischen Lösung solcher Aufzählungsprobleme, das von Asinowski, Bacher, Banderier und Gittenberger entwickelt wurde. Die Methode kodiert die Struktur erlaubter Pfade mithilfe eines endlichen Automaten, dessen Zustände den Fortschritt bei der Erkennung des verbotenen Musters verfolgen. Das resultierende Gleichungssystem wird durch die Übergangsmatrix des Automaten und geeignete algebraische Techniken simultan gelöst, was explizite erzeugende Funktionen für alle klassischen Pfadfamilien liefert.

Die vorliegende Arbeit gibt eine vollständige und detaillierte Darstellung dieser Methode, einschließlich aller Beweise, die im Originalartikel nur skizziert werden.

Darüber hinaus wird die Methode auf konkrete Beispiele angewendet, darunter eine eingehende Untersuchung der Folge A094507, welche Dyck-Pfade nach der Anzahl der Vorkommen eines bestimmten Musters klassifiziert. Für diese Folge wird eine neue bivariate erzeugende Funktion hergeleitet, aus der sich mehrere bekannte Zahlenfolgen als Spezialfälle ergeben.

Die Ergebnisse sind relevant, weil die vectorial kernel method einen einheitlichen Rahmen für eine breite Klasse kombinatorischer Aufzählungsprobleme bereitstellt und gleichzeitig präzise asymptotische Aussagen über das Wachstumsverhalten der gezählten Objekte ermöglicht.

Abstract

The enumeration of lattice paths is a classical area of combinatorics with close connections to probability theory and computer science. Objects such as Dyck paths, meanders, and excursions model random walks subject to non-negativity constraints and arise naturally in the analysis of algorithms and queueing theory. A central question in this field is how many lattice paths of a given length avoid a prescribed local pattern of consecutive steps, or how often such a pattern occurs in a typical path. This thesis studies the vectorial kernel method, a powerful algebraic tool for the systematic solution of such enumeration problems, developed by Asinowski, Bacher, Banderier, and Gittenberger. The method encodes the structure of admissible paths using a finite automaton whose states track progress in recognising the forbidden pattern. The resulting system of functional equations is solved simultaneously via the transition matrix of the automaton and suitable algebraic techniques, yielding explicit generating functions for all classical path families. The present work provides a complete and detailed exposition of the method, including all proofs that are only sketched in the original paper. Beyond this, the method is applied to concrete examples, among them a thorough investigation of the sequence A094507, which classifies Dyck paths by the number of occurrences of a given pattern. For this sequence a new bivariate generating function is derived, from which several well-known integer sequences emerge as special cases. The results are significant because the vectorial kernel method provides a unified framework for a broad class of combinatorial enumeration problems while simultaneously enabling precise asymptotic statements about the growth behaviour of the objects being counted.

Acknowledgements

I would like to express my sincere gratitude to my supervisor Prof. Gittenberger for his generous guidance, his careful feedback, and the time he dedicated to this thesis.

I am deeply grateful to my family for their unwavering support throughout my studies. In particular, I thank my parents for always being there for me, and my sisters for their constant encouragement. I also thank my grandfather for proofreading this thesis with such care and patience.

I thank my fellow student Dina for accompanying me through all the challenges of this degree, her companionship made the hard times much easier to bear.

Finally, I gratefully acknowledge fem*MA for selecting me as an award recipient of their programme supporting women in master's studies.

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Diplomarbeit selbstständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt bzw. die wörtlich oder sinngemäß entnommenen Stellen als solche kenntlich gemacht habe.

Wien, am Datum

Name des Autors

Contents

1 Introduction

1.1 Motivation and Context

The enumeration of lattice paths is a classical topic in combinatorics, with deep connections to probability theory. Objects such as Dyck paths, Motzkin paths, meanders, and excursions arise naturally in a wide range of contexts: they model random walks constrained to stay non-negative, encode the execution traces of pushdown automata, and appear as fundamental structures in the theory of formal languages [FS09]. Beyond their intrinsic combinatorial interest, these models serve as a testing ground for powerful analytic techniques, and their generating functions exhibit a striking universality that connects discrete combinatorics to complex analysis.

A key theme in this area is the imposition of additional constraints on lattice paths. Rather than counting all paths of a given length, one may require that the path *avoids* a prescribed pattern of steps, or count the number of times a pattern *occurs*. The challenge is to develop a unified framework that handles arbitrary patterns, arbitrary step sets, and all classical path families simultaneously.

This thesis is based on and extends the work of Asinowski, Bacher, Banderier, and Gittenberger [ABBG20], who introduced the *vectorial kernel method* as a powerful and unified framework for the enumeration of lattice paths with forbidden or counted patterns. The present work provides a more thorough exposition of the underlying concepts, explicit descriptions of the calculations, and detailed presentations of the proofs. Where the original paper states results concisely, we provide complete proofs and resolve several gaps in the original source and we illustrate this through concrete examples.

1.2 The Kernel Method and Its Vectorial Extension

The kernel method has long been a fundamental tool in the study of functional equations arising from lattice path enumeration. Its origins can be traced back to classical work including [Knu68], and it was later systematically developed in the context of analytic combinatorics by Flajolet and Sedgewick [FS09].

In its classical form, the kernel method handles a single functional equation in one catalytic variable, and has been successfully applied to directed lattice paths by Banderier and Flajolet [BF02]. However, other problems, in particular those involving pattern avoidance or pattern counting, lead to *systems* of functional equations rather than a single one. Each state of a pattern-tracking automaton contributes an equation, and the equations are coupled through the transition structure of the automaton. The classical kernel method does not extend directly to this setting, which motivates the *vectorial kernel method* developed in [ABBG20].

The central idea of the vectorial kernel method is to encode the combinatorial structure of constrained lattice paths via a finite automaton whose states represent partial matches of the forbidden or counted pattern. This leads to a system of functional equations involving a vector of generating functions, one per automaton state. Combining this system with algebraic properties of the transition matrix yields a generalized kernel equation whose analysis produces explicit generating functions for all classical path families simultaneously. A crucial role is played by the *autocorrelation polynomial* of the forbidden pattern, introduced by Guibas and Odlyzko [GO81] in the context of rational languages, which encodes the self-overlap structure of the pattern and determines the shape of the kernel.

The analytic component of this work rests on the principles of analytic combinatorics as developed in [FS09]. Generating functions are treated as complex analytic objects, and the asymptotic behaviour of their coefficients is determined by the nature of their dominant singularities. The generating functions arising from the vectorial kernel method are *algebraic*. Their dominant singularities are generically of square-root type, which leads to a universal asymptotic regime of the form

$$f_n \sim C \cdot n^{-3/2} \cdot \rho^{-n},$$

for excursions and bridges, where ρ is the radius of convergence and C is an explicit constant determined by the kernel and the autocorrelation polynomial. Meanders exhibit different behaviour depending on the drift of the underlying walk.

1.3 Outline of the Thesis

The thesis is organized as follows.

Chapter 2: Mathematical background. We introduce the analytic and combinatorial tools required throughout the thesis. This includes the theory of generating functions and the symbolic method, the classification of lattice paths into walks, meanders, bridges, and excursions, and the relevant results from complex analysis and singularity analysis that underpin the asymptotic arguments.

Chapter 3: Patterns, overlaps, and automata. We formalize the notion of a pattern in a lattice path and introduce the autocorrelation polynomial, following Guibas and Odlyzko [GO81]. We construct the pattern-tracking automaton and derive the system of functional equations to which the kernel method will be applied.

Chapter 4: The vectorial kernel method. We develop the vectorial kernel method in full generality. This includes the construction of the transition matrix, the derivation of the generalized kernel equation, the analysis of its small and large roots via Newton polygon theory, and the explicit formulas for the generating functions of excursions, bridges, meanders, and walks with a forbidden pattern.

Chapter 5: Asymptotic analysis. We derive the asymptotic behaviour of the counting sequences for each path family, establishing the universal $n^{-3/2}\rho^{-n}$ regime for excursions and bridges, and the three drift-dependent regimes for meanders.

Chapter 6: Applications and examples. We apply the vectorial kernel method to specific patterns, including peaks, valleys, and the pattern $UDUD$ in Dyck paths. For each example we derive explicit generating functions, verify the results against known sequences in the Online Encyclopedia of Integer Sequences [The26], and determine the asymptotic behaviour of the associated counting sequences. In particular, for the triangle A094507 we derive a trivariate generating function $M(t, u, v)$ and an explicit closed form for $E(t, v)$ via the vectorial kernel method, which to the best of the author's knowledge are new. The asymptotic constants for the three illustrative drift examples are also believed to be new, though no exhaustive literature search has been conducted for these cases.

Chapter 7: Conclusion. We summarise the contributions of the thesis and discuss directions for further research.

Overall, this work demonstrates how the vectorial kernel method provides a flexible and unified framework for the enumeration of constrained lattice paths, bridging combinatorial constructions, algebraic techniques, and analytic methods in a way that is both systematic and computationally explicit.

2 Preliminaries Miscellaneous

In this chapter, we introduce the main tools that will be used throughout this work. These include combinatorial constructions, generating functions, and algebraic techniques such as the kernel method and adjugate matrices.

We also provide a brief overview of analytic tools that will later allow us to extract asymptotic information from generating functions.

2.1 The Kernel Method

The kernel method is a classical technique in enumerative combinatorics, particularly well suited for the study of lattice paths and constrained walks. Both of which we will do in this work. It apparently can be traced back to early combinatorial arguments, appearing in the exercise sections of the first edition of Knuth's *The Art of Computer Programming* [Knu68], according to [BF02] and [FS09].

Over time, the method has evolved into a systematic tool, with significant developments by several authors. For example, the work of Banderier and Flajolet [BF02, Section 2] provides a comprehensive and analytic treatment of directed lattice paths, placing the kernel method into a broader framework that connects combinatorics, algebra, and complex analysis.

At its core, the kernel method applies to functional equations involving generating functions. Typically, one encounters an equation of the form

$$K(z,u)\,F(z,u) = A(z,u),$$

where $F(z,u)$ and $A(z,u)$ are unknown and $K(z,u)$ is a polynomial or Laurent polynomial in u, called the *kernel*. The key idea is to exploit the structure of this equation by choosing a function $u = u(z)$ such that the kernel and thus the entire left hand side vanishes, that is,

$$K(z,u(z)) = 0.$$

Under suitable analyticity conditions, this substitution eliminates the unknown function $F(z,u)$ from the equation, leading to a new equation that determines the remaining unknown quantities, see, for instance, [FS09, Section VII.4].

The strength of the kernel method lies in its ability to reduce a complicated functional equation to algebraic relations by a smart choice of specialization.

In the work of [ABBG20], the authors build upon this classical method and extend it to a vectorial setting. Systems of functional equations arise from walks encoded by finite automata. This framework allows one to treat pattern-constrained walks by working with vector-valued generating functions and matrix kernels. This generalization allows us to handle pattern-avoiding walks encoded by finite automata, thereby significantly enlarging the scope of problems that can be treated within the kernel method framework.

The development and application of this extended kernel method form the main focus of the present work.

2.2 The Adjugate Matrix

Later on we will be working with matrix equations. A useful method to manipulate the matrix is (2.1), which we will explain in this section.

Definition 2.1 (Adjugate Matrix)**.** *Let $A = (a_{ij})$ be a square matrix of order n with entries in $\mathbb{R}$. For each entry a_{ij}, define its* cofactor *A_{ij} as*

$$A_{ij} = (-1)^{i+j} \det(M_{ij}),$$

where M_{ij} is the $(n-1) \times (n-1)$ minor of A obtained by deleting row i and column j.

The adjugate matrix*, which is also called the classical adjoint, of A, denoted* $\operatorname{adj}(A)$, *is the transpose of the cofactor matrix*

$$\operatorname{adj}(A) = (A_{ji})_{i,j=1}^{n}.$$

Thus, the (i, j)-entry of $\operatorname{adj}(A)$ *is the cofactor of the entry a_{ji} of A. This definition can be found for instance at [Ste98].*

The adjugate matrix is closely related to the determinant and allows one to express the inverse of a nonsingular matrix as

$$A^{-1} = \frac{1}{\det(A)} \operatorname{adj}(A), \tag{2.1}$$

whenever $\det(A) \neq 0$, this can also be found in [Ste98]. This property follows from the following theorem.

Theorem 2.2 (Fundamental Property of the Adjugate Matrix)**.** *Let R be a commutative ring with unity, and let $A \in R^{n \times n}$ be a square matrix of order n. Then,*

$$A \cdot \operatorname{adj}(A) = \operatorname{adj}(A) \cdot A = \det(A) \cdot I_n,$$

where $\det(A)$ *is the determinant of A, and I_n denotes the identity matrix of order n.*

Proof. Let $A = (a_{ij})$, and let A_{ij} denote the cofactor of a_{ij} in A.

We first show the right multiplication

$$A \cdot \operatorname{adj}(A) = \det(A) \cdot I_n.$$

The (i, j)-th entry of this product is

$$(A \cdot \operatorname{adj}(A))_{ij} = \sum_{k=1}^{n} a_{ik} A_{jk}.$$

If $i = j$, the Laplace expansion of $\det(A)$ along row i gives

$$\det(A) = \sum_{k=1}^{n} a_{ik} A_{ik},$$

so $(A \cdot \operatorname{adj}(A))_{ii} = \det(A)$.

If $i \neq j$, let A' be the matrix obtained from A by replacing row j with row i. The matrix A' has two identical rows, so $\det(A') = 0$. Expanding $\det(A')$ along row j yields

$$0 = \sum_{k=1}^{n} a_{ik} A_{jk}.$$

Hence $(A \cdot \operatorname{adj}(A))_{ij} = 0$ for $i \neq j$, and the result follows.

Now there is only the multiplication from the left missing. The proof of $\operatorname{adj}(A) \cdot A = \det(A) \cdot I_n$ is analogous, using Laplace expansion along columns instead of rows. The argument proceeds in the same way by considering matrices with two identical columns, whose determinant is zero. □

2.3 Generating Functions

This section follows the notation and general framework of [FS09]. Throughout this work we study combinatorial objects and their enumeration. In combinatorics, one is interested in determining the number of objects of a given size within a specified class. To formalize this idea, we begin with the notion of a combinatorial class.

Definition 2.3. *A* combinatorial class $\mathcal{S}$ *is a countable set together with a size function*

$$\omega : \mathcal{S} \to \mathbb{N},$$

such that for every $n \in \mathbb{N}$*,*

$$|\,\omega^{-1}(n)\,| < \infty.$$

In other words, each object in the class is assigned a non-negative integer, called its *size*, and for every fixed size n there are only finitely many objects of that size. This finiteness condition ensures that enumeration is well-defined. This notation is taken from [FS09].

Definition 2.4. *The* counting sequence *associated with a combinatorial class* $\mathcal{S}$ *is the sequence* $(A_n)_{n\geq 0}$ *defined by*

$$A_n := |\,\omega^{-1}(n)\,|,$$

that is, A_n *is the number of objects in* $\mathcal{S}$ *of size* n*.*

By construction, the numbers A_n are non-negative integers. The central goal of enumerative combinatorics is to determine these numbers, either exactly or asymptotically. More about counting sequences can be found in [FS09].

A fundamental tool for encoding and studying counting sequences is the notion of a generating function. Generating functions transform combinatorial counting problems into algebraic or analytic problems, which can often be treated more systematically.

Definition 2.5. *Let* $(A_n)_{n\geq 0}$ *be a sequence of non-negative integers. The* ordinary generating function *(OGF) of this sequence is the formal power series*

$$A(z) := \sum_{n\geq 0} A_n z^n.$$

Remark 2.6. *Besides ordinary generating functions, one also encounters* exponential generating functions *(EGFs), defined by*

$$A^{\exp}(z) := \sum_{n\geq 0} A_n \frac{z^n}{n!}.$$

EGFs are particularly useful for labeled combinatorial structures. Since the objects studied in this work are unlabeled lattice paths, we will restrict our attention to ordinary generating functions. For more information about exponential generating functions see [FS09].

The generating function $A(z)$ is interpreted as a *formal power series.* At this stage, we do not consider convergence issues and thus, the variable z is treated as an indeterminate. Analytic aspects such as radius of convergence and singular behaviour become relevant later, when asymptotic properties of the counting sequence are studied.

The generating function encodes the entire counting sequence into a single formal object. To recuperate the original counting series from the generating function we use coefficient extraction.

Definition 2.7. *For a formal power series*

$$A(z) = \sum_{n\geq 0} A_n z^n,$$

the notation

$$[z^n]A(z)$$

denotes the coefficient of z^n *in* $A(z)$*. Thus,*

$$[z^n]A(z) = A_n.$$

The power of generating functions stems from the fact that combinatorial constructions translate directly into algebraic operations: disjoint union corresponds to addition, Cartesian product to multiplication, and sequences to geometric series. This correspondence is the foundation of the *symbolic method*, which will be illustrated in Section 2.4 in the enumeration of binary trees.

2.4 Binary Trees

We briefly introduce binary trees, which serve as a first illustration of the symbolic method and motivate the Catalan numbers that appear throughout this work.

A *tree* consists of a root and nodes connected in a hierarchical, acyclic manner, where every node except the root has exactly one parent. Nodes with no children are called *leaves* or external nodes, all other nodes are *internal nodes*. A *plane binary tree* is a tree in which every internal node has exactly two ordered children. We define the *size* of a binary tree as its number of internal nodes. This definition is introduced in [FS09].

The class $\mathcal{B}$ of all binary trees admits the recursive decomposition

$$\mathcal{B} \;=\; \{\varepsilon\} \;+\; \mathcal{Z} \times \mathcal{B} \times \mathcal{B},$$

where ε denotes the empty tree, a single leaf, and $\mathcal{Z}$ marks one internal node. An explanation for this equation can be given by the following idea. The set of all binary trees consists of trees that are a single leaf, and of trees where the root has two ordered children, which themselves are binary trees again. This relationship warrants this equation. Translating this relation into generating functions gives

$$B(z) = 1 + z\,B(z)^2.$$

This quadratic equation has two solutions

$$B(z) = \frac{1 \pm \sqrt{1-4z}}{2z}.$$

To select the correct branch, we remind ourselves that we are looking for an analytic solution. Thus we require $B(z)$ to be analytic at $z = 0$. The positive branch diverges as $z \to 0$. Thus we know that the other branch must be the correct solution. We verify this by applying the rule of l'Hôpital

$$\lim_{z\to 0^+} \frac{1-\sqrt{1-4z}}{2z} \;=\; \lim_{z\to 0^+} \frac{2}{\sqrt{1-4z}} \cdot \frac{1}{1} = 1,$$

so the correct generating function is the negative branch

$$B(z) = \frac{1 - \sqrt{1-4z}}{2z}.$$

To extract coefficients, we expand using the generalized binomial theorem. For arbitrary $x \in \mathbb{R}$ and $k \in \mathbb{N}$, recall

$$\binom{x}{k} = \frac{x(x-1)\cdots(x-k+1)}{k!},$$

which allows us to expand $(1-4z)^{1/2}$ as a power series. This formula can be found in various textbooks including [FS09]. A standard computation then yields

$$[z^n]\,B(z) = \frac{1}{n+1}\binom{2n}{n} =: C_n,$$

the n-th *Catalan number*. The catalan numbers play a frequent role in combinatorics. In [FS09] there is a lot of additional information on this.

To complete our calculation we now just give the form of the generating function

$$B(z) = \sum_{n=0}^{\infty} C_n\, z^n.$$

The Catalan numbers will reappear in the context of Dyck paths in Section 2.5, illustrating a deep connection between these two families.

2.5 Lattice Paths

We now introduce lattice paths, which form the basic combinatorial objects studied throughout this work.

Definition 2.8. *Let $\mathcal{S} \subset \mathbb{Z}$ be a finite set of integers, called the* step set*, containing at least one positive and one negative element. A* lattice path *p of length n with step set $\mathcal{S}$ is a sequence of steps*

$$p := (s_1, s_2, \dots, s_n), \qquad s_i \in \mathcal{S}.$$

The final altitude *of the path is*

$$alt(p) := \sum_{i=1}^{n} s_i.$$

Since each step advances the path by one unit horizontally, a path of length n consists of exactly n steps.

This definition is from [ABBG20, Section 2].

Geometrically, a lattice path can be drawn as a polygonal line in the plane, starting at the origin $(0,0)$ and connecting the points (i, h_i) for $i = 0, 1, \dots, n$, where $h_0 = 0$ and $h_i = \sum_{j=1}^{i} s_j$. At each step, the horizontal coordinate increases by one and the vertical coordinate changes according to the chosen step. Figure 2.1 shows an example.

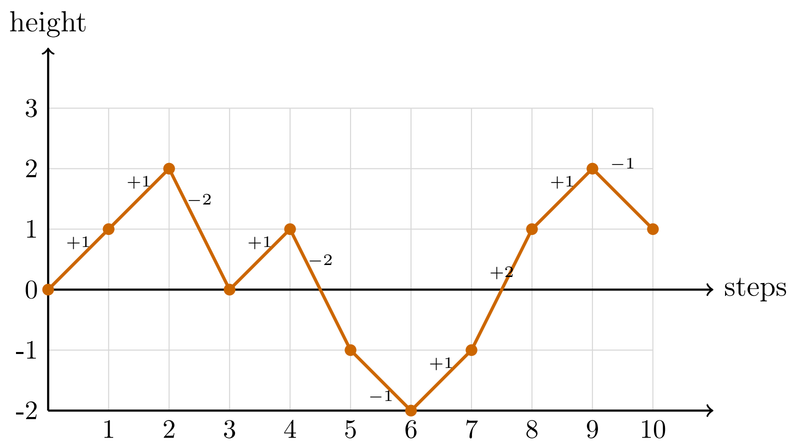


Figure 2.1: A general lattice path of length 10 with step set $\mathcal{S} = \{-2, -1, +1, +2\}$.

In the following subsections we describe two classical families of lattice paths that are constrained to remain non-negative, distinguished by their step sets and endpoint conditions.

2.5.1 Dyck Paths

Definition 2.9. *A* Dyck path *is a lattice path with step set $\mathcal{S} = \{-1, +1\}$ that starts at height 0, never goes below the x-axis, and returns to height 0 at the end. The two steps are called an* up-step $\nearrow = (1,1)$ *and a* down-step $\searrow = (1,-1)$ *respectively.*

A Dyck path of length $2n$ consists of exactly n up-steps and n down-steps. In the literature one also encounters the term *semilength* n to refer to a Dyck path of length $2n$. Since every Dyck path has even length, it is convenient to index the family by half the length. The two conventions describe the same object; the semilength is simply the length divided by two. Figure 2.2 shows a typical example.

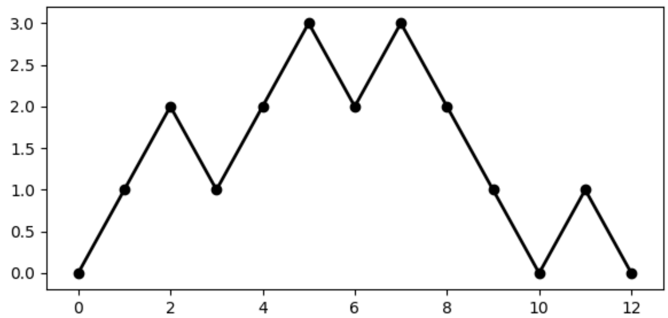


Figure 2.2: A Dyck path of semilength 6 and length 12.

Dyck paths are in bijection with rooted binary trees: an up-step corresponds to descending into a subtree, and a down-step to backtracking to the parent. This bijection explains why they are counted by the Catalan numbers, as we now derive.

The class $\mathcal{D}$ of Dyck paths satisfies the recursive decomposition

$$\mathcal{D} \;=\; \{\varepsilon\} \cup \nearrow\mathcal{D}\searrow \mathcal{D},$$

where ε denotes the empty path. A nonempty Dyck path consists of a first up-step, a possibly empty Dyck path that returns to the same height, a matching down-step, and a final Dyck path. Translating this relation into generating functions, where z marks each step, gives

$$D(z) = 1 + z^2\, D(z)^2,$$

whose unique power-series solution is

$$D(z) = \frac{1 - \sqrt{1 - 4z^2}}{2z^2}.$$

As computed in Section 2.4, the coefficient of z^{2n} is the n-th Catalan number

$$[z^{2n}]\, D(z) = C_n = \frac{1}{n+1}\binom{2n}{n},$$

confirming the bijection with binary trees mentioned above. This relation can also be found in [FS09, Chapter I].

2.5.2 Motzkin Paths

Definition 2.10. *A* Motzkin path *is a lattice path with step set* $\mathcal{S} = \{-1, 0, +1\}$ *that starts at height* 0 *and never goes below the* x*-axis. The three possible steps are called an* up-step

(+1), *a* level step (0), *and a* down-step (−1). *If additionally the path ends at height* 0, *it is called a* Motzkin excursion. *Generally, Motzkin paths can end at any height.*

Motzkin paths generalize Dyck paths by permitting horizontal movement. This means that at each position the path may stay at the same height rather than being forced to rise or fall. This additional freedom means that, unlike Dyck paths, a Motzkin path of length n need not have even length, and the final altitude can be any nonnegative integer.

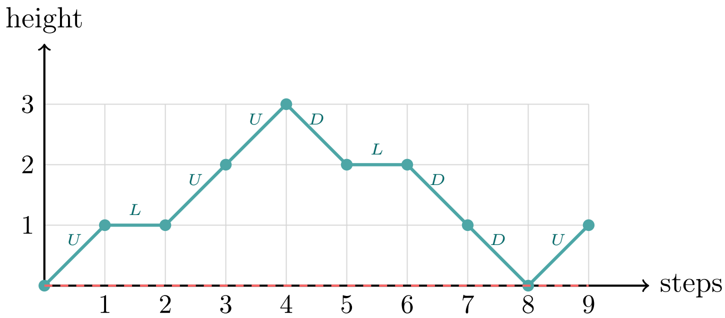


Figure 2.3: A Motzkin path of length 9 with steps $U, L, U, U, D, L, D, D, U$, ending at height 1. The dashed red line indicates the non-negativity constraint.

To count Motzkin excursions, we use a recursive decomposition analogous to the Dyck path case. A Motzkin excursion is either the empty path, a level step followed by a Motzkin excursion, or an up-step followed by a Motzkin excursion, a down-step, and another Motzkin excursion:

$$\mathcal{M} \;=\; \{\varepsilon\} \cup \rightarrow\!\mathcal{M} \cup \nearrow\!\mathcal{M} \searrow\!\mathcal{M}.$$

Translating into generating functions, where z marks each step, gives the functional equation

$$M(z) = 1 + z\,M(z) + z^2\,M(z)^2.$$

This is again a quadratic equation in $M(z)$. Solving it and selecting the branch that is analytic at $z = 0$ yields

$$M(z) = \frac{1 - z - \sqrt{1 - 2z - 3z^2}}{2z^2}.$$

The Motzkin numbers $m_n = [z^n]\,M(z)$ begin

$$1,\ 1,\ 2,\ 4,\ 9,\ 21,\ 51,\ 127,\ 323,\ \ldots$$

Like the Catalan numbers, the Motzkin numbers count a wide variety of combinatorial objects see [FS09] and [The26] for a detailed discussion.

2.6 Analytic Tools

Until now we have treated generating functions as formal power series, manipulating them purely algebraically. In this section we introduce the analytic perspective, which interprets a generating function

$$A(z) = \sum_{n\geq 0} A_n z^n$$

as a function of a complex variable $z \in \mathbb{C}$. This shift of viewpoint is essential, the asymptotic growth of the coefficients A_n is governed not by the algebraic form of $A(z)$, but by the location and nature of its singularities in the complex plane. The fundamental principle of analytic combinatorics is that *singularities of generating functions encode asymptotics of counting sequences* [FS09].

Throughout this section, U denotes an open connected subset of $\mathbb{C}$. Proofs of the classical results stated here can be found in standard references on complex analysis, such as [Jän06].

2.6.1 Power Series and Radius of Convergence

A power series centered at $z_0 \in \mathbb{C}$ is a series of the form

$$\sum_{n\geq 0} a_n (z - z_0)^n.$$

The fundamental convergence theorem characterises the region in which such a series defines an analytic function.

Theorem 2.11. *Every power series $\sum_{n\geq 0} a_n(z-z_0)^n$ has a radius of convergence*

$$R = \frac{1}{\limsup_{n\to\infty} |a_n|^{1/n}} \;\in\; [0, +\infty],$$

such that the series converges absolutely for $|z - z_0| < R$ and diverges for $|z - z_0| > R$. Inside the disk $|z - z_0| < R$, the series defines a holomorphic function [FS09, Appendix B.2], [Jän06, Chapter 1].

We use this theorem in Chapter 5.

2.6.2 Holomorphic Functions and Cauchy's Integral Formula

A function $f : U \to \mathbb{C}$ is *holomorphic* on U if it is complex-differentiable at every point of U. Holomorphic functions enjoy remarkably strong properties compared to their real counterparts, they are infinitely differentiable, they are determined by their values on any open subset, and they can always be locally represented as convergent power series.

Theorem 2.12 (Taylor expansion of holomorphic functions, [FS09, Section IV.1], [Jän06, Chapter 2])**.** *Let $f : U \to \mathbb{C}$ be holomorphic and let $z_0 \in U$. Then f is represented by a convergent power series*

$$f(z) = \sum_{n\geq 0} c_n (z - z_0)^n$$

on any disk $\overline{D}(z_0, r) \subset U$, where the coefficients are given by

$$c_n \;=\; \frac{f^{(n)}(z_0)}{n!} \;=\; \frac{1}{2\pi i} \oint_{|z-z_0|=r} \frac{f(z)}{(z - z_0)^{n+1}}\, dz.$$

We will use this theorem a lot implicitly in chapter 5 to give a different representation of the holomorphic function. The following corollary is a reiteration of Theorem 2.12. It depicts how this knowledge is used in this work.

Corollary 2.13 (Coefficient extraction formula, Section IV.1 [FS09]). *If* $A(z) = \sum_{n\geq 0} A_n z^n$ *is holomorphic in a neighbourhood of the origin, then*

$$A_n \;=\; [z^n]\, A(z) \;=\; \frac{1}{2\pi i} \oint_{|z|=r} \frac{A(z)}{z^{n+1}}\, \mathrm{d}z,$$

for any r *such that the circle of radius* r *lies entirely within the domain of holomorphicity of* A.

This formula is the analytic foundation of coefficient asymptotics. By deforming the contour of integration towards the singularities of $A(z)$, one can extract precise asymptotic information about A_n. This strategy is explained in Section 2.8.

2.6.3 Singularities and Laurent Series

Any point z_0 where a function f fails to be holomorphic is called a *singularity*, [Jän06, Chapter 4]. If f is holomorphic in a punctured neighbourhood $0 < |z - z_0| < r$ of z_0, then z_0 is called an *isolated singularity*, and f admits a *Laurent expansion*

$$f(z) \;=\; \sum_{n=-\infty}^{+\infty} a_n (z - z_0)^n$$

in that punctured neighbourhood. The part with the negative powers,

$$\sum_{n<0} a_n (z - z_0)^n,$$

is called the *principal part* of f at z_0. The classification of isolated singularities is determined by the principal part.

Definition 2.14 (Classification of isolated singularities). *An isolated singularity* z_0 *of* f *is called:*

- removable *if the principal part vanishes, i.e.* $a_n = 0$ *for all* $n < 0$*;*
- *a* pole of order k *if the principal part has finitely many terms and* $a_{-k} \neq 0$, $a_{-n} = 0$ *for* $n < -k$*;*
- *an* essential singularity *if the principal part has infinitely many nonzero terms.*

This definition can be found in [Jän06, Chapter 4].

In analytic combinatorics, the generating functions we encounter are rarely meromorphic. Instead, they typically have *algebraic singularities* of the form $(1 - z/\rho)^\alpha$ for $\alpha \notin \mathbb{Z}$, which are neither poles nor essential singularities in the classical sense, see [FS09, Section VI.1–VI.2]. These require the theory of algebraic functions and Puiseux expansions, developed in Section 2.7.

2.7 Algebraic Functions and Puiseux Expansions

The generating functions arising from the kernel method are algebraic, thus, they satisfy a polynomial equation

$$P(z, A(z)) = 0$$

for some bivariate polynomial P. This algebraic structure constrains the singularities of $A(z)$ in a precise way and is the key to extracting asymptotics [FS09, Chapter VII].

Definition 2.15 (Algebraic function)**.** *A function $A(z)$ is* algebraic *if there exists a polynomial $P(z, w) \not\equiv 0$ with $P(z, A(z)) = 0$. The* degree *of A is the degree of P in w, [FS09, chapter VII].*

To illustrate the meaning of this definition we look at an already familar example.

Example 2.16. The generating function $B(z) = \frac{1-\sqrt{1-4z}}{2z}$ of binary trees satisfies $zB(z)^2 - B(z) + 1 = 0$, so it is algebraic of degree 2.

2.7.1 Puiseux Expansions

Near a singularity z_0, an algebraic function cannot always be expanded as a power series in $(z - z_0)$. Instead, it admits a *Puiseux expansion*, a power series in a fractional power of $(z - z_0)$.

Theorem 2.17 (Newton–Puiseux, , [FS09, Appendix B.5])**.** *Let $A(z)$ be an algebraic function defined by $P(z, A(z)) = 0$. Near any singularity $z_0 \in \mathbb{C}$, every branch of $A(z)$ admits a convergent Puiseux expansion of the form*

$$A(z) \;=\; \sum_{k \geq k_0} c_k \, (z - z_0)^{k/q}$$

for some $q \in \mathbb{N}$ and $k_0 \in \mathbb{Z}$, with $c_{k_0} \neq 0$.

When $q = 1$ the expansion is a Laurent series; when $q = 2$ one obtains square-root behaviour, which is the generic case for the generating functions studied in this thesis.

2.7.2 Finite Branching

A key structural property of algebraic functions is that they have only finitely many singularities and finitely many branches.

Proposition 2.18 (Finite branching)**.** *Let $P(z, w)$ be an irreducible polynomial of degree d in w. The equation $P(z, w) = 0$ has exactly d branches (counted with multiplicity) at every non-singular point z_0. Singularities, where branches meet or ramify, are finite in number and are located at the roots of the discriminant* $\operatorname{disc}_w P(z, w)$*, see [FS09, Section VII.7].*

This finiteness is what makes the singularity analysis of algebraic generating functions tractable: there are only finitely many candidate dominant singularities, and near each one the Puiseux expansion is explicitly computable.

2.7.3 The Implicit Function Theorem

The implicit function theorem is the principal tool for understanding how the branches of an algebraic equation $P(z, w) = 0$ behave as analytic functions of z.

Theorem 2.19 (Implicit Function Theorem, [Jän06, Chapter 5])**.** *Let $F(z, w)$ be holomorphic in a neighbourhood of (z_0, w_0) with $F(z_0, w_0) = 0$ and $F_w(z_0, w_0) \neq 0$. Then there exists a unique holomorphic function $w(z)$ defined in a neighbourhood of z_0 such that*

$$w(z_0) = w_0 \qquad \text{and} \qquad F(z, w(z)) = 0.$$

In the context of the kernel method, $F = K(t, u)$ is the kernel, $z = t$ is the length variable, and $w = u$ represents the altitude variable. The implicit function theorem guarantees that, away from the branching points where $K_u(t, u) = 0$, each small root $u_i(t)$ is a well-defined analytic function of t.

2.7.4 Pringsheim's Theorem

The following classical theorem is indispensable for locating the dominant singularity of a generating function with non-negative coefficients.

Theorem 2.20 (Pringsheim, [FS09, Theorem IV.6])**.** *Let $f(z) = \sum_{n\geq 0} a_n z^n$ be a power series with radius of convergence $R \in (0, +\infty)$. If $a_n \geq 0$ for all $n \geq 0$, then $z = R$ is a singularity of f.*

Remark 2.21. *For a generating function $A(z) = \sum_{n\geq 0} A_n z^n$ counting combinatorial objects ($A_n \geq 0$), Pringsheim's theorem guarantees that the dominant singularity lies on the* positive real axis*. This is a crucial simplification: instead of searching all of $\mathbb{C}$ for the dominant singularity, we need only examine the positive real line.*

In combination with the Pólya–Fatou–Carlson theorem, which states that an algebraic power series with non-negative integer coefficients and radius of convergence 1 must be rational [FS09], Pringsheim's theorem implies that the dominant singularity ρ of a non-rational algebraic generating function satisfies $0 < \rho < 1$.

2.8 Singularity Analysis

Singularity analysis is the systematic method for extracting asymptotic information about the coefficients $[z^n]f(z)$ from the singular behaviour of $f(z)$ near its dominant singularity. It was developed into a general and precise framework by Flajolet and Odlyzko [FO90], and is presented comprehensively in [FS09].

2.8.1 The Transfer Theorem

The central result of singularity analysis is the *transfer theorem*, which converts singular expansions of $f(z)$ near ρ into asymptotic expansions of $[z^n]f(z)$.

Theorem 2.22 (Transfer theorem, square-root case, [FO90]). *Let $f(z)$ be analytic in a domain of the form*

$$\Delta(\rho, \phi) \;=\; \bigl\{z \in \mathbb{C} \;\big|\; |z| < \rho + \varepsilon,\; z \neq \rho,\; |\arg(z-\rho)| > \phi\bigr\}$$

for some $\varepsilon > 0$ and $0 < \phi < \pi/2$ (a so-called Δ-domain *or* Hankel domain*). If $f(z)$ has the singular expansion*

$$f(z) \;=\; f(\rho) - c\sqrt{1 - z/\rho} \;+\; O(1 - z/\rho) \quad \text{as } z \to \rho,\; z \in \Delta,$$

then

$$[z^n]\, f(z) \;\sim\; \frac{c}{2\sqrt{\pi}}\, n^{-3/2}\, \rho^{-n} \quad \text{as } n \to \infty.$$

2.9 Walks, Meanders, Bridges, and Excursions

We now apply the general framework of combinatorial classes to lattice paths. Let $\mathcal{S} \subset \mathbb{Z}$ be a finite set of allowed steps containing at least one positive and one negative element. A *lattice path* of length n is a sequence of heights $(w_0, w_1, \ldots, w_n)$ with $w_0 = 0$ and $w_k - w_{k-1} \in \mathcal{S}$ for all $k = 1, \ldots, n$. Since each step advances the path by exactly one unit horizontally, the length n is also the natural size measure, and the set of all lattice paths of a given length forms a combinatorial class in the sense of Section 2.3.

Depending on whether we impose constraints on the intermediate heights or the final altitude, we obtain four classical families of lattice paths. These are illustrated in Figure 2.4 and defined precisely below.

Definition 2.23 (The four path families). *Let $\mathcal{S} \subset \mathbb{Z}$ be a fixed step set and let $n \geq 0$.*

1. *A* walk *of length n is a sequence*

   $$w = (s_1, \ldots, s_n), \qquad s_i \in \mathcal{S}.$$

   The altitude after k steps *is defined as*

   $$\mathrm{alt}_k(w) = \sum_{i=1}^{k} s_i,$$

   and the final altitude *is $\mathrm{alt}(w) = \mathrm{alt}_n(w)$. No further constraints are imposed on intermediate altitudes or the final altitude.*

2. *A* meander *of length n is a walk with*

   $$\mathrm{alt}_k(w) \geq 0 \qquad \text{for all } 0 \leq k \leq n.$$

3. *A* bridge *of length n is a walk with*

   $$\mathrm{alt}(w) = 0.$$

   Intermediate altitudes may be negative.

4. An excursion *of length n is a walk that is simultaneously a meander and a bridge:*

$$\mathrm{alt}(w) = 0 \qquad \textit{and} \qquad \mathrm{alt}_k(w) \geq 0 \textit{ for all } k.$$

Equivalently, an excursion is a non-negative bridge.

These conventions stem from [ABBG20, Section 2].
These four families are ordered by increasing restrictiveness: every excursion is a meander, every excursion is a bridge, and every meander and bridge is a walk. The relationships are summarized in Figure 2.5.

Each of these four classes leads to a distinct generating function, and their analytic behaviour differs substantially, particularly in asymptotic regimes. A systematic treatment via the kernel method is developed in the following sections and is given in [ABBG20].

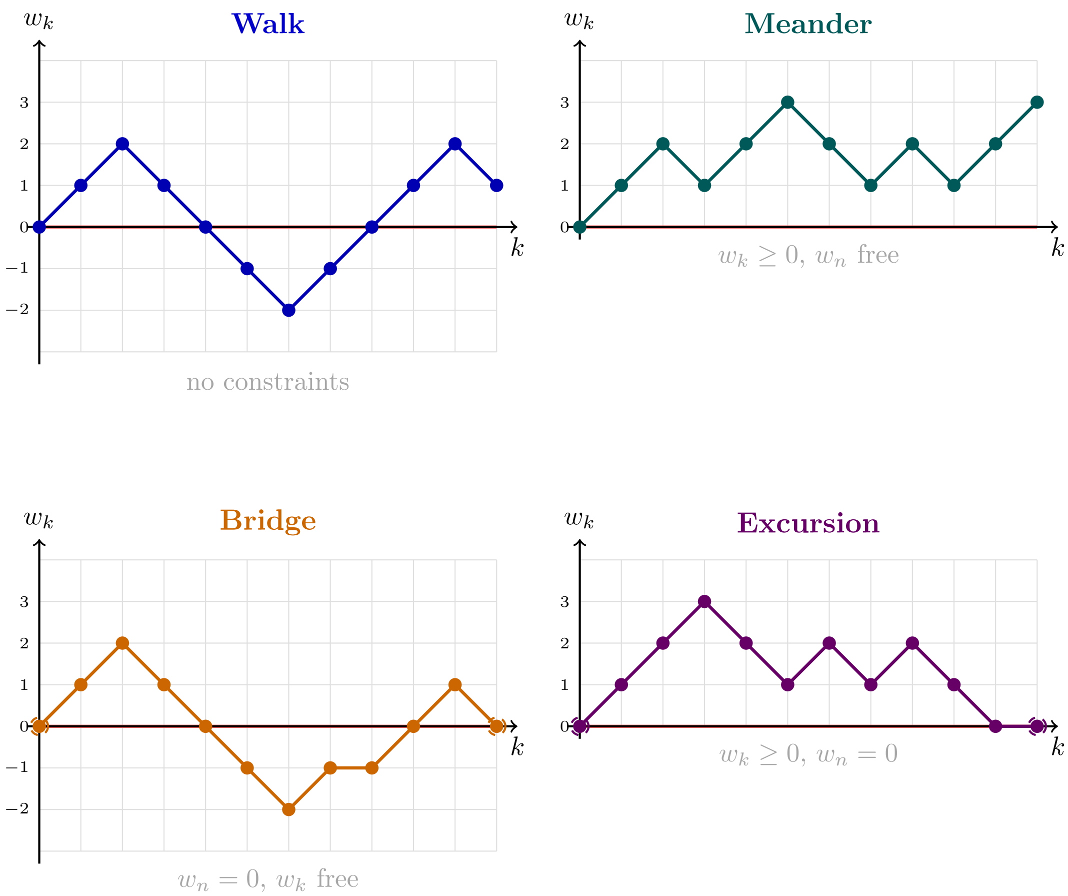


Figure 2.4: The four classical families of lattice paths with step set $\mathcal{S} = \{-1, 0, +1\}$, each of length 11. The red line marks height zero.

These four families are ordered by increasing restrictiveness, as summarized in the following inclusion diagram:

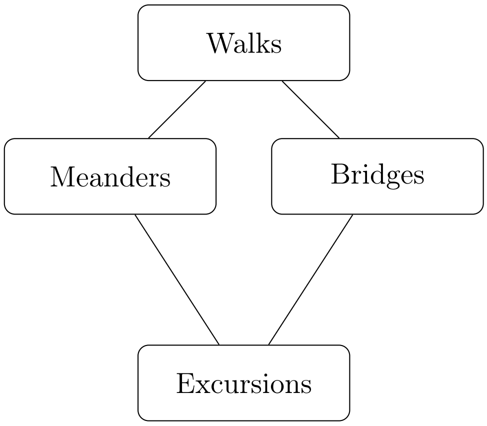


Figure 2.5: Hierarchy of lattice path classes.

Each family leads to a distinct generating function. Writing t for the variable marking length, we denote by $W(t)$, $M(t)$, $B(t)$, and $E(t)$ the generating functions for walks, meanders, bridges, and excursions respectively.

3 Patterns, Overlaps, and Automata

This chapter prepares the combinatorial framework needed for the vectorial kernel method. The focus here is on explaining the mechanisms that will later appear in a more formal and general setting. All technical results will be presented and proved in the next chapter.

We introduce the main ideas using a mock example with the step set

$$\mathcal{S} = \{-2, -1, 0, 1, 2\}.$$

This example will accompany us throughout the chapter and illustrate how forbidden patterns naturally lead to finite automata.

3.1 Walks and the Basic Setting

We consider lattice walks that start at height 0 and evolve by successive vertical increments from a fixed finite step set $\mathcal{S} \subset \mathbb{Z}$. Such walks can be visualized as paths in the plane, where each step moves one unit to the right and changes the height according to the chosen increment. In this way, the entire walk is determined by its sequence of steps.

We use the notation of Definition 2.23; in particular, $\mathrm{alt}_k(w)$ denotes the altitude of walk w after k steps.

At this stage, we impose no global restriction such as non-negativity or return to the origin. We are simply interested in enumerating admissible walks of a given length that satisfy certain local conditions. An example for such a walk can be seen in Figure 3.1. The figure illustrates how different step sizes lead to a path that fluctuates both upwards and downwards.

Let $\mathcal{S} \subset \mathbb{Z}$ be a fixed finite step set. Throughout this work, we assume that $\mathcal{S}$ contains at least one negative element, so that the notion of descent is meaningful. Classical examples include $\mathcal{S} = \{1, -1\}$ for Dyck paths or $\mathcal{S} = \{-1, 0, 1\}$ for Motzkin paths see Section 2.5. Our present example can be seen as an extension of these basic models.

3.2 Forbidden Patterns

We now impose a local restriction on the walks. Instead of allowing all possible sequences of steps, we declare a specific pattern to be forbidden. This means that certain configurations are not allowed to appear as consecutive steps within a walk.

Definition 3.1. *A* pattern *is a finite sequence*

$$p = (p_1, \ldots, p_\ell), \qquad p_i \in \mathcal{S}.$$

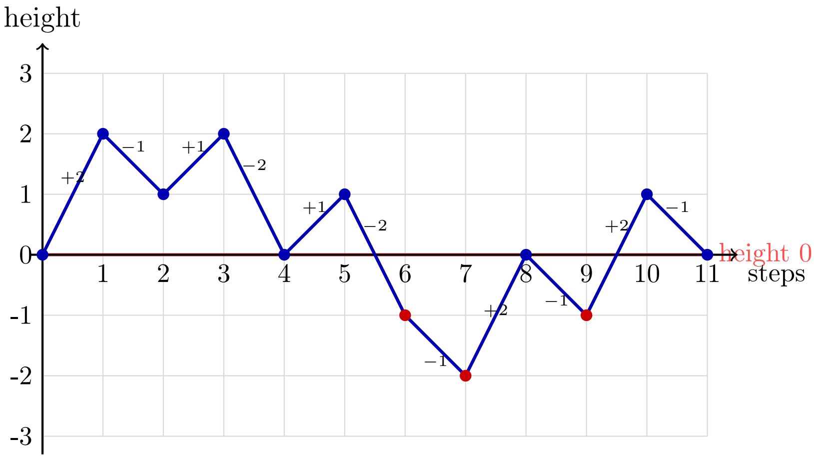


Figure 3.1: A walk of length 11 with step set $\mathcal{S} = \{-2, -1, 0, +1, +2\}$. The red line marks height 0; the shaded region shows where the walk dips below zero. Red dots indicate positions at negative altitude.

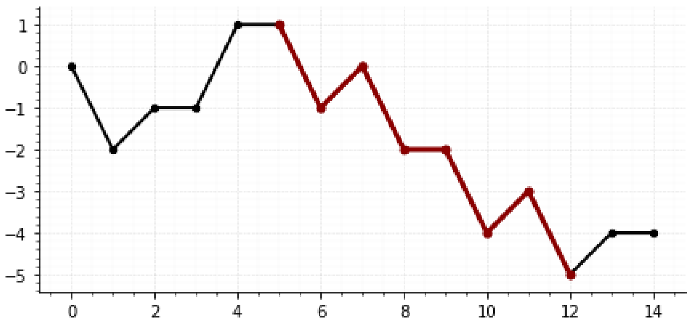


Figure 3.2: A walk containing the forbidden pattern highlighted in red.

A walk $w = (s_1, \ldots, s_n)$ *is said to* contain p *if there exists* i *such that*

$$(s_i, \ldots, s_{i+\ell-1}) = (p_1, \ldots, p_\ell).$$

Otherwise, w *is said to* avoid p*, [ABBG20].*

Throughout this chapter, we focus on the pattern

$$p = (-2, 1, -2, 0, -2, 1, -2)$$

which has length $\ell = 7$. Any walk containing this sequence will be declared illegal. An example of such a walk is shown in Figure 3.2, where the forbidden portion is highlighted. All other walks will be considered admissible.

The introduction of a forbidden pattern changes the nature of the problem. In the unrestricted setting, the admissibility of a step depends only on the current position. Once a pattern is forbidden, however, the situation changes: whether a step is allowed now depends on what has happened in the recent past.

To see this more concretely, consider a walk evolving step by step. As long as no part of the forbidden pattern has appeared, all steps from the set $\mathcal{S}$ are admissible. However, the situation changes as soon as the walk begins to resemble the beginning of the pattern. For instance, once the step -2 appears, the walk has matched the first element of the forbidden pattern. While this is not yet problematic, it creates the potential for completing the pattern in future steps.

To illustrate this mechanism, we consider our running example with step set $\mathcal{S} = \{-2, -1, 0, 1, 2\}$ and forbidden pattern $(-2, 1, -2, 0, -2, 1, -2)$. Let us examine a walk evolving step by step. We start with the following arbitrary steps

$$-1, -1, 0, 0, 2, 1, -1, 0, \ldots.$$

At this stage, no part of the forbidden pattern has appeared, and all steps behave as in the unrestricted model. The situation changes as soon as the first occurrence of -2 appears:

$$-1, -1, 0, 0, 2, 1, -1, 0, -2.$$

This step initiates a partial match with the forbidden pattern. Although no constraint is violated yet, the walk has entered a state in which future steps must be monitored more carefully.

Continuing the walk, suppose we obtain

$$-1, -1, 0, 0, 2, 1, -1, 0, \color{red}{-2, 1, -2}\color{black}.$$

At this point, the subsequence $(-2, 1, -2)$ forms a longer partial match, which we must monitor. Depending on which step is taken next the pattern will either break or continue. Lets say the next step of the pattern is

$$-1, -1, 0, 0, 2, 1, -1, 0, \color{red}{-2, 1, -2}\color{black}, 2.$$

The last step 2 breaks the match. Importantly, this does not simply erase all memory: one must check whether the resulting suffix still coincides with a shorter prefix of the pattern. This could be the case were the next step of height 1, we would get the following sequence

$$-1, -1, 0, 0, 2, 1, -1, 0, \color{red}{-2, 1, -2}\color{black}, 1.$$

While the forbidden pattern does not continue with a 1 as the fourth step, the pair of $-2, 1$ at the end does also give us a match of the forbidden pattern. This illustrates that partial matches can both grow and collapse, and that mismatches may still leave residual structure.

In general, a step is only forbidden if the walk is one step away from completing the pattern. In the present example, this occurs after the prefix $(-2, 1, -2, 0, -2, 1)$ has been matched. At this stage, the step -2 must be explicitly excluded, as it would complete the forbidden configuration. This example highlights the necessity of keeping track of partial matches in order to ensure admissibility.

In other words, the problem can only be solved if we create some sort of memory.

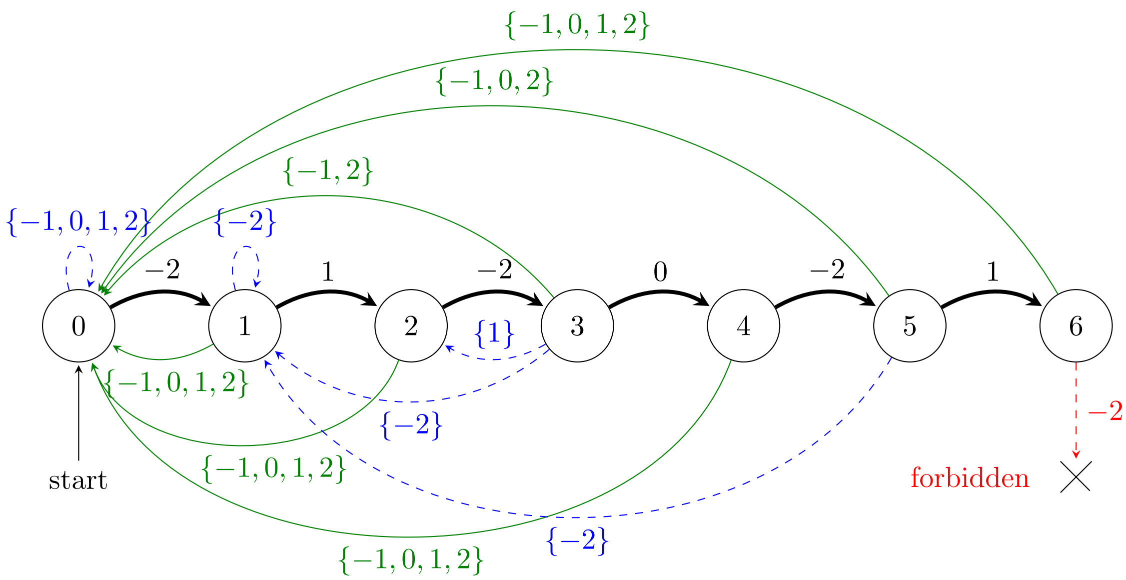
A natural way to encode this memory is through a finite automaton. The idea is to represent the current state of the walk not only by its position, but also by how much of the forbidden pattern has already been matched. The forbidden pattern has length 7, but reaching a full match would correspond to an illegal walk. Therefore, we do not model the state, that the full forbidden pattern has been matched. The theory of automata and much more information can be found in [HU79].

We introduce states labeled by integers

$$0, 1, 2, 3, 4, 5, 6,$$

where state k indicates that the last k steps of the walk coincide with the prefix $(p_1, \ldots, p_k)$ of the pattern. The state 0 represents the situation where no partial match is currently present.

The automaton evolves as the walk grows. Each time a new step is appended, the state is updated according to whether the current partial match is extended, broken, or replaced by a shorter overlapping match. In this way, the automaton keeps track of exactly the information needed to avoid completing the forbidden pattern. Since we never want to complete the pattern, the transition from the last state to -2 is removed from the automaton. This ensures that only admissible walks are generated.



The automaton contains three different types of transitions, each reflecting a different combinatorial situation. First, the forward transitions, drawn in black, correspond to extending a current partial match of the forbidden pattern. Each such transition increases the length of the matched prefix by one. Second, the green and blue transitions represent situations in which the current step breaks the partial match. The green arrows represent the automaton returning to state 0 and the dashed blue ones to intermediate states. Finally, the red transition at state 6 indicates the forbidden step. At this point, the walk has already matched the prefix $(-2, 1, -2, 0, -2, 1)$, and taking the step -2 would complete the forbidden pattern. This transition is therefore removed from the automaton, ensuring that only admissible walks are generated.

Altogether, the automaton encodes precisely how memory is updated, it tracks how partial matches are extended, destroyed, or restarted, and prevents the walk from ever completing the forbidden configuration.

The automaton provides a clear combinatorial description of admissible walks, but in order to apply generating function techniques, we must translate this structure into algebraic form. This is achieved by encoding the transitions of the automaton in a matrix.

To each state k we associate a generating function $F_k(t, u)$, where t marks the length of the walk and u marks its altitude. Thus, $F_k(t, u)$ counts all walks that end in state k. The collection of all these generating functions forms a vector, which will later satisfy a system of functional equations.

Each transition of the automaton contributes to this system. If there is a transition from state k to state j using a step s, then this corresponds to multiplying by tu^s: the variable t records the additional step, while u^s records the change in altitude. By collecting all such contributions, we obtain a transition matrix $A(u)$ that encodes the entire automaton. Note that the transition matrix $A(u)$ is not dependent on t, since we will add this later on.

This automaton is deterministic, for every state and every step in $\mathcal{S}$, there is exactly one transition describing how the memory is updated.

For this we want to define a transition matrix. This matrix must reflect the structure of the automaton. The rows of the matrix will correspond to the current state, and the columns the desired state. So the entry $A(k, j)$ describes the weight change that we must consider if we go from state k to state j. If there is no way to get from state k to state j the matrix entry will be 0.

For example, consider transitions starting from state 0. Taking a step -2 initiates a new partial match and leads to state 1, contributing the term u^{-2} to the entry $A_{0,1}(u)$. All other steps do not produce any potential match and remain in state 0, contributing the terms u^{-1}, 1, u, and u^2 to $A_{0,0}(u)$.

A similar reasoning applies to the other rows. For instance, from state 1, the step 1 extends the current match and leads to state 2, contributing u^1 to $A_{1,2}(u)$. The step -2 restarts the matching process and leads back to state 1, contributing u^{-2} to $A_{1,1}(u)$. All remaining steps break the current match and return to state 0, contributing the corresponding monomials to $A_{1,0}(u)$.

In this way, each entry of the matrix records all possible transitions between states, weighted according to their effect on the altitude. The matrix therefore provides an algebraic encoding of the automaton.

Ordering the states as $(0, 1, 2, 3, 4, 5, 6)$, the transition matrix has the form

$$A(u) = \begin{pmatrix} u^{-1}+1+u+u^2 & u^{-2} & 0 & 0 & 0 & 0 & 0 \\ u^{-1}+1+u^2 & u^{-2} & u & 0 & 0 & 0 & 0 \\ u^{-1}+1+u+u^2 & 0 & 0 & u^{-2} & 0 & 0 & 0 \\ u^{-1}+u^2 & u^{-2} & u & 0 & 1 & 0 & 0 \\ u^{-1}+1+u+u^2 & 0 & 0 & 0 & 0 & u^{-2} & 0 \\ u^{-1}+1+u^2 & u^{-2} & 0 & 0 & 0 & 0 & u \\ u^{-1}+1+u+u^2 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

.

We observe that, in all of the rows but the last, the sum of all entries is equal to the step polynomial

$$P(u) = u^{-2} + u^{-1} + 1 + u + u^2.$$

This reflects the fact that in all states except the last, all steps are allowed. The only exception occurs when the walk is close to completing the forbidden pattern, in which case one step, in our example -2 must be excluded. This is precisely what happens in the last row of the matrix.

Each column corresponds to a fixed target state, and all entries in that column describe the different ways of reaching it. For instance, the column corresponding to state 1 contains only terms of the form u^{-2}, since the only way to enter this state is by taking a step -2. In contrast, the first column collects all transitions that break the current partial match and return to state 0.

The transition matrix allows us to model one step taken in the sense of generating functions. If we write the generating functions of the different states of length n

$$\mathbf{F}(t,u)_n = \begin{pmatrix} F_0(t,u)_n \\ F_1(t,u)_n \\ \vdots \\ F_6(t,u)_n \end{pmatrix},$$

then the generating functions of length $n+1$ can be expressed by

$$\mathbf{F}(t,u)_{n+1} = tA(u)\mathbf{F}(t,u)_n$$

this connection can be used in Chapter 4 to express this as a functional equation and we will also present a solution using the vectorial kernel method, leading to explicit expressions and asymptotic results.

3.3 Overlaps and Autocorrelation

We have already encountered situations in which overlaps of the forbidden pattern influence the structure of the transition matrix.

In this section, we study these overlaps systematically. This leads to the concept of *autocorrelation*, which captures how a pattern can overlap with itself. The associated *autocorrelation polynomial* will play a central role later, as it encodes precisely the combinatorial information needed to control these overlaps in the functional equations.

Definition 3.2. *Let $p = (p_1, \ldots, p_\ell)$ be a pattern. For $k \in \{1, \ldots, \ell\}$, the prefix $(p_1, \ldots, p_k)$ is called a* presuffix *if it coincides with the suffix $(p_{\ell-k+1}, \ldots, p_\ell)$.*

To visualize this idea, we align two copies of the pattern and look for overlaps between a prefix and a suffix.

For example, consider the pattern

$$p = (-2, 1, -2, 0, -2, 1, -2).$$

An overlap of length $k = 3$ can be seen as follows:

$$\begin{array}{l} -2, 1, -2, 0, \underline{-2, 1, -2} \\ \underline{-2, 1, -2}, 0, -2, 1, -2 \end{array}$$

Here, the underlined part indicates the common prefix and suffix. The remaining part of the second copy is called the *complement* of the presuffix. Note that the definition also includes the trivial case where the entire pattern is taken as both prefix and suffix. Thus, this is the trivial presuffix since every pattern has it.

$$\begin{array}{l} \underline{-2, 1, -2, 0, -2, 1, -2} \\ \underline{-2, 1, -2, 0, -2, 1, -2} \end{array}$$

Similarly, for $k = 1$ we obtain the overlap

$$\begin{array}{l} -2, 1, -2, 0, -2, 1, \underline{-2} \\ \underline{-2}, 1, -2, 0, -2, 1, -2 \end{array}$$

Definition 3.3. *Let $p = (p_1, \ldots, p_\ell)$ be a pattern, and let $\mathcal{P}_p$ denote the set of all presuffixes of p. This set is nonempty, since it always contains the full pattern.*

For $s \in \mathcal{P}_p$, we denote by $\overline{s}$ the complement *of s, defined by the relation*

$$p = s\,\overline{s}.$$

The autocorrelation polynomial *of p is defined as*

$$R(t, u) = \sum_{s \in \mathcal{P}_p} t^{\ell - |s|} u^{\mathrm{alt}(\overline{s})} = \sum_{s \in \mathcal{P}_p} t^{|\overline{s}|} u^{\mathrm{alt}(\overline{s})}.$$

Each term in this sum corresponds to one possible overlap of the pattern with itself. The exponent $\ell - |s|$ records how many additional steps are needed to complete a full occurrence of the pattern after an overlap of length $|s|$. The exponent of u records the total altitude change contributed by the remaining part of the pattern, that is, by the complement of the presuffix. The complement $\overline{s}$ represents precisely the steps that must be appended after an overlap in order to complete a full occurrence of the pattern.

In our running example, the admissible values of k are 7, 1, and 3. Computing the corresponding contributions gives us

$$R(t,u) = 1 + t^6u^{-4} + t^4u^{-3}.$$

The autocorrelation polynomial will later appear naturally in the kernel equation of the functional system governing pattern-avoiding walks. It provides a compact encoding of all possible self-overlaps of the pattern and is therefore essential for controlling repeated occurrences.

In the next Proposition we want to study the structure of the autocorrelation vector as it will many times. This is an alternative way to obtain the autocorrelation vector. In chapter 4 we will introduce another method to obtain this but it involves matrix inversion and is therefore numerically very costly.

Proposition 3.4 (Structure of the autocorrelation vector, [ABBG20])**.** *Proposition 42.*

The autocorrelation vector

$$v := \mathrm{adj}(I - tA) \cdot \mathbf{1}$$

satisfies

$$v = R(t,u)\,\mathbf{1} - s(t,u), \tag{3.1}$$

where the j-th entry of $s(t,u)$ is a polynomial, namely the generating function of a finite set S_j of walks (which we call the subtracted set *of walks). The subtracted set S_j consists of all walks of length smaller than ℓ having the factorization*

$$w \cdot a_\ell \cdot \overline{q},$$

where

1. *w is any walk starting in state X_j and ending in state X_ℓ,*

2. *a_ℓ is the last step of the pattern p,*

3. *$\overline{q}$ corresponds to a term of $R(t,u)$ (that is, $\overline{q}$ is the complement of some presuffix q of p).*

Proof. We begin by using the definition of the autocorrelation vector that we will learn only in the next chapter.

$$\vec{v} := \mathrm{adj}(I - tA)\,\mathbf{1}. \tag{3.2}$$

From Theorem 2.2 and Theorem 4.3 we know that

$$(I - tA)\,\mathrm{adj}(I - tA) = K(t,u)\,I, \tag{3.3}$$

where $K(t,u)$ denotes the kernel polynomial.

Multiplying (3.3) on the right by $\mathbf{1}$ and using the definition of $\vec{v}$ in 3.2, we obtain

$$\begin{aligned}(I - tA)\vec{v} &= (I - tA)\,\mathrm{adj}(I - tA)\mathbf{1} \\ &= K(t,u)I\mathbf{1} \\ &= K(t,u)\mathbf{1}.\end{aligned} \tag{3.4}$$

Next, we recall the structural identity for the matrix A that we have in (3.5),

$$A\mathbf{1} = P(u)\mathbf{1} - u^{a_\ell}\vec{e}_\ell, \tag{3.5}$$

where $\vec{e}_\ell \in \mathbb{R}^\ell$ denotes the ℓ-th standard basis vector, i.e. the vector whose last component equals 1 and all other components equal 0.

We now introduce the auxiliary vector

$$\vec{s} := R(t,u)\mathbf{1} - \vec{v}. \tag{3.6}$$

Our goal is to determine the structure of $\vec{s}$. To this end, we apply the operator $(I - tA)$ to both sides of 3.6:

$$\begin{aligned}(I - tA)\vec{s} &= (I - tA)\big(R(t,u)\mathbf{1} - \vec{v}\big) \\ &= (I - tA)R(t,u)\mathbf{1} - (I - tA)\vec{v}.\end{aligned} \tag{3.7}$$

We now manipulate the terms to get a different representation for $\vec{s}$. For this purpose we split up the right hand term of (3.7) into two parts and look at them separately.

We start with the computation of $(I - tA)R(t,u)\mathbf{1}$.

Since $R(t,u)$ is a polynomial, it commutes with matrices. Therefore,

$$\begin{aligned}(I - tA)R(t,u)\mathbf{1} &= R(t,u)(I - tA)\mathbf{1} \\ &= R(t,u)\mathbf{1} - tR(t,u)A\mathbf{1}.\end{aligned} \tag{3.8}$$

Now substitute (3.5) into the above

$$\begin{aligned}(I - tA)R(t,u)\mathbf{1} &= R(t,u)\mathbf{1} - tR(t,u)\big(P(u)\mathbf{1} - u^{a_\ell}\vec{e}_\ell\big) \\ &= R(t,u)\mathbf{1} - tR(t,u)P(u)\mathbf{1} + tR(t,u)u^{a_\ell}\vec{e}_\ell.\end{aligned} \tag{3.9}$$

For the second term of (3.7) we want to manipulate $(I - tA)\vec{v}$.

From (3.3) we know that

$$(I - tA)\vec{v} = K(t,u)\mathbf{1}.$$

Substituting both of these manipulations into the original equation (3.7), we obtain

$$(I - tA)\vec{s} = \Big(R(t,u)\mathbf{1} - tR(t,u)P(u)\mathbf{1} + tR(t,u)u^{a_\ell}\vec{e}_\ell\Big) - K(t,u)\mathbf{1}. \tag{3.10}$$

We now substitute the expression for the kernel (from Proposition 4.3):

$$K(t,u) = (1 - tP(u))R(t,u) + t^{|p|}u^{\mathrm{alt}(p)}. \tag{3.11}$$

We recall that $|p|$ is the length of the forbidden pattern and is the same as ℓ Multiplying this equality by by $\mathbf{1}$ gives

$$K(t,u)\mathbf{1} = (1 - tP(u))R(t,u)\mathbf{1} + t^{|p|}u^{\mathrm{alt}(p)}\mathbf{1}. \tag{3.12}$$

Insert the information from above into (3.10):

$$\begin{aligned}(I - tA)\vec{s} = {} & R(t,u)\mathbf{1} - tR(t,u)P(u)\mathbf{1} + tR(t,u)u^{a_\ell}\vec{e}_\ell \\ & - (1 - tP(u))R(t,u)\mathbf{1} - t^{|p|}u^{\mathrm{alt}(p)}\mathbf{1}.\end{aligned} \tag{3.13}$$

Now expand the term

$$(1 - tP(u))R(t,u)\mathbf{1} = R(t,u)\mathbf{1} - tP(u)R(t,u)\mathbf{1}.$$

Substituting this gives

$$\begin{aligned}(I - tA)\vec{s} = {} & R(t,u)\mathbf{1} - tR(t,u)P(u)\mathbf{1} + tR(t,u)u^{a_\ell}\vec{e}_\ell \\ & - R(t,u)\mathbf{1} + tP(u)R(t,u)\mathbf{1} - t^{|p|}u^{\mathrm{alt}(p)}\mathbf{1}.\end{aligned} \tag{3.14}$$

We now observe cancellation. Thus only two terms remain

$$(I - tA)\vec{s} = tu^{a_\ell}R(t,u)\vec{e}_\ell - t^{|p|}u^{\mathrm{alt}(p)}\mathbf{1}. \tag{3.15}$$

Finally, multiply equation above from the left by the term $(I - tA)^{-1}$

$$\vec{s} = (I - tA)^{-1}\vec{e}_\ell\, tu^{a_\ell}R(t,u) - (I - tA)^{-1}\mathbf{1}\, t^{|p|}u^{\mathrm{alt}(p)}. \tag{3.16}$$

Now that we have found an expression for $\vec{s}$ we want to interpret it combinatorially to see whether it fulfills the claims made in the proposition.

Recall that

$$(I - tA)^{-1} = \sum_{n\geq 0}(tA)^n. \tag{3.17}$$

By construction of the automaton, the (i, j)-entry of $(tA)^n$ is the generating function of walks of length n that start in state X_i, end in state X_j, and avoid the pattern p. Summing over n therefore shows that the (i, j)-entry of $(I - tA)^{-1}$ is precisely the generating function of all p-avoiding walks from X_i to X_j.

Consider the first term in (3.16). The vector $(I - tA)^{-1}\vec{e}_\ell$ extracts the ℓ-th column of $(I - tA)^{-1}$. This is the last column since A is of the form $\ell \times \ell$. Hence its j-th component enumerates all p-avoiding walks that start in state X_j and end in state X_ℓ.

Multiplication by tu^{a_ℓ} corresponds to appending the final step a_ℓ of the pattern p. Multiplication by $R(t, u)$ corresponds to appending the complement $\overline{q}$ of some presuffix q of p.

Thus the first term enumerates walks of the form

$$w \cdot a_\ell \cdot \overline{q},$$

where w avoids p and ends in state X_ℓ.

The second term in (3.16),

$$(I - tA)^{-1}\mathbf{1}\, t^{|p|}u^{\mathrm{alt}(p)},$$

enumerates all p-avoiding walks, followed by an appended full copy of the pattern p.

Both constructions produce walks ending with p. Walks with exactly one occurrence of p at the end, or two overlapping occurrences, appear in both terms and cancel in the subtraction. The only remaining walks are those for which the prefix w is too short to contain p, that is, of length strictly smaller than $|p|$. This set is finite, and its generating function is therefore a polynomial.

Hence each component of $\vec{s}$ is a polynomial, and we obtain

$$\vec{v} = R(t, u)\mathbf{1} - \vec{s}. \qquad \square$$

3.4 Kernel Structure

This section establishes a key structural result about the roots of the kernel polynomial, which will be the central object of the vectorial kernel method developed in Chapter 4. We state and prove it here because the argument is entirely self-contained and geometric, relying only on the Newton polygon introduced below, and because it will be needed repeatedly once the kernel method is in place.

The result concerns a bivariate Laurent polynomial $K(t, u)$ in the variables t (marking path length) and u (marking altitude). Its precise definition — in terms of the step polynomial $P(u)$, the autocorrelation polynomial $R(t, u)$ of the forbidden pattern, and the pattern data $|p|$ and $\mathrm{alt}(p)$, is given in Definition 4.1 in Chapter 4. For the purposes of the present proposition, all that matters is that $K(t, u)$ is a Laurent polynomial in u with analytic coefficients in t, and that $K(0, u) \neq 0$. A root $u(t)$ of $K(t, u) = 0$ is called *small* if $\lim_{t\to 0} u(t) = 0$ and *large* if $\lim_{t\to 0} |u(t)| = \infty$; the proposition below shows that every root is one or the other, and counts each type.

Proposition 3.5 (Small and large roots of the kernel K)**.** *All roots $u(t)$ of $K(t,u)$ are either* small *(meaning $\lim_{t\to 0} u(t) = 0$) or* large *(meaning $\lim_{t\to 0} |u(t)| = \infty$), [ABBG20, Proposition 4.4].*

Let d_K denote the degree of $K(t,u)$ in u, i.e.

$$d_K = \max\{j \mid [u^j]K(t,u) \neq 0\},$$

and let ℓ_K denote the lowest power of u in the monomials of K, i.e.

$$\ell_K = \min\{j \mid [u^j]K(t,u) \neq 0\}.$$

Then K has e small roots and f large roots, where

$$e = \max(0, -\ell_K) \qquad \text{and} \qquad f = \max(0, d_K).$$

Remark 3.6. *We note that ℓ_K here denotes the lowest power of u in K, and should not be confused with the path length $|p| = \ell$.*

3.4.1 Newton Polygons

Before turning to the technical details, we briefly introduce a geometric tool that will play a key role later: the *Newton polygon*. The Newton polygon construction is classical; see e.g. [Wal62, Chapter IV].

Roughly speaking, every monomial $t^{r_1}u^{r_2}$ of a polynomial can be represented as a point (r_1, r_2) in the plane. By plotting all these points and taking their convex hull, one obtains a polygonal shape whose lower boundary, the Newton polygon, encodes important information about the behaviour of the solutions $u(t)$ that become central later.

The idea is surprisingly visual:

- The *points on the axes* often correspond to the simplest or most dominant terms.
- The *edges* of the polygon determine how different terms balance each other.
- In particular, the *slope of an edge* predicts the leading behaviour of a solution $u(t)$ as $t \to 0$.

One can therefore read off qualitative information about the roots directly from a simple diagram, without solving the equation explicitly. Figure 3.3 shows a typical example of such a picture. We will return to this construction in detail below and use it to systematically analyze the small roots of the kernel equation. To illustrate, consider the polynomial

$$F(t,u) = u - tu^2 - t^2 - tu.$$

We determine the nature and count of the roots $u(t)$ of $F(t,u) = 0$ as $t \to 0$ using the Newton polygon alone, without solving the equation.

Each monomial $t^{r_1}u^{r_2}$ corresponds to the point (r_1, r_2):

$$u \;\to\; (0,1), \qquad -tu^2 \;\to\; (1,2), \qquad -t^2 \;\to\; (2,0), \qquad -tu \;\to\; (1,1).$$

Figure 3.3 shows the four lattice points and their convex hull. The lower-left boundary, the Newton polygon, has two edges:

- a **green edge** from $(0, 1)$ to $(2, 0)$ with slope $-\frac{1}{2}$, and
- a **red edge** from $(0, 1)$ to $(1, 2)$ with slope $+1$.

The interior point $(1, 1)$, corresponding to $-tu$, lies strictly inside the convex hull.

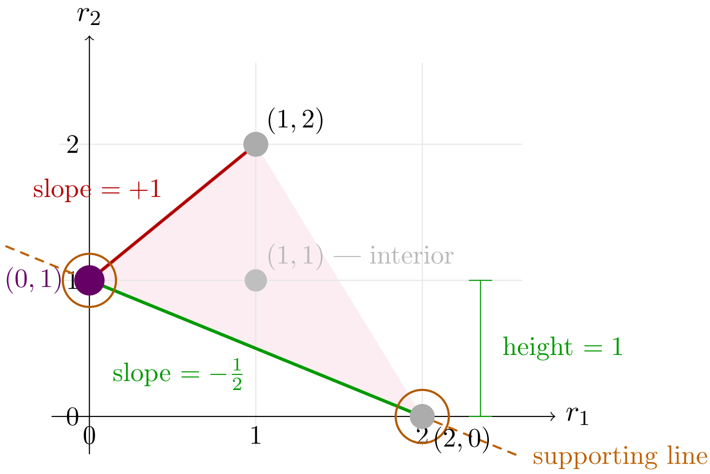


Figure 3.3: Newton polygon of $F(t, u) = u - tu^2 - t^2 - tu$.

We will need this information in the proof of Proposition 3.5.

Proof. We analyze the behaviour of solutions $u = u(t)$ of $K(t, u) = 0$ as $t \to 0$. Throughout we focus on small roots. The argument for large roots follows by the substitution $u \mapsto 1/u$, which reflects the Newton polygon vertically and exchanges small and large roots. The charakter of this proof is of the following form: We take the kernel $K(t, u)$ and multiply it with a power u such that there are no negative powers of $K(t, u)$ left. Then we look at the shifted polynomial and analyze its structure via the Newton polygon. From the Newton polygon we get some observation which part of the Polynomial contributes the most as we go to $t \to 0$.

Recall that the kernel is defined as

$$K(t, u) = (1 - t\,P(u))\,R(t, u) + t^{|p|}\,u^{\mathrm{alt}(p)}.$$

The step polynomial $P(u)$ must contain negative powers of u, namely one for each negative step in $\mathcal{S}$, so $K(t, u)$ is in general a Laurent polynomial in u, not a polynomial. To apply standard polynomial theory we clear these negative powers. Set

$$e := \max(0, -\ell_K),$$

and define

$$\widetilde{K}(t, u) := u^e\,K(t, u).$$

By construction $\widetilde{K}$ is a polynomial in u. Since $u^e \neq 0$ for $u \neq 0$, the nonzero roots of K and $\widetilde{K}$ coincide exactly, so it suffices to count the small roots of $\widetilde{K}$. If $\ell_K \geq 0$ then $e = 0$ and $\widetilde{K} = K$ already.

Now we want to build the Newton Polygon for this $\widetilde{K}(t,u)$ polynomial.
Write

$$\widetilde{K}(t,u) = \sum_{(r_1,r_2)} c_{r_1,r_2}\, t^{r_1} u^{r_2},$$

where the sum runs over finitely many pairs $(r_1, r_2) \in \mathbb{N}^2$ with $c_{r_1,r_2} \neq 0$. To each such monomial we associate the lattice point (r_1, r_2) in the plane, and we let $\mathcal{P}$ denote this finite set.

The *Newton polygon* of $\widetilde{K}$ is the boundary of the convex hull of $\mathcal{P}$. Next we analyze the set $\mathcal{P}$. Two points are guaranteed to lie in $\mathcal{P}$:

- The point $(0, e)$: the kernel K contains the term $1 \cdot R(t,u)$, and $R(t,u)$ itself contains 1 as a summand, since it encodes the empty overlap word. After multiplication by u^e this contributes the monomial u^e, corresponding to the point $(0, e)$.
- A point $(\lambda, 0)$ with $\lambda > 0$: by definition of e, the lowest power of u in K is ℓ_K, and after multiplication by u^e it becomes $u^0 = 1$. Since the step set contains at least one negative step, the monomial with u-power zero is always accompanied by a positive power of t, so $\lambda > 0$.

Of particular interest are the edges of the Newton polygon that are visible when the polygon is viewed from the left — that is, the edges connecting $(0, e)$ down to $(\lambda, 0)$ along the lower-left boundary. These are the edges of negative slope.

Since $\widetilde{K}(t,u)$ is a polynomial in u with analytic coefficients in t, the Newton–Puiseux theorem 2.17 guarantees that every root admits a *Puiseux expansion*

$$u(t) = C\, t^{\nu} + \text{higher order terms}, \qquad C \neq 0, \quad \nu \in \mathbb{Q}.$$

A root is small if and only if $\nu > 0$.

To determine which values of ν are possible, we substitute $u \approx Ct^{\nu}$ into a monomial, giving us

$$t^{r_1} u^{r_2} \approx C^{r_2}\, t^{r_1 + \nu r_2}.$$

The quantity $E(r_1, r_2) = r_1 + \nu r_2$ is the *effective t-exponent* of this monomial. As $t \to 0$, monomials with larger effective exponent vanish faster and become negligible.

For $\widetilde{K}(t, u(t)) = 0$ to hold, the leading-order terms cannot cancel unless at least two monomials share the minimal effective exponent. A single dominant monomial with nonzero coefficient cannot sum to zero on its own. Therefore, the dominant monomials must satisfy

$$r_1 + \nu r_2 = \text{const},$$

which defines a straight line in the (r_1, r_2)-plane. The dominant monomials thus correspond to lattice points of $\mathcal{P}$ lying on a common supporting line. By definition of the convex hull, such supporting lines are exactly the edges of the Newton polygon.

Let Σ be an edge of the Newton polygon with slope $-\beta/\alpha$, where $\alpha, \beta > 0$. The condition that points on Σ share a constant effective exponent gives

$$\frac{\Delta r_2}{\Delta r_1} = -\frac{\beta}{\alpha} \implies \nu = \frac{\alpha}{\beta} > 0.$$

Hence edges of *negative* slope correspond exactly to small roots ($\nu > 0$), while edges of positive slope yield $\nu < 0$ and correspond to large roots.

Fix a segment Σ with endpoints (γ_1, γ_2) (upper left) and (γ_3, γ_4) (lower right), so that $\gamma_2 > \gamma_4$. To leading order,

$$\widetilde{K}(t, u(t)) \;\approx\; \sum_{(r_1,r_2)\in\Sigma} c_{r_1,r_2}\, t^{r_1} u^{r_2},$$

since all off-segment monomials have strictly larger effective exponent and are negligible. Substituting $u = Ct^{\alpha/\beta}$ and factoring out the common power of t yields a polynomial in C of degree $\gamma_2 - \gamma_4$. By the fundamental theorem of algebra this polynomial has exactly $\gamma_2 - \gamma_4$ roots, each giving a valid Puiseux expansion. Hence Σ contributes exactly $\gamma_2 - \gamma_4$ small roots, a number we call the *height* of Σ.

The lower-left boundary of the Newton polygon is a polygonal path from $(0, e)$ to $(\lambda, 0)$. The heights of all its segments sum to the total vertical drop:

$$\sum_j \text{height}(\Sigma_j) \;=\; e - 0 \;=\; e.$$

Hence the total number of small roots of $\widetilde{K}$, and therefore of K, is exactly

$$e = \max(0, -\ell_K). \qquad \square$$

4 The Vectorial Kernel Method

This chapter derives explicit generating functions for the four classical families of lattice paths: walks, bridges, meanders, and excursions in the presence of a forbidden pattern p.

The central objects are the *autocorrelation polynomial* $R(t,u)$ and the *kernel*, defined as follows.

Definition 4.1 (Kernel). *Let* $P(u) = \sum_{s\in\mathcal{S}} u^s$ *be the step polynomial and let* p *be a forbidden pattern of length* $|p|$ *with final altitude* $\operatorname{alt}(p)$. *The* kernel *is*

$$K(t,u) \;=\; (1 - tP(u))\,R(t,u) + t^{|p|}u^{\operatorname{alt}(p)}.$$

Its small roots $u_1(t),\ldots,u_e(t)$ (those satisfying $\lim_{t\to 0} u_i(t) = 0$) carry all the analytic information needed.

The number of small roots equals $e = \max(0, -\ell_K)$, the absolute value of the most negative step, as established in Proposition 3.5.

The four path families and their generating functions are summarised in Table 4.1 below. Each formula is proved in this chapter; together they form the main output of the vectorial kernel method.

Family	**Constraint**	**Generating function**	**Theorem**
Walks	none	$W(t,u) = \dfrac{R(t,u)}{K(t,u)}$	4.2
Bridges	$w_n = 0$	$B(t) = -\displaystyle\sum_{i=1}^{e} \frac{u_i'(t)}{u_i(t)} \cdot \frac{R(t,u_i(t))}{K_t(t,u_i(t))}$	4.2
Meanders	$w_k \geq 0$	$M(t,u) = \dfrac{G(t,u)}{u^e K(t,u)} \displaystyle\prod_{i=1}^{e}(u - u_i(t))$	4.4
Excursions	$w_k \geq 0$, $w_n = 0$	$E(t) = M(t,0)$	4.5

Table 4.1: Generating functions for the four path families avoiding a pattern p, derived via the vectorial kernel method. Here $u_1(t),\ldots,u_e(t)$ denote the small roots of $K(t,u) = 0$, and $G(t,u)$ is a polynomial in u characterised during the proof of Theorem 4.4.

The four formulas are related by increasing restrictiveness. The walk formula $W(t,u) = R(t,u)/K(t,u)$ is the most general: no constraint is imposed on intermediate heights or the final altitude, and the generating function is simply a ratio of known polynomials. The bridge formula imposes the endpoint condition $w_n = 0$, which is extracted from $W(t, u)$ by a residue calculation at the small roots; the result is a sum over those roots. The meander formula imposes non-negativity throughout, which introduces a boundary correction and requires the small roots as explicit factors in the numerator; the function $G(t, u)$ captures the remaining combinatorial information and is determined by substituting each small root into the functional equation. The excursion formula combines both constraints and is recovered simply by evaluating the meander generating function at $u = 0$, since setting the altitude variable to zero selects paths that end at height zero.

4.1 Walks Avoiding a Generic Pattern

We begin with the fundamental result describing the generating function of walks avoiding a fixed pattern. This proof also reveals essential structural properties of the kernel and the autocorrelation polynomial, which will be used repeatedly in later sections.

Theorem 4.2 (Generating function of walks avoiding a generic pattern, [ABBG20] theorem 3.1)**.** *Let $\mathcal{S}$ be a finite step set and let p be a pattern with steps in $\mathcal{S}$.*

1. *The bivariate generating function of walks avoiding the pattern p is given by*

$$W(t, u) = \frac{R(t, u)}{K(t, u)}.$$

If the final altitude is not recorded, this reduces to

$$W(t) = W(t, 1) = \frac{1}{1 - tP(1) + t^{\ell}/R(t, 1)}.$$

2. *The generating function of bridges avoiding the pattern p satisfies*

$$B(t) = -\sum_{i=1}^{e} \frac{u_i'(t)}{u_i(t)} \frac{R(t, u_i(t))}{K_t(t, u_i(t))},$$

where $u_1(t), \ldots, u_e(t)$ denote the small roots of the kernel $K(t, u)$.

This proof follows the approach of the vectorial kernel method developed in [ABBG20], with a few more intermediate steps and explanations made.

Proof of Theorem 4.2, Part (1). The coefficient extraction $[t^n][u^k]W(t, u)$ counts the number of walks of length n ending at altitude k and avoiding the pattern p. To refine this enumeration, we decompose walks according to the state of the pattern-matching automaton.

Let $X_1, \ldots, X_\ell$ denote the states of the automaton associated with p, and let $W_i(t, u)$ be the generating function of walks that avoid p and end in state X_i. The total generating function then satisfies

$$W(t, u) = (W_1(t, u), \ldots, W_\ell(t, u))\, \mathbf{1},$$

where $\mathbf{1}$ denotes the column vector of ones.

A walk is either empty, in which case it ends in the initial state X_1, or it is obtained by appending one step to a shorter walk. This leads to the vectorial functional equation

$$(W_1, \dots, W_\ell) = (1, 0, \dots, 0) + t(W_1, \dots, W_\ell)A,$$

where A is the transition matrix of the automaton. Rewriting this equation yields

$$(W_1, \dots, W_\ell)(I - tA) = (1, 0, \dots, 0).$$

Multiplying by $(I - tA)^{-1}$ and applying the adjugate formula 2.2 gives

$$(W_1, \dots, W_\ell) = (1, 0, \dots, 0)\frac{\operatorname{adj}(I - tA)}{\det(I - tA)}.$$

Summing over all states, we obtain

$$W(t, u) = \frac{(1, 0, \dots, 0)\operatorname{adj}(I - tA)\mathbf{1}}{\det(I - tA)}.$$

We now derive an alternative expression for $W(t, u)$ using combinatorial considerations. Let $W^{\{p\}}(t, u)$ denote the generating function of walks that end with exactly one occurrence of the pattern p and contain no other occurrence of p.
Appending one step to a p-avoiding walk either produces another p-avoiding walk or creates a terminal occurrence of p. We want to translate this principle into an equation where appending one step signifies multiplying the generating function by t and the step polynomial. This yields the relation

$$tP(u)W(t, u) = W(t, u) - 1 + W^{\{p\}}(t, u),$$

where the subtraction of 1 accounts for the empty walk that cannot be achieved by adding a step.

Because the pattern p may overlap with itself, appending p to a p-avoiding walk may create multiple occurrences. The autocorrelation polynomial $R(t, u)$ encodes these overlaps. Accounting for them leads to the identity

$$W(t, u)\, t^\ell u^{\operatorname{alt}(p)} = W^{\{p\}}(t, u)R(t, u).$$

Solving the resulting system of equations for $W(t, u)$ yields

$$W(t, u) = \frac{R(t, u)}{(1 - tP(u))R(t, u) + t^\ell u^{\operatorname{alt}(p)}} = \frac{R(t, u)}{K(t, u)}.$$

Finally, we compare both expressions obtained for $W(t, u)$. We now have

$$W(t, u) = \frac{R(t, u)}{(1 - tP(u))R(t, u) + t^\ell u^{\operatorname{alt}(p)}} = \frac{R(t, u)}{K(t, u)} = \frac{(1, 0, \dots, 0)\operatorname{adj}(I - tA)\mathbf{1}}{\det(I - tA)}.$$

We now consider the denominators of both of these expressions. The kernel $K(t, u)$ is a polynomial in t and contains at least a term of the power ℓ. Since the length of a complement of a presuffix is always strictly less than ℓ, it is also the highest power of t. By the construction of $K(t,u)$ it is also clear that the constant term of the kernel is 1.

Looking now at $\det(I - tA)$ we can draw similar conclusions. Since A is a $\ell \times \ell$ matrix, $\det(I - tA)$ is a polynomial for t of degree ℓ. Because of the identity matrix, the constant term for this denominator is also 1. This means that the denominators are of equal power of t and have the same constant term.

Therefore, the fractions are not only equal but also identical. This completes the first part of the proof. □

Before turning to the proof of the second part of the theorem, we pause to summarize the information obtained so far. The argument above does not only yield an explicit expression for the generating function of walks avoiding a generic pattern, but it also reveals two fundamental structural properties. First, it identifies the determinant of the matrix $I - tA$ with the kernel function $K(t,u)$. Second, it shows that the autocorrelation polynomial $R(t,u)$ arises naturally as a specific minor of this matrix. These observations are sufficiently important to merit a separate formulation, which we state in the following proposition.

Proposition 4.3 (Kernel structure, [ABBG20])**.** *Let $\mathcal{S}$ be a step set and let p be a pattern with steps in $\mathcal{S}$. Then the adjacency matrix A of the associated automaton satisfies*

$$(1, 0, \dots, 0)\,\mathrm{adj}(I - tA)\mathbf{1} = R(t,u),$$

and

$$\det(I - tA) = K(t,u) = (1 - tP(u))R(t,u) + t^{|p|}u^{\mathrm{alt}(p)}.$$

Proof of Theorem 4.2, Part (2). We now turn to the proof of the formula for bridges. By definition, a bridge is a walk whose final altitude is equal to 0. Consequently, the generating function of bridges is obtained by extracting the coefficient of u^0 from the bivariate generating function of walks. That is,

$$B(t) = [u^0]\,W(t,u).$$

To perform this extraction, we use Cauchy's coefficient formula. We fix t to be a sufficiently small positive real number, so that all singularities of the integrand are well separated. Then the coefficient extraction can be written as a contour integral around the origin:

$$B(t) = \frac{1}{2\pi i}\int_{|u|=\varepsilon} \frac{W(t,u)}{u}\,\mathrm{d}u = \frac{1}{2\pi i}\int_{|u|=\varepsilon} \frac{R(t,u)}{K(t,u)\,u}\,\mathrm{d}u,$$

where $\varepsilon > 0$ is chosen small enough so that the circle $|u| = \varepsilon$ contains exactly the poles corresponding to the small roots of the kernel.

Since the integrand is a rational function of u, the residue theorem applies. The only poles inside the contour are the small roots $u_1(t), \dots, u_e(t)$ of the kernel $K(t,u)$. Therefore,

$$B(t) = \sum_{i=1}^{e} \mathrm{Res}_{u=u_i(t)}\,\frac{R(t,u)}{K(t,u)\,u}.$$

We now compute the residue at a fixed small root $u_i(t)$. Since $u_i(t)$ is a simple zero of $K(t,u)$, the pole is simple, and the residue is given by

$$\mathrm{Res}_{u=u_i(t)} \frac{R(t,u)}{K(t,u)\,u} = \left.\frac{R(t,u)}{\frac{\mathrm{d}}{\mathrm{d}u}\big(K(t,u)\,u\big)}\right|_{u=u_i(t)}.$$

We first compute the derivative in the denominator. Using the product rule, we obtain

$$\frac{\mathrm{d}}{\mathrm{d}u}\big(K(t,u)\,u\big) = K(t,u) + u\,K_u(t,u).$$

Since $u_i(t)$ is a root of the kernel, we have $K(t,u_i(t)) = 0$, and therefore

$$\left.\frac{\mathrm{d}}{\mathrm{d}u}\big(K(t,u)\,u\big)\right|_{u=u_i(t)} = u_i(t)\,K_u(t,u_i(t)).$$

Next, we compute the partial derivative $K_u(t,u)$. Recalling that

$$K(t,u) = (1 - tP(u))R(t,u) + t^{|p|}u^{\mathrm{alt}(p)},$$

we differentiate with respect to u to obtain

$$K_u(t,u) = R_u(t,u)(1 - tP(u)) - R(t,u)\,tP'(u) + t^{|p|}\,\mathrm{alt}(p)\,u^{\mathrm{alt}(p)-1}.$$

Multiplying by $u_i(t)$ and evaluating at $u = u_i(t)$ yields

$$u_i\,K_u(t,u_i) = u_i(1 - tP(u_i))R_u(t,u_i) - u_i\,tP'(u_i)R(t,u_i) + t^{|p|}\,\mathrm{alt}(p)\,u_i^{\mathrm{alt}(p)}. \tag{4.1}$$

At this stage, the presence of the term $P'(u_i)$ is inconvenient, since u_i depends on t. To eliminate this term, we use the fact that $K(t,u_i(t)) = 0$. Differentiating this identity with respect to t and applying the chain rule yields

$$0 = K_t(t,u_i) = \frac{\partial K}{\partial t}(t,u_i) + \frac{\partial K}{\partial u}(t,u_i)\,u_i'(t).$$

We compute the partial derivative with respect to t explicitly

$$\begin{aligned} K_t(t,u_i) &= (R_t(t,u_i) + R_u(t,u_i)u_i')(1 - tP(u_i)) \\ &\quad + R(t,u_i)(-tP'(u_i)u_i' - P(u_i)) \\ &\quad + |p|\,t^{|p|-1}u_i^{\mathrm{alt}(p)} \\ &\quad + t^{|p|}u_i^{\mathrm{alt}(p)-1}(\mathrm{alt}(p))u_i'. \end{aligned}$$

Rearranging this expression gives us

$$\begin{aligned} P'(u_i)u_i'tR(t,u_i) = {} & (R_t(t,u_i) + R_u(t,u_i)u_i')(1 - tP(u_i)) - R(t,u_i)P(u_i) + \\ & |p|\,t^{|p|-1}u_i^{\mathrm{alt}(p)} + t^{|p|}u_i^{\mathrm{alt}(p)-1}(\mathrm{alt}(p))u_i'. \end{aligned}$$

Plugging this into (4.1) we get the expression

$$u_i \cdot K_u(t,u_i) = -\frac{u_i}{u_i'}\Big((R_t(t,u_i) + R_u(t,u_i)u_i')(1-tP(u_i)) - R(t,u_i)P(u_i) + \\ |p|\, t^{|p|-1}u_i^{\mathrm{alt}(p)} + t^{|p|}u_i^{\mathrm{alt}(p)-1}(\mathrm{alt}(p))u_i'\Big) \\ + u_i((1-tP(u_i))R_u(t,u_i)) \\ + \mathrm{alt}(p)\, t^{|p|}u_i^{\mathrm{alt}(p)}.$$

We can reduce this to

$$u_i \cdot K_u(t,u_i) = -\frac{u_i}{u_i'}\Big(R_t(t,u_i)(1-tP(u_i)) - R(t,u_i)P(u_i) + |p|\, t^{|p|-1}u_i^{\mathrm{alt}(p)}\Big).$$

We recognize the content of the parentheses on the right-hand side as $K_t(t,u_i)$ and we therefore obtain

$$\mathrm{Res}_{u=u_i(t)}\frac{R(t,u)}{K(t,u)\,u} = -\frac{u_i'(t)}{u_i(t)}\frac{R(t,u_i(t))}{K_t(t,u_i(t))}.$$

Summing over all small roots completes the proof

$$B(t) = -\sum_{i=1}^{e}\frac{u_i'(t)}{u_i(t)}\frac{R(t,u_i(t))}{K_t(t,u_i(t))}.$$

□

The preceding proof shows how the kernel equation not only determines the generating function of walks, but also governs the analytic structure required to extract refined subclasses such as bridges. The key step is the reduction of coefficient extraction to a residue computation at the small roots of the kernel. In the next section, we apply similar ideas to the study of meanders. In that setting, the nonnegativity constraint leads to additional boundary terms, and the kernel method takes on its characteristic vectorial form.

4.2 Meanders and Excursions

Theorem 4.4 (Generating function of meanders avoiding a generic pattern, [ABBG20] Theorem 3.2)**.** *The bivariate generating function of meanders avoiding the pattern p is*

$$M(t,u) = \frac{G(t,u)}{u^e K(t,u)}\prod_{i=1}^{e}\big(u-u_i(t)\big), \tag{4.2}$$

where $u_1(t),\dots,u_e(t)$ are the small roots of the kernel equation $K(t,u)=0$, and $G(t,u)$ is a polynomial in u (and a formal power series in t) which will be characterized during the proof.

Proof of Theorem 4.4. We follow the vectorial kernel method as developed in [ABBG20]. As in the proof for walks, we decompose the generating function of meanders according to the terminal state of the automaton.

Let $M_i(t,u)$ denote the generating function of meanders that end in state X_i, for $i = 1,\dots,\ell$. Collecting these components into a row vector yields

$$(M_1,\dots,M_\ell).$$

The construction of meanders proceeds as follows. We start with the empty walk, which contributes the vector $(1,0,\dots,0)$. From any existing meander, we may append one step according to the transition matrix A, contributing the term $t(M_1,\dots,M_\ell)A$. However, unlike in the case of unrestricted walks, the nonnegativity constraint may be violated by such a transition. Any contribution that produces a negative altitude must therefore be removed.

This correction is achieved by subtracting the part of the expansion that carries strictly negative powers of u. We thus obtain the functional equation

$$(M_1,\dots,M_\ell) = (1,0,\dots,0) + t(M_1,\dots,M_\ell)A - t\,\{u^{<0}\}\big((M_1,\dots,M_\ell)A\big), \tag{4.3}$$

where $\{u^{<0}\}$ denotes the operator that extracts all terms with negative powers of u.

Rearranging terms yields

$$(M_1,\dots,M_\ell)(I-tA) = (1,0,\dots,0) - t\,\{u^{<0}\}\big((M_1,\dots,M_\ell)A\big). \tag{4.4}$$

We now introduce the vector

$$\mathbf{F}(t,u) = (F_1(t,u),\dots,F_\ell(t,u)) := (1,0,\dots,0) - t\,\{u^{<0}\}\big((M_1,\dots,M_\ell)A\big), \tag{4.5}$$

defined by the right-hand side of this equation. Each component $F_i(t,u)$ is a polynomial in t and a Laurent polynomial in u. While the explicit form of $\mathbf{F}$ is not known at this stage, its combinatorial meaning is clear: it encodes precisely those transitions that would violate the nonnegativity constraint.

With this notation of the right-hand side (4.4) becomes

$$(M_1,\dots,M_\ell)(I-tA) = \mathbf{F}. \tag{4.6}$$

We now multiply from the right by $(I-tA)^{-1}$ and subsequently by the column vector $\mathbf{1}$ in order to sum over all terminal states. This yields

$$M(t,u) = (M_1,\dots,M_\ell)\mathbf{1} = \mathbf{F}(I-tA)^{-1}\mathbf{1}. \tag{4.7}$$

Using the adjugate formula for the inverse matrix, we have

$$(I-tA)^{-1} = \frac{\operatorname{adj}(I-tA)}{\det(I-tA)}.$$

Recalling that $\det(I-tA) = K(t,u)$ is the kernel polynomial, we define

$$\mathbf{v}(t,u) := \operatorname{adj}(I-tA)\mathbf{1}.$$

With this notation (4.7) becomes

$$M(t,u) = \frac{\mathbf{F}(t,u)\,\mathbf{v}(t,u)}{K(t,u)}.$$

We now exploit the structure of the kernel. Let $u_i(t)$ be a small root of the kernel equation $K(t,u_i) = 0$. Since $K(t,u) = \det(I - tA)$, the matrix $(I - tA)\big|_{u=u_i(t)}$ is singular. In particular, its determinant vanishes, so its rank is strictly smaller than ℓ.

By Theorem 2.2, for any square matrix M,

$$M \cdot \operatorname{adj}(M) = \det(M)\, I.$$

Applying this identity to $M = (I - tA)\big|_{u=u_i(t)}$ yields

$$(I - tA)\big|_{u=u_i} \operatorname{adj}\big((I - tA)\big|_{u=u_i}\big) = \det((I - tA)|_{u=u_i})I = 0.$$

Multiplying from the right by $\mathbf{1}$ gives

$$(I - tA)\big|_{u=u_i} \mathbf{v}\big|_{u=u_i} = \mathbf{0}.$$

Thus $\mathbf{v}\big|_{u=u_i}$ belongs to the kernel of the matrix $(I - tA)\big|_{u=u_i}$. For generic step sets and patterns, this kernel is one-dimensional, which implies that $\mathbf{v}\big|_{u=u_i}$ spans the nullspace. This fact ensures that any vector annihilated by $(I - tA)\big|_{u=u_i}$ must be proportional to $\mathbf{v}\big|_{u=u_i}$.

Returning to equation (4.6), we multiply both sides from the right by $\mathbf{v}\big|_{u=u_i}$. The left-hand side vanishes, and we obtain

$$\mathbf{F}(t,u_i)\,\mathbf{v}(t,u_i) = 0.$$

This identity holds for every small root $u_i(t)$ of the kernel.

We now define

$$\phi(t,u) := u^e\,\mathbf{F}(t,u)\,\mathbf{v}(t,u),$$

where $e = \max\{c, -\operatorname{alt}(p)\}$. By construction, $\phi(t,u)$ is a Laurent polynomial in u and a formal power series in t.

On the other hand, from the representation

$$\phi(t,u) = u^e\, M(t,u)\, K(t,u),$$

We have already seen in (3.5) that $u^e \cdot K(t,u)$ is a polynomial in u. And since $M(t,u)$ does not have any negative powers of u since it is a power series $\phi(t,u)$ has no negative powers in u which makes it a polynomial.

Since $\phi(t,u)$ is a polynomial that vanishes at each small root $u_i(t)$ of the kernel, it follows that

$$\phi(t,u) = G(t,u) \prod_{i=1}^{e} (u - u_i(t)),$$

for some polynomial $G(t,u)$ in u whose coefficients are formal power series in t.

Dividing by $u^e K(t,u)$ yields the desired expression

$$M(t,u) = \frac{G(t,u)}{u^e K(t,u)} \prod_{i=1}^{e} (u - u_i(t)),$$

which completes the proof. □

Theorem 4.5. *(Generating function of excursions, generic pattern, [ABBG20, Theorem 3.2]) Let $M(t,u)$ be the bivariate generating function of meanders avoiding a fixed pattern p, where the variable t marks the length and u marks the final altitude. Let*

$$M(t,u) = \frac{G(t,u)}{u^e K(t,u)} \prod_{i=1}^{e} \big(u - u_i(t)\big),$$

be the representation obtained in (4.2)*, where $u_1(t), \ldots, u_e(t)$ are the small roots of $K(t,u) = 0$, and $G(t,u)$ is a polynomial in u (and a formal power series in t).*

Then the generating function of excursions avoiding p, that is,

$$E(t) = \sum_{n \geq 0} e_n t^n,$$

is given by the specialization

$$E(t) = [u^0]\, M(t,u) = \lim_{u \to 0} M(t,u).$$

Proof. Recall that in the bivariate generating function $M(t,u)$, the variable u marks the final altitude of a meander. Thus, a path of length n ending at altitude k contributes to the coefficient of $t^n u^k$. In particular, $M(t,u)$ encodes all meanders, allowing arbitrary non-negative final altitudes.

Excursions form a subclass of meanders, namely those paths that return to altitude 0 at their endpoint. Hence, in terms of the generating function, excursions are precisely those contributions corresponding to final altitude 0. Extracting these paths amounts to taking the coefficient of u^0 in $M(t,u)$, that is,

$$E(t) = [u^0]\, M(t,u).$$

Equivalently, since $M(t,u)$ is expressed as a rational function in u whose denominator vanishes at $u = 0$ with multiplicity e, the coefficient of u^0 is obtained by evaluating the expression at $u \to 0$, provided the numerator cancels the singularity. This cancellation is ensured by the construction of $G(t,u)$, which encodes the admissible combinations of the small roots $u_i(t)$.

From a combinatorial perspective, this specialization reflects the distinction between meanders and excursions: while meanders may end at any non-negative altitude, excursions are constrained to end at altitude 0. Since u tracks the altitude, extracting the term of degree zero in u precisely isolates those paths whose endpoint is at level 0.

This yields the stated expression for $E(t)$. □

We have now established the general construction of generating functions for generic patterns in meanders, which is the situation most frequently encountered in applications. The proof illustrates the overall strategy and the sequence of steps required to obtain explicit generating functions via the vectorial kernel method.

Among these steps, the most delicate part is the determination of the vector **F**. This is the only component of the construction that is not fully algorithmic, as it depends on the interaction between the automaton and the boundary conditions. In special situations, however, the structure of $\mathbf{F}$ and hence of the auxiliary function $G(t,u)$, can be predicted directly from combinatorial considerations.

In this section, we focus on one such situation. A key assumption is that the forbidden pattern is a *quasi-meander*. When we cut of the last step of a quasi-meander it becomes a meander. So it satisfies the same non-negativity condition as a meander, except for the final step, where there is no constraint.

The proof in this case relies on the fact that the pattern itself is almost a meander. As a consequence, no forbidden transition can arise from intermediate states of the automaton; the only forbidden transition corresponds to stepping into negative height in the initial state. Although a quasi-meander may become negative at its last step, this situation never occurs in paths that avoid the pattern, and therefore does not need to be considered.

4.3 Special Subcases

In this subsection, we consider a special subcase of forbidden patterns, namely quasi-meanders. This class of patterns exhibits additional structural properties that significantly simplify the application of the vectorial kernel method. In particular, the interaction between the automaton and the boundary conditions becomes more transparent, allowing for a more direct determination of the generating function.

Related variants have been studied in the literature. For instance, [ABBG20] also investigate reverse meanders. However, this case is not included in the present work, as it introduces additional technical complications that go beyond the scope of our approach.

The quasi-meander setting illustrates an important principle: once certain structural characteristics of the forbidden pattern are known, the general framework developed earlier can be specialized to yield explicit and uniform formulas for the associated generating functions.

Theorem 4.6 (Generating function of meanders, quasimeander pattern subcase, [ABBG20] Theorem 3.4)**.** *Let p be a quasimeander.*

1. *The bivariate generating function of meanders avoiding the pattern p is*

$$M(t,u) = \frac{R(t,u)}{u^c K(t,u)} \prod_{i=1}^{c} \big(u - u_i(t)\big), \tag{4.8}$$

where $u_1(t), \ldots, u_c(t)$ are the small roots of $K(t,u) = 0$. If one does not keep track of the final altitude, this specializes to

$$M(t) = M(t,1) = \frac{R(t,1)}{K(t,1)} \prod_{i=1}^{c} \big(1 - u_i(t)\big). \tag{4.9}$$

2. *The generating function of excursions avoiding the pattern p is*

$$E(t) = M(t,0) = \begin{cases} \dfrac{(-1)^c}{-t} \displaystyle\prod_{i=1}^{c} u_i(t), & \textit{if } \operatorname{alt}(p) > -c, \\ \dfrac{(-1)^c}{t^{|p|} - t} \displaystyle\prod_{i=1}^{c} u_i(t), & \textit{if } \operatorname{alt}(p) = -c. \end{cases} \tag{4.10}$$

Proof. We begin by recalling the defining property of a quasimeander. By definition, a quasimeander is a path that remains non-negative at all intermediate steps, with the possible exception of its final step. Since in our encoding the final step is not represented in the state space, this exception does not affect the functional equations. Consequently, all prefixes of the pattern p have non-negative altitude.

This observation has an important consequence for the autocorrelation polynomial $R(t, u)$. Since all prefixes have non-negative altitude, all presuffixes also have non-negative altitude. In particular, no negative powers of u occur in $R(t, u)$.

Moreover, the only way the full pattern p can attain a negative final altitude is through its last step. The maximal possible drop is therefore given by the largest negative step $-c$, which implies

$$\text{alt}(p) \geq -c.$$

We now derive the functional equation. As in the general framework, we obtain

$$(M_1, \ldots, M_\ell)(I - tA) = (1, 0, \ldots, 0) - t\{u^{<0}\}((M_1, \ldots, M_\ell)A).$$

For quasimeanders, the right-hand side simplifies significantly. Indeed, the non-negativity condition ensures that no transitions between non-initial states can produce negative altitude contributions. Thus, only transitions into the initial state may contribute to the negative part. This yields

$$\phi(t, u) = u^c \,(F_1, 0, \ldots, 0)\, \vec{v}(t, u),$$

where $\vec{v}(t, u) = \text{adj}(I - tA)\mathbf{1}$, as given by Proposition 4.3.

Taking the first component, we obtain

$$(1, 0, \ldots, 0)\ \text{adj}(I - tA)\mathbf{1} = R(t, u),$$

so that

$$\phi(t, u) = u^c\, R(t, u)\, F_1(t, u).$$

It remains to analyse $F_1(t, u)$, which encodes the contributions that violate the non-negativity condition at the initial state. By construction,

$$F_1(t, u) = 1 - t\,\{u^{<0}\} \qquad \text{(transitions to the initial state)}.$$

Since the largest possible negative step is $-c$, the lowest possible power of u that can appear is u^{-c}. Therefore,

$$u^c F_1(t, u) = u^c - t \cdot Q(t, u),$$

where $Q(t, u)$ is a polynomial in u of degree strictly less than c.

It follows that $u^c F_1(t, u)$ is a polynomial of degree c, and we may factor it as

$$u^c F_1(t, u) = \prod_{i=1}^{c} (u - u_i(t)),$$

where the $u_i(t)$ are precisely the small roots of the kernel equation $K(t,u) = 0$.

Substituting this factorization into the general solution

$$M(t,u) = \frac{\phi(t,u)}{u^c K(t,u)},$$

we obtain

$$M(t,u) = \frac{R(t,u)}{u^c K(t,u)} \prod_{i=1}^{c} (u - u_i(t)),$$

which proves (4.8). The specialization $u = 1$ immediately yields (4.9). This completes the first part of the proof. Now we consider excursions for the second part.

For this, we use the relation

$$E(t) = M(t,0).$$

From the expression obtained above,

$$M(t,u) = \frac{R(t,u)}{u^c K(t,u)} \prod_{i=1}^{c} (u - u_i(t)).$$

We analyze the limit as $u \to 0$. First, observe that

$$\prod_{i=1}^{c} (u - u_i(t))\Big|_{u=0} = (-1)^c \prod_{i=1}^{c} u_i(t).$$

It remains to evaluate the prefactor

$$\frac{R(t,u)}{u^c K(t,u)} \quad \text{at } u = 0.$$

Recall that

$$K(t,u) = (1 - tP(u))R(t,u) + t^{|p|} u^{\mathrm{alt}(p)}.$$

We distinguish two cases.

Case 1: $\mathrm{alt}(p) > -c$**.**

In this case, the term $u^{\mathrm{alt}(p)}$ vanishes after multiplication by u^c and evaluation at $u = 0$. Hence,

$$u^c K(t,u)\big|_{u=0} = -t \cdot R(t,0),$$

since $u^c P(u)$ has constant term 1.

Canceling $R(t,0)$, we obtain

$$\frac{R(t,u)}{u^c K(t,u)}\Big|_{u=0} = \frac{-1}{t}.$$

Thus,

$$E(t) = \frac{(-1)^c}{-t} \prod_{i=1}^{c} u_i(t).$$

Case 2: $\mathrm{alt}(p) = -c$**.**

This case can only be achieved if the last step of the pattern is $-c$ and the altitude before the last step was 0. This is actually the lowest possible altitude for a quasimeander. This is also why we do not need to consider p $< -c$.

We claim that this forces

$$R(t,u) = 1.$$

Indeed, suppose a non-trivial presuffix exists. Then it must be both a prefix and a suffix of p. Since all prefixes (except possibly the full pattern) are meanders, they have non-negative altitude. However, a suffix ending with a step $-c$ necessarily attains a strictly negative altitude. This contradicts the possibility of matching prefix and suffix. Hence, no non-trivial presuffix exists and $R(t,u) = 1$.

We now obtain

$$u^c K(t,u)\big|_{u=0} = t^{|p|} - t,$$

and therefore

$$\frac{R(t,u)}{u^c K(t,u)}\bigg|_{u=0} = \frac{1}{t^{|p|} - t}.$$

Combining with the product term yields

$$E(t) = \frac{(-1)^c}{t^{|p|} - t} \prod_{i=1}^{c} u_i(t).$$

This completes the second part of the proof. □

In the following example, we illustrate why the distinction between the general case and the quasi-meander case is essential. We construct two forbidden patterns that share the same basic characteristics—namely length, total altitude, and step set, yet behave fundamentally differently with respect to the non-negativity constraint.

More precisely, one of the patterns is a meander and hence also a quasi-meander, while the other is not. This example demonstrates that the additional structural property of staying above the horizontal axis is not merely technical, but plays a crucial role in the correctness of the kernel method. It highlights where the boundary condition "never go below zero" enters the construction and why it cannot be ignored.

4.3.1 Example: The Patterns DDUU and UUDD

Consider the two patterns

$$p_1 = (\mathtt{D},\mathtt{D},\mathtt{U},\mathtt{U}) = (-1,-1,+1,+1) \quad \text{and} \quad p_2 = (\mathtt{U},\mathtt{U},\mathtt{D},\mathtt{D}) = (+1,+1,-1,-1).$$

Both patterns have the same length $\ell = 4$ and total altitude

$$\mathrm{alt}(p_1) = \mathrm{alt}(p_2) = 0.$$

Nevertheless, they exhibit a different combinatorial behaviour when used as forbidden patterns.

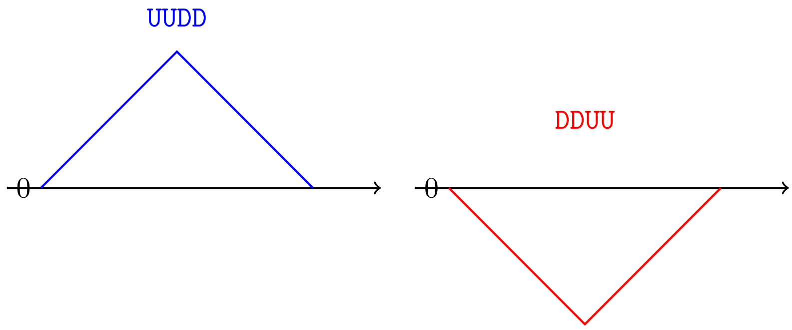


Figure 4.1: Comparison of the patterns UUDD and DDUU relative to the horizontal axis.

In the enumeration of meanders (that is, paths that stay weakly above the horizontal axis), forbidding DDUU and forbidding UUDD lead to different sets of admissible paths. For instance, for paths of length 4, both

UDUD, UUDD

avoid the pattern DDUU. However, if UUDD is forbidden, only the path UDUD remains admissible.

Thus, from a combinatorial point of view, the two patterns should give rise to different generating functions. However, when applying the vectorial kernel method, both patterns lead to the same formal expression, since only the parameters entering the kernel—namely the length ℓ and the total altitude $\mathrm{alt}(p)$—are taken into account.

This example clearly shows where the limitation arises: the method does not detect whether a pattern violates the non-negativity constraint internally. As a consequence, it fails to distinguish between patterns that stay above the axis and those that dip below it.

This observation explains why, in the case of excursions and meanders, it is essential to restrict attention to patterns that are themselves meanders. Only in this situation does the interaction with the boundary at height zero behave in a controlled way, ensuring that the kernel method produces correct enumerative results.

5 Asymptotic Behaviour

The central question of this chapter is: how many lattice paths of length n avoid a given pattern p, for large n? Rather than seeking exact formulas, as we have done in the previous chapter, we ask for the leading asymptotic behaviour of the counting sequences associated with walks, bridges, meanders, and excursions.

The answer takes the universal form

$$f_n \sim C \cdot n^{\alpha} \cdot \rho^{-n} \quad \text{as } n \to \infty,$$

where $\rho > 0$ is the radius of convergence of the generating function, α is a rational exponent determined by the type of singularity, and C is an explicit constant depending on the model [FS09, Theorem VII.8]. The nature of the singularity is what distinguishes the four path families and gives each its characteristic exponent.

The results of this chapter follow [ABBG20, Section 6]. We proceed as follows.

Not every step set and forbidden pattern leads to well-behaved generating functions. We therefore introduce the notion of a *generic model* (Section 5.1), which is a list of regularity conditions under which the asymptotic analysis can be carried out systematically. Genericity is not a deep structural theorem, rather it is a wishlist of properties that rules out degenerate cases such as rational generating functions, multiple dominant singularities, or unexpected cancellations. Many natural models arising in practice are generic.

Before analyzing each path family separately, we establish a key structural result: for any generic model, the generating functions $E(t)$ and $B(t)$ share the same dominant singularity ρ, which is a square-root branch point (Lemma 5.3). This singularity arises where the small branch $u_1(t)$ and a large branch $v_1(t)$ of the kernel equation $K(t,u) = 0$ first collide on the positive real axis.

In Theorem 5.4 we derive the asymptotics of walks. The walk generating function $W(t) = R(t,1)/K(t,1)$ has a simple pole at ρ_K, the smallest positive root of $K(t,1)$.

For the asymptotics of excursions and bridges (Theorems 5.7 and 5.8), both generating functions have a square-root singularity at ρ. By the transfer theorem [FS09, Theorem VI.3], this produces a polynomial correction of order $n^{-3/2}$ for excursions and $n^{-1/2}$ for bridges.

Meanders (Theorem 5.12) are the most complicated case. Since the endpoint is free, the average displacement per step determines which singularity of $M(t)$ dominates. We distinguish three sub cases, which yield corrections of order $n^{-1/2}$, $n^{-3/2}$, and no correction at all. The drift, that differentiates these sub cases is introduced formally in Definition 5.10.

Table 5.1 collects the results of this chapter for quick reference.

Path family	Singularity type	Exponent α	Base
Walks	simple pole	0	ρ_K
Bridges	square root	$-\frac{1}{2}$	ρ
Excursions	square root	$-\frac{3}{2}$	ρ
Meanders (zero drift)	square root	$-\frac{1}{2}$	ρ
Meanders (negative drift)	square root	$-\frac{3}{2}$	ρ
Meanders (positive drift)	simple pole	0	ρ_K

Table 5.1: Summary of asymptotic regimes for a generic model. The constant C in $f_n \sim C \cdot n^{\alpha} \cdot \rho^{-n}$ depends on the model in each case; see the individual theorems for explicit formulas.

5.1 Generic Models and Their Role in the Asymptotic Analysis

In order to derive precise asymptotic estimates for the coefficients of the generating functions associated with walks avoiding a fixed pattern p, we restrict our attention to a class of models exhibiting regular analytic behaviour. These models will be referred to as *generic*, following [ABBG20, Definition 6.3].

Definition 5.1 (Generic model). *Let $\mathcal{S} \subset \mathbb{Z}$ be a finite step set and let p be a forbidden pattern. The model $(\mathcal{S}, p)$ is called* generic *if the following properties hold:*

1. ***Algebraicity and non-rationality.*** *The generating functions $B(t)$, $M(t)$, and $E(t)$ are algebraic functions of t, but not rational.*
2. ***Unique dominant singularity.*** *Each of $B(t)$, $M(t)$, and $E(t)$ admits a unique dominant singularity $\rho > 0$, which is algebraic, not a pole.*
3. ***Polynomial numerator condition.*** *In the product representation obtained via the vectorial kernel method, the function $G(t, u)$ is polynomial in t.*
4. ***Small–large branch collision.*** *Let ρ be the smallest positive real number at which a small branch and a large branch of $K(t, u) = 0$ coincide. At $t = \rho$, no small branch with negative value meets a large branch with negative value.*
5. ***Simple critical point.*** *The smallest positive root ρ_K of $K(t, 1) = 0$ is simple:*

$$K(\rho_K, 1) = 0, \qquad K_t(\rho_K, 1) \neq 0.$$

5.1.1 Why the Generic Conditions Are Required

The conditions stated above are analysed in [ABBG20, Section 6.1]. We briefly discuss what each condition rules out and why its failure would complicate the asymptotic analysis.

(1) Algebraicity and non-rationality. Algebraicity guarantees that the generating functions admit only algebraic singularities, i.e. branch points of finite order. The exclusion of rational functions eliminates the case where the dominant singularity is a pole. Rational generating functions are in fact well understood by classical methods and do not require the kernel approach developed here. Genericity simply excludes them from the present treatment.

(2) Unique dominant singularity. The existence of a unique dominant singularity ensures that the exponential growth rate of the coefficients is determined by a single value ρ. If several singularities of equal modulus were present, periodic oscillations could appear in the asymptotics, complicating the leading-order formula.

A further degenerate case occurs when the dominant singularity is a pole rather than a branch point; this situation is already handled by the walk asymptotics (Theorem 5.4) and does not require the full machinery of singularity analysis for algebraic functions.

(3) Polynomial numerator condition. The factor $G(t,u)$ in the meander generating function is required to be a polynomial in t. This ensures that $G(t,u)$ introduces no additional singularities beyond those already present in the kernel.

(4) Small–large branch collision. Property 4 excludes the case where a small negative branch meets a large negative branch at $t=\rho$. Such a collision would introduce a new square-root singularity of a different type at ρ, changing the nature of the asymptotic problem entirely.

Note that two small branches may meet at ρ without causing problems, since all small roots appear together in $Y(t) = u_2(t)\cdots u_e(t)$, which remains analytic at ρ. Only the collision of a small branch with a large branch creates a new singularity.

The authors of [ABBG20] conjecture that this property holds in general, but a complete proof remains open.

(5) Simple critical point. The simplicity condition $K_t(\rho_K,1) \neq 0$ ensures that ρ_K is a genuine simple zero of $K(t,1)$, not a higher-order one. This guarantees that the meander generating function has a true square-root singularity at ρ_K in the zero-drift case, as opposed to a higher-order branch point that would alter the exponent α. When this condition fails, the singularity at ρ_K is of higher order, leading to a different asymptotic exponent. Such cases can in principle be analysed by the same methods but require more involved calculations depending on the multiplicity of the zero.

The class of generic models includes a large family of well-behaved step sets and forbidden patterns. Non-generic cases arise when singularities coincide, when algebraicity fails, or when the drift structure is degenerate.

5.2 Dominant Singularity

With genericity in place, we can now identify the dominant singularity that governs the asymptotics of excursions and bridges. Before we start with the next theorem we want to discuss the possibility of small branch collisions.

Remark 5.2. *Why small-small collisions do not create singularities. The excursion generating function is, up to an analytic prefactor, the product of all small roots:*

$$E(t) \;\sim\; (-1)^e \frac{G(t,0)}{D(t)}\, u_1(t) \cdots u_e(t).$$

A singularity of $E(t)$ must therefore come from a singularity of this product. Now, multiplying the kernel by u^e to clear negative powers gives the polynomial $\widetilde{K}(t,u) = u^e K(t,u)$, whose roots in u are exactly the small and large roots of $K(t,u)$. By Vieta's formulas, the product of all *roots of $\widetilde{K}$ in u equals the ratio of its constant term to its leading coefficient (up to sign), and both of these are rational functions of t alone. Hence the product of all roots — small and large together — is a rational function of t, analytic away from its poles.*

Suppose now that at some point t_0, only small roots collide while all large roots remain analytic. The product of the large roots is then analytic at t_0. Since the product of all roots is rational (hence analytic at t_0, assuming t_0 is not a pole), the product of the small roots must also be analytic at t_0. Therefore a collision among small roots alone cannot produce a singularity in $E(t)$, and ρ is correctly identified as the smallest t at which a small root meets a large root. This is the content of [ABBG20, Proposition 4.1].

Lemma 5.3 (Location and nature of the dominant singularity)**.** *For any generic model, the dominant singularity of $B(t)$ and $E(t)$ is ρ, the smallest real positive number such that a small branch meets a large branch at $t = \rho$. The corresponding branches, denoted u_1 and v_1, have a square-root type branching point at $t = \rho$.*

Proof. We establish the following facts in sequence: that the dominant singularity of $E(t)$ is a simple square-root with exponent $\alpha = -3/2$, that this singularity occurs at a small-large branch collision point ρ with $0 < \rho < 1$, and finally that $B(t)$ shares the same ρ.

Lattice paths avoiding a fixed pattern are generated by a pushdown automaton constructed from the pattern-tracking automaton of Chapter 3. Any language generated by a pushdown automaton is context-free [FS09, Chapter I], and by a theorem of Chomsky and Schützenberger [Mat67], the generating function of any context-free language satisfies a *positive* system of polynomial equations — positive meaning all coefficients are non-negative.

Such generating functions are therefore algebraic, and their coefficients grow as $f_n \sim C \cdot \rho^{-n} \cdot n^{\alpha}$ for some rational exponent α, by the general theory of positive algebraic systems [FS09, Chapter VII] and [Mat67].

We now determine α for excursions. The system of functional equations arising from the pattern-tracking automaton has a *strongly connected* dependency graph: every state of the automaton can reach every other state along some path, ensuring that each equation in the system involves all others directly or indirectly. For strongly connected positive systems, the Drmota–Lalley–Woods theorem [FS09, Theorem VII.6] guarantees that all component generating functions share the same radius of convergence ρ, and that the dominant singularity is a simple square-root branch point. The singular expansion therefore takes the form

$$E(t) \;\sim\; A - B\sqrt{1 - t/\rho} \quad \text{as } t \to \rho,$$

for constants $A, B > 0$. By the transfer theorem for square-root singularities [FS09, Theorem VI.3]:

$$[t^n]\,E(t) \;\sim\; \frac{B}{2\sqrt{\pi}}\,n^{-3/2}\rho^{-n},$$

establishing $\alpha = -3/2$ for excursions. We note that the exponent for bridges ($\alpha = -1/2$) is derived separately in Theorem 5.8; the present lemma concerns only the location of ρ, which is shared.

We next locate ρ on the positive real axis by the following argument. Since all coefficients of $E(t)$ count lattice paths, they are non-negative integers. By Pringsheim's theorem [FS09, Theorem IV.6], the dominant singularity therefore lies on the positive real axis, so $\rho \in \mathbb{R}_{>0}$. The Pólya–Fatou–Carlson theorem [FS09, Note IV.23] states that a power series with non-negative integer coefficients and radius of convergence exactly 1 must be rational. Since $E(t)$ is algebraic but not rational by the generic assumption, we conclude $\rho \neq 1$, and hence $0 < \rho < 1$.

It remains to show that ρ is a small-large branch collision point. By the product formula for excursions (Theorem 5.7),

$$E(t) \;=\; (-1)^e\,\frac{G(t,0)\,Y(t)}{D(t)}\,u_1(t)\cdots u_e(t),$$

where the prefactor $G(t,0)/D(t)$ is analytic at ρ. The singularity of $E(t)$ must therefore come from one of the small roots $u_i(t)$.

Each $u_i(t)$ is a continuously moving root of the polynomial $K(t,u) = 0$ in u. By the implicit function theorem 2.7.3, $u_i(t)$ is analytic at any point where $K_u(t,u_i(t)) \neq 0$. It can only fail to be analytic and hence only become singular when K_u vanishes, which happens precisely when two roots of $K(\cdot,u) = 0$ collide and the branch point appears.

It remains to show that ρ is a small-large branch collision point. By the product formula, the singularity of $E(t)$ must arise from the product $u_1(t)\cdots u_e(t)$ of small roots, since the prefactor $G(t,0)/D(t)$ is analytic at ρ. A collision among small roots alone cannot create such a singularity: by Vieta's formulas applied to $\widetilde{K}(t,u) = u^e K(t,u)$, the product of all roots is a rational function of t, so if the large roots remain analytic at a collision point, the product of small roots must also remain analytic there. See Remark 5.2 for the detailed argument. By Property 4 of the generic model (Definition 5.1), the dominant singularity therefore arises from a small root $u_1(t)$ meeting a large root $v_1(t)$ at $t = \rho$, and the negative-branch condition ensures this is the only collision occurring at ρ. At this collision, $K(\rho,\tau) = 0$ and $K_u(\rho,\tau) = 0$ simultaneously, which is the condition for a square-root branch point. The singularity must consequently arise from a small root $u_1(t)$ meeting a large root $v_1(t)$, and at such a collision the kernel satisfies

$$K(\rho,\tau) = 0 \qquad \text{and} \qquad K_u(\rho,\tau) = 0,$$

which is precisely the condition for a square-root branch point, consistent with the exponent $\alpha = -3/2$ above.

Finally, we show that $B(t)$ has the same radius of convergence ρ as $E(t)$. Recall that we are counting lattice paths avoiding a fixed pattern. Every excursion is, by definition, a bridge, since it starts and ends at zero and remains non-negative throughout. In particular, excursions never attain negative altitude. Bridges, however, are allowed to visit negative altitudes, and therefore there exist bridges which are not excursions. It follows that

$$e_n \leq b_n.$$

For the reverse inequality, consider an excursion of length n. By applying cyclic shifts to its increments, one obtains at most n paths of length n, each of which starts and ends at zero, and is therefore a bridge. Conversely, every bridge arises as a cyclic shift of some excursion. However, this correspondence is not one-to-one: distinct cyclic shifts may coincide, and moreover some cyclic shifts may introduce occurrences of the forbidden pattern and are therefore not admissible in our counting. Consequently, fewer than n distinct valid bridges may be produced from a given excursion. This yields the inequality

$$b_n \leq n \cdot e_n.$$

Combining the two inequalities gives the sandwich

$$e_n \leq b_n \leq n \cdot e_n,$$

which translates directly into a comparison of exponential growth rates. Taking n-th roots throughout, we obtain

$$e_n^{1/n} \;\leq\; b_n^{1/n} \;\leq\; n^{1/n} \cdot e_n^{1/n},$$

and then passing to the limit superior as $n \to \infty$. Since $n^{1/n} \to 1$, the outer terms converge to the same limit, and we conclude that

$$\limsup_{n\to\infty} b_n^{1/n} \;=\; \limsup_{n\to\infty} e_n^{1/n}.$$

This common value determines the radius of convergence of the associated generating functions. It follows that $B(t)$ and $E(t)$ have the same radius of convergence ρ. □

Having located the dominant singularity, we begin with the simplest case: walks, where no non-negativity constraint is imposed and the generating function is simply a ratio of polynomials.

5.2.1 Asymptotics of Walks, Excursions, and Bridges

We now state and prove the asymptotic formulas for each path family in turn. The theorems below are based on [ABBG20, Theorems 6.3, 6.5]; in each case we present what we believe to be a corrected formulation and supply the detailed calculations that the original source omits.

Theorem 5.4 (Asymptotics of walks)**.** *Let ρ_K be the smallest positive root of $K(t,1)$. For any generic model, the asymptotic number of walks of length n avoiding a pattern p is*

$$W_n \;\sim\; -\frac{R(\rho_K,1)}{\rho_K\, K_t(\rho_K,1)} \cdot \rho_K^{-n}.$$

Proof. By Theorem 4.2, the walk generating function is

$$W(t) \;=\; \frac{R(t,1)}{K(t,1)},$$

obtained by evaluating the meander generating function at $u = 1$, which removes the altitude weight since walks are unconstrained in their final height.

We first identify the dominant singularity of $W(t)$ and show it is a simple pole at ρ_K. By genericity condition (5), the root ρ_K of $K(t,1)$ is simple, meaning

$$K(\rho_K, 1) = 0 \qquad \text{and} \qquad K_t(\rho_K, 1) \neq 0.$$

Hence $K(t,1)$ has a simple zero at ρ_K. It remains to verify that the numerator $R(\rho_K, 1)$ does not vanish there, so that the zero in the denominator is not canceled. The autocorrelation polynomial $R(t,1)$ has the explicit form

$$R(t,1) = \sum_{\text{overlaps of } p} t^{|\text{overlap}|},$$

a polynomial with non-negative integer coefficients and constant term 1. In particular $R(t,1) > 0$ for all real $t \geq 0$, so $R(\rho_K, 1) > 0$ and ρ_K is indeed a simple pole of $W(t)$.

We now compute the residue of $W(t)$ at ρ_K. Since $K(\rho_K, 1) = 0$, the Taylor expansion of $K(t,1)$ around $t = \rho_K$ begins

$$K(t,1) \;=\; K(\rho_K,1) + K_t(\rho_K,1)(t-\rho_K) + O\big((t-\rho_K)^2\big) \;=\; K_t(\rho_K,1)(t-\rho_K) + O\big((t-\rho_K)^2\big).$$

Substituting into $W(t) = R(t,1)/K(t,1)$ and expanding $R(t,1)$ around ρ_K gives

$$W(t) \;=\; \frac{R(\rho_K,1) + O(t-\rho_K)}{K_t(\rho_K,1)(t-\rho_K) + O\big((t-\rho_K)^2\big)} \;=\; \frac{R(\rho_K,1)}{K_t(\rho_K,1)} \cdot \frac{1}{t-\rho_K} \;+\; H(t),$$

where $H(t)$ is holomorphic in a neighbourhood of ρ_K. Rewriting the polar part using $t - \rho_K = -\rho_K(1 - t/\rho_K)$:

$$W(t) \;=\; \frac{-R(\rho_K,1)}{\rho_K\, K_t(\rho_K,1)} \cdot \frac{1}{1 - t/\rho_K} \;+\; H(t).$$

Extracting the coefficient of t^n using the geometric series $[t^n]\,\frac{1}{1-t/\rho_K} = \rho_K^{-n}$ gives

$$W_n \;=\; \frac{-R(\rho_K,1)}{\rho_K\, K_t(\rho_K,1)} \cdot \rho_K^{-n} \;+\; [t^n]\, H(t).$$

Since ρ_K is by definition the smallest positive real singularity of $W(t)$, all other singularities of $W(t)$ have strictly larger modulus $\rho' > \rho_K$. The contribution of $H(t)$ and all remaining singularities to $[t^n]W(t)$ is therefore $O(\rho'^{-n})$ for some $\rho' > \rho_K$, which is exponentially smaller than ρ_K^{-n}. Asymptotically,

$$W_n \;\sim\; \frac{-R(\rho_K,1)}{\rho_K\, K_t(\rho_K,1)} \cdot \rho_K^{-n},$$

which is the claimed formula. □

Remark 5.5. *The formula stated in [ABBG20, Theorem 6.3] has a typographical error.*

For excursions and bridges the structure is richer: the generating function has a square-root singularity rather than a pole, and the transfer theorem produces a polynomial correction factor.

5.2.2 The Square-Root Singularity and the $n^{-3/2}$ Exponent

We now explain why the exponent $n^{-3/2}$ arises universally for excursion generating functions. The argument has two steps.

Near the dominant singularity ρ, the kernel $K(t,u)$ satisfies $K(\rho,\tau)=0$ and $K_u(\rho,\tau)=0$ but $K_{uu}(\rho,\tau)\neq 0$, $K_t(\rho,\tau)\neq 0$. A Taylor expansion of K around (ρ,τ) gives

$$K(t,u) \;\approx\; K_t(\rho,\tau)(t-\rho)+\tfrac{1}{2}K_{uu}(\rho,\tau)(u-\tau)^2.$$

Setting $K(t,u)=0$ and solving for u yields

$$u_1(t) \;=\; \tau \;-\; \sqrt{\frac{-2K_t(\rho,\tau)}{K_{uu}(\rho,\tau)}}\cdot\sqrt{1-t/\rho} \;+\; O(1-t/\rho),$$

so the small root $u_1(t)$ has a square-root singularity at ρ. Since the excursion generating function is (up to analytic prefactors) a product of the small roots, it inherits this square-root behaviour:

$$E(t) \;\sim\; E(\rho)-c\sqrt{1-t/\rho} \quad \text{as } t\to\rho.$$

The expansion above is precisely the input required by Theorem 2.22, provided $E(t)$ is analytic in a Δ-domain at ρ. This analyticity is guaranteed for algebraic functions by the structure of their Riemann surfaces [FS09]. Applying the transfer theorem then gives the universal asymptotic formula

$$\boxed{E_n \;\sim\; C\cdot n^{-3/2}\cdot\rho^{-n}} \quad \text{as } n\to\infty,$$

where the constant C is determined explicitly by the kernel and the autocorrelation polynomial, as given in Theorem 5.7.

Remark 5.6. *The $n^{-3/2}$ exponent is not specific to our problem: it is the universal signature of a square-root singularity. Any algebraic generating function whose dominant singularity is a simple square-root branching point will have coefficients growing as $C\cdot n^{-3/2}\cdot\rho^{-n}$. This universality is one of the central themes of analytic combinatorics [FS09].*

Theorem 5.7 (Asymptotics of excursions)**.** *Let ρ be the dominant singularity and $\tau := u_1(\rho)$ the corresponding critical value of the small branch. For any generic model, the number E_n of excursions of length n avoiding the pattern p satisfies*

$$E_n \;\sim\; C\cdot n^{-3/2}\cdot\rho^{-n},$$

where

$$C \;=\; (-1)^{e-1}\frac{Y(\rho)\,G(\rho,0)}{D(\rho)}\sqrt{\frac{\rho\,K_t(\rho,\tau)}{2\pi\,K_{uu}(\rho,\tau)}},$$

with $Y(t)=u_2(t)\cdots u_e(t)$ and $D(t)=[u^0]\,u^eK(t,u)$, [ABBG20, Theorem 6.4].

Proof. By the vectorial kernel method, the excursion generating function (4.5) is

$$E(t) \;=\; \lim_{u\to 0} M(t,u) \;=\; \lim_{u\to 0} \frac{G(t,u)}{u^e K(t,u)} \prod_{i=1}^{e} (u - u_i(t)).$$

As $u \to 0$, each factor $(u - u_i(t))$ contributes $-u_i(t)$, and the u^e in the denominator is absorbed into $K(t,u)$ making it a polynomial in u, giving

$$E(t) \;=\; (-1)^{e-1} \frac{G(t,0)}{D(t)}\, u_1(t)\, Y(t),$$

where $D(t) = [u^0]\, u^e K(t,u)$ and $Y(t) = u_2(t) \cdots u_e(t)$.

We show that $D(\rho)$ is nonzero by examining the monomial structure of $u^e K(t,u)$. The kernel is

$$K(t,u) = (1 - tP(u))R(t,u) + t^{\ell} u^{\mathrm{alt}(p)},$$

so after multiplication by u^e the constant term in u receives contributions from two possible sources: the term $-tP(u)R(t,u)$ (first part) and the term $t^{\ell} u^{\mathrm{alt}(p)}$ (second part). Both of which are always accompanied by some factor t.

We now argue that even if both parts contribute a u^0 term, their t-exponents must differ, or else they cancel each other and another term altogether contributes the lowest power of u. So we know that $D(t)$ is either of the form $\pm t^a$ or of the form $t^a - t^b$ with $a \neq b$. In particular $D(\rho) \neq 0$ for any $0 < \rho < 1$, and the prefactor $(-1)^e G(t,0)Y(t)/D(t)$ is analytic in a neighbourhood of ρ. The singular behaviour of $E(t)$ near ρ is therefore determined entirely by $u_1(t)$. The factors $u_2(t), \ldots, u_e(t)$ collected in $Y(t)$ are analytic at ρ: by Proposition 3.5, the small roots are ordered so that $u_1(t)$ is the one whose Puiseux exponent matches the dominant singularity, while the remaining small roots have their own singularities at strictly larger values of t and are therefore analytic at ρ. The singular behaviour of $E(t)$ near ρ is thus determined entirely by $u_1(t)$.

At the dominant singularity (ρ, τ), we have

$$K(\rho,\tau) = 0, \qquad K_u(\rho,\tau) = 0, \qquad K_{uu}(\rho,\tau) \neq 0.$$

Since $K_u(\rho,\tau) = 0$, the Taylor expansion of K around (ρ,τ) has no linear term in $(u - \tau)$

$$K(t,u) \;\approx\; K_t(\rho,\tau)(t-\rho) \;+\; \tfrac{1}{2} K_{uu}(\rho,\tau)(u-\tau)^2$$

Setting $K(t,u) = 0$ and solving gives

$$(u-\tau)^2 \;\approx\; -\frac{2\,K_t(\rho,\tau)}{K_{uu}(\rho,\tau)}\,(t-\rho).$$

Substituting $t - \rho = -\rho\,(1 - t/\rho)$:

$$(u-\tau)^2 \;\approx\; \frac{2\rho\,K_t(\rho,\tau)}{K_{uu}(\rho,\tau)} \left(1 - \frac{t}{\rho}\right).$$

Taking the square root gives two branches. We choose the branch that matches $u_1(\rho) = \tau$: as t increases toward ρ from below, the small root $u_1(t)$ approaches τ from above, so the correct branch carries a minus sign in front of the square root. This gives

$$u_1(t) \;=\; \tau \;-\; \sqrt{\frac{2\rho\, K_t(\rho,\tau)}{K_{uu}(\rho,\tau)}} \cdot \sqrt{1-t/\rho} \;+\; O(1-t/\rho).$$

Since the prefactor $(-1)^e G(t,0)Y(t)/D(t)$ is analytic at ρ, we may evaluate it at $t=\rho$ to leading order while retaining t only in the singular factor $\sqrt{1-t/\rho}$, giving

$$E(t) \;\sim\; (-1)^{e-1}\,\frac{G(\rho,0)\,Y(\rho)}{D(\rho)}\left(\tau \;-\; \sqrt{\frac{2\rho\, K_t(\rho,\tau)}{K_{uu}(\rho,\tau)}} \cdot \sqrt{1-t/\rho}\right).$$

Recall that $\tau = u_1(\rho)$ is the value of the small branch at the dominant singularity; it is a finite positive constant. Multiplying the analytic prefactor by τ therefore yields the finite value $E(\rho)$, which does not affect the asymptotic behaviour of the coefficients E_n.

$$E(t) \;\sim\; E(\rho) \;-\; (-1)^{e-1}\,\frac{G(\rho,0)\,Y(\rho)}{D(\rho)}\sqrt{\frac{2\rho\, K_t(\rho,\tau)}{K_{uu}(\rho,\tau)}} \cdot \sqrt{1-t/\rho}.$$

This is equivalent to

$$E(t) \;\sim\; E(\rho) \;+\; (-1)^{e}\,\frac{G(\rho,0)\,Y(\rho)}{D(\rho)}\sqrt{\frac{2\rho\, K_t(\rho,\tau)}{K_{uu}(\rho,\tau)}} \cdot \sqrt{1-t/\rho}.$$

By the transfer theorem for square-root singularities [FS09, Theorem VI.3]

$$[t^n]\,\sqrt{1-t/\rho} \;\sim\; -\frac{1}{2\sqrt{\pi}}\,n^{-3/2}\,\rho^{-n}.$$

Extracting coefficients and noting that the two minus signs cancel,

$$E_n \;\sim\; (-1)^{e-1}\,\frac{G(\rho,0)\,Y(\rho)}{D(\rho)} \cdot \sqrt{\frac{2\rho\, K_t(\rho,\tau)}{K_{uu}(\rho,\tau)}} \cdot \frac{1}{2\sqrt{\pi}} \cdot n^{-3/2}\,\rho^{-n}.$$

Therefore

$$E_n \;\sim\; (-1)^{e-1}\,\frac{G(\rho,0)\,Y(\rho)}{D(\rho)}\sqrt{\frac{\rho\, K_t(\rho,\tau)}{2\pi\, K_{uu}(\rho,\tau)}} \cdot n^{-3/2}\,\rho^{-n},$$

which is the claimed formula with ρ in the numerator inside the square root. □

Bridges are walks that start and end at height 0, without any restriction in between. In contrast, excursions are additionally constrained to stay nonnegative throughout. Thus, bridges can be viewed as a relaxation of excursions. The endpoint constraint is the same, but the non-negativity condition is dropped. This difference is reflected in the asymptotic behaviour, where bridges exhibit a weaker polynomial correction.

Theorem 5.8 (Asymptotics of bridges)**.** *For any generic model, the asymptotic number of bridges of length n avoiding a pattern p is*

$$B_n \sim -\frac{R(\rho,\tau)}{\tau}\sqrt{\frac{1}{2\pi\rho\, K_t(\rho,\tau)\, K_{uu}(\rho,\tau)}} \cdot n^{-1/2}\, \rho^{-n}.$$

Equivalently,

$$B_n \sim -\frac{R(\rho,\tau)}{\tau\, K_t(\rho,\tau)}\sqrt{\frac{K_t(\rho,\tau)}{2\pi\rho\, K_{uu}(\rho,\tau)}} \cdot n^{-1/2}\, \rho^{-n}.$$

Proof. By Theorem 4.2 part 2, the bridge generating function is

$$B(t) = -\sum_{i=1}^{e} \frac{u_i'(t)}{u_i(t)} \cdot \frac{R(t,\, u_i(t))}{K_t(t,\, u_i(t))}.$$

Each summand corresponds to one small root $u_i(t)$ of the kernel.

By Lemma 5.3, $B(t)$ and $E(t)$ share the same dominant singularity ρ. Among the small roots, only $u_1(t)$ develops a square-root singularity at ρ, while all other branches remain analytic there. Consequently, only the term $i = 1$ contributes to the singular behaviour, and we obtain

$$B(t) \sim -\frac{R(\rho,\tau)}{K_t(\rho,\tau)} \cdot \frac{u_1'(t)}{u_1(t)} \qquad \text{as } t \to \rho,$$

where the prefactor is analytic and has been evaluated at (ρ,τ).

The singular behaviour is therefore entirely determined by the ratio $u_1'(t)/u_1(t)$. From Theorem 5.7, we know the expansion of $u_1(t)$ near ρ

$$u_1(t) = \tau - \sqrt{\frac{2\rho\, K_t(\rho,\tau)}{K_{uu}(\rho,\tau)}} \cdot \sqrt{1-t/\rho} + O(1-t/\rho).$$

We are now also interested in the expansion u_1', thus we differentiate. Thus we first note that

$$\frac{\mathrm{d}}{\mathrm{d}t}\sqrt{1-t/\rho} = -\frac{1}{2\rho}(1-t/\rho)^{-1/2}.$$

This gives

$$u_1'(t) = \frac{1}{2}\sqrt{\frac{2K_t(\rho,\tau)}{\rho\, K_{uu}(\rho,\tau)}} \cdot (1-t/\rho)^{-1/2} + O(1).$$

Since $u_1(t) \to \tau \neq 0$ as $t \to \rho$, division by $u_1(t)$ simply replaces it by τ at leading order, and we obtain

$$\frac{u_1'(t)}{u_1(t)} \sim \frac{1}{2\tau}\sqrt{\frac{2K_t(\rho,\tau)}{\rho\, K_{uu}(\rho,\tau)}} \cdot (1-t/\rho)^{-1/2}.$$

(The regular $O(1)$ term does not contribute to the dominant singularity and can be discarded.)

Substituting this into the expression for $B(t)$ yields

$$B(t) \sim -\frac{R(\rho,\tau)}{2\tau\, K_t(\rho,\tau)}\sqrt{\frac{2K_t(\rho,\tau)}{\rho\, K_{uu}(\rho,\tau)}} \cdot (1-t/\rho)^{-1/2}.$$

Rewriting the constant gives

$$B(t) \sim -\frac{R(\rho,\tau)}{\tau}\sqrt{\frac{1}{2\rho\, K_t(\rho,\tau)\, K_{uu}(\rho,\tau)}} \cdot (1-t/\rho)^{-1/2}.$$

Applying the transfer theorem [FS09, Theorem VI.3],

$$[t^n]\,(1-t/\rho)^{-1/2} \sim \frac{1}{\sqrt{\pi}}\, n^{-1/2}\,\rho^{-n},$$

we finally obtain

$$B_n \sim -\frac{R(\rho,\tau)}{\tau}\sqrt{\frac{1}{2\pi\rho\, K_t(\rho,\tau)\, K_{uu}(\rho,\tau)}} \cdot n^{-1/2}\,\rho^{-n}. \qquad \square$$

Remark 5.9. *The asymptotic formula in [ABBG20] evaluates the kernel derivatives at* $(\rho, 1)$*, implicitly assuming* $\tau = u_1(\rho) = 1$*. This assumption does not hold in general; the correct evaluation point is* (ρ,τ)*, the branch point where* u_1 *and* v_1 *meet, as derived above.*

5.3 Asymptotic Behaviour of Meanders

For meanders, the situation is more complicated. While the endpoint is unconstrained, the non-negativity condition remains in force, and the drift of the walk determines which singularity dominates.

For the conceptual understanding of the three cases that we differentiate in the proof we discuss here the concept of the drift. [ABBG20, Definition 6.6]

Definition 5.10 (Drift)**.** *For a given step set* $\mathcal{S}$ *and forbidden pattern* p*, let* $\overline{h}_n$ *denote the average final altitude of walks of length* n *on* $\mathbb{Z}$*. The* drift *of the model is defined as*

$$\delta := \lim_{n\to\infty} \frac{\overline{h}_n}{n} \in \{-\infty\} \cup \mathbb{R} \cup \{+\infty\}.$$

Accordingly, the model has

$$\delta > 0 \quad \text{(positive drift)}, \qquad \delta < 0 \quad \text{(negative drift)}, \qquad \delta = 0 \quad \text{(zero drift)}.$$

Remark 5.11. *The drift governs the long-run tendency of the walk to rise, fall, or hover. For excursions and bridges, the endpoint constraint* $w_n = 0$ *forces the walk to compensate for any drift, effectively killing its influence.*

For meanders the endpoint is free and the drift plays a decisive role, as Theorem 5.12 shows.

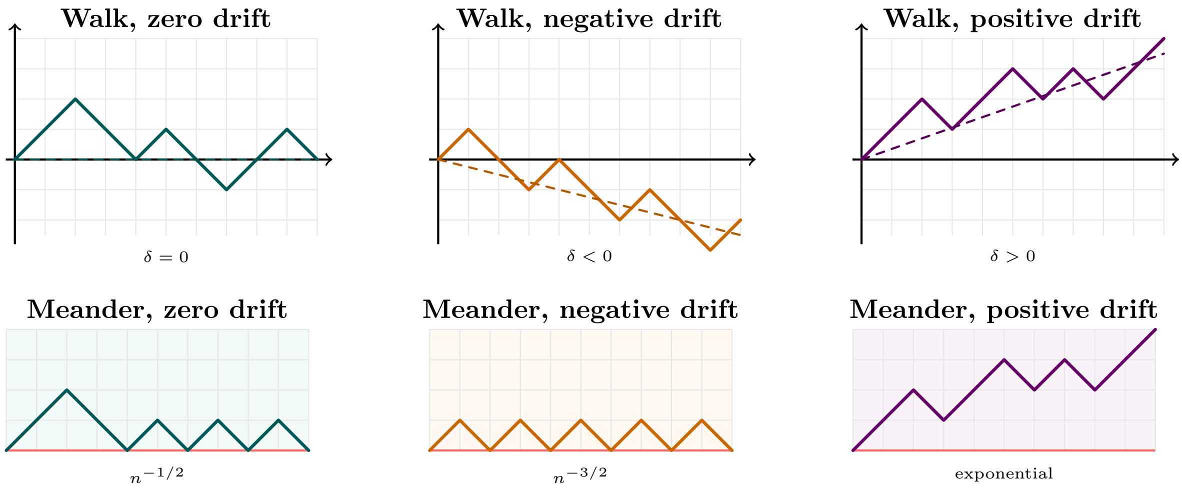


Figure 5.1: Compact comparison of drift regimes for walks (top) and meanders (bottom).

The following theorem is formulated in [ABBG20] as Theorem 6.7.

Theorem 5.12 (Asymptotics of Meanders)**.** *Assume the model is generic, and let ρ, ρ_K, and τ be defined as above in 5.4. Then one of the following three cases occurs:*

- *If $\tau = 1$ and $\rho_K = \rho$, we are in the* zero drift *case.*
- *If $\tau > 1$, $u_1(\rho_K) = 1$, and all large roots v satisfy $v(t) \neq 1$ for $\rho_K < t < \rho$, we are in the* negative drift *case.*
- *If either $\tau < 1$, or $\tau = 1$ but $\rho_K < \rho$, or $\tau > 1$ but some large root satisfies $v(\rho_K) = 1$, we are in the* positive drift *case.*

Then the asymptotics of the coefficients of

$$M(t) = \frac{(1 - u_1(t))\, Y(t)\, G(t, 1)}{K(t, 1)}, \qquad Y(t) = \prod_{i=2}^{c} (1 - u_i(t)),$$

is given by

$$M_n \sim \begin{cases} \dfrac{G(\rho, 1)\, Y(\rho) \sqrt{2}}{\sqrt{\pi \rho K_t(\rho, 1) K_{uu}(\rho, 1)}}\, n^{-1/2}\, \rho^{-n}, & \textit{(zero drift)}, \\[2ex] -\dfrac{G(\rho, 1)\, Y(\rho)}{K(\rho, 1)} \sqrt{\dfrac{\rho K_t(\rho, \tau)}{2\pi K_{uu}(\rho, \tau)}}\, n^{-3/2}\, \rho^{-n}, & \textit{(negative drift)}, \\[2ex] -\dfrac{(1 - u_1(\rho_K))\, Y(\rho_K)\, G(\rho_K, 1)}{\rho_K\, K_t(\rho_K, 1)}\, \rho_K^{-n}, & \textit{(positive drift)}. \end{cases}$$

Before turning to the proof, we provide a geometric interpretation of the quantities involved.

Consider the branches $u(t)$ of the algebraic equation

$$K(t,u) = 0,$$

viewed as functions of t. In the figures, we plot these branches in the (t,u)-plane, restricting attention to real values.

The horizontal line $u = 1$ plays a distinguished role: whenever a branch intersects this line, we obtain a solution to $K(t,1) = 0$. The smallest positive such value is denoted by ρ_K.

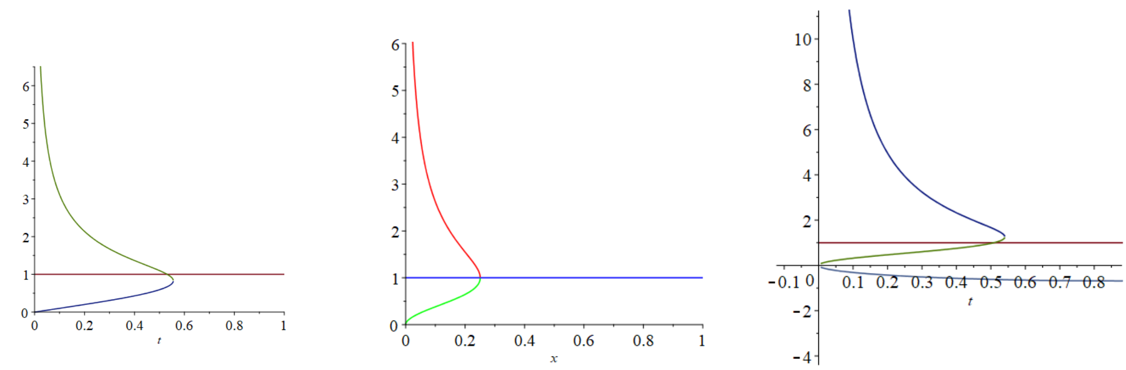
Another key feature is the collision of two branches. At such a point (t^*, u^*), two solutions of $K(t,u) = 0$ coincide, and necessarily

$$K(t^*, u^*) = 0 \quad \text{and} \quad K_u(t^*, u^*) = 0.$$

Among all such collisions, the one with smallest positive t determines the dominant singularity ρ, and the corresponding height is denoted by τ.

We emphasize that several intersections and collisions may occur, but only the leftmost one on the positive axis is relevant for the asymptotic behaviour.

We also note that the branches $u(t)$ are algebraic functions. In particular, they may leave the real domain and re-enter it, so the figures only represent their real traces.



(a) Positive drift (b) Zero drift (c) Negative drift

Figure 5.2: Visualization of branch intersections and collisions in the three drift regimes.

Proof of Theorem 5.12. Throughout, the meander generating function takes the form

$$M(t) \;=\; \frac{(1-u_1(t))\,Y(t)\,G(t,1)}{K(t,1)}, \qquad Y(t) \;:=\; \prod_{i=2}^{c}(1-u_i(t)),$$

as seen in (4.4).

Zero drift

With the zero drift the conditions are $\rho_K = \rho$ and $\tau = 1$. Both $(1 - u_1(t))$ and $K(t, 1)$ vanish at $\rho_K = \rho$. At the branch point $(\rho, 1)$ the kernel satisfies $K(\rho, 1) = 0$ and $K_u(\rho, 1) = 0$. Expanding $K(t, u_1(t)) = 0$ to second order around $(\rho, 1)$, gives

$$K(t,u) \sim K_t(\rho,1)(t-\rho) + \tfrac{1}{2}K_{uu}(\rho,1)(u-1)^2$$

so, giving

$$1 - u_1(t) \;\sim\; \sqrt{\frac{2\rho\, K_t(\rho,1)}{K_{uu}(\rho,1)}}\left(1 - \frac{t}{\rho}\right)^{1/2}.$$

Since $K(t, 1)$ has a simple zero at ρ by genericity,

$$K(t,1) \;=\; K_t(\rho,1)(t-\rho) + O\big((t-\rho)^2\big)$$

The remaining factors $Y(t)$ and $G(t, 1)$ are analytic and non-zero at ρ. Combining everything gives

$$M(t) \;\sim\; \frac{G(\rho,1)\,Y(\rho)\sqrt{\dfrac{2\rho\, K_t(\rho,1)}{K_{uu}(\rho,1)}}\left(1-\dfrac{t}{\rho}\right)^{1/2}}{-\rho\, K_t(\rho,1)\left(1-\dfrac{t}{\rho}\right)}.$$

Since $K(0, 1) = 1 > 0$ and ρ is the first zero of $K(t, 1)$, which is also simple, the function $K(t, 1)$ decreases from 1 to 0 along t, so $K_t(\rho, 1) < 0$. Writing $K_t(\rho, 1) = -|K_t(\rho, 1)|$ and absorbing the sign gives

$$= \;\frac{G(\rho,1)\,Y(\rho)}{\rho}\sqrt{\frac{2\rho}{K_t(\rho,1)\,K_{uu}(\rho,1)}}\left(1-\frac{t}{\rho}\right)^{-1/2}.$$

By the standard transfer lemma, $[t^n](1 - t/\rho)^{-1/2} \sim (\pi n)^{-1/2}\rho^{-n}$, so

$$M_n \;\sim\; \frac{G(\rho,1)\,Y(\rho)}{\rho}\sqrt{\frac{2\rho}{K_t(\rho,1)\,K_{uu}(\rho,1)}}\cdot\frac{\rho^{-n}}{\sqrt{\pi n}}.$$

Hence

$$\boxed{M_n \;\sim\; G(\rho,1)\,Y(\rho)\sqrt{\frac{2}{\pi\rho\, K_t(\rho,1)K_{uu}(\rho,1)}}\cdot n^{-1/2}\,\rho^{-n}.}$$

Negative drift

Here the following inequalities hold: $\tau > 1$, $\rho_K < \rho$, and $u_1(\rho_K) = 1$.

Since $K(t, 1)$ has a simple zero at ρ_K and $u_1(\rho_K) = 1$, the factor $(1 - u_1(t))$ vanishes at ρ_K as well. Pole and zero cancel, and $M(t)$ is holomorphic at ρ_K. Thus the dominant singularity is ρ.

Formula 5.2.2 of Theorem 5.7 gives

$$u_1(t) \;\sim\; \tau \;-\; \sqrt{\frac{2\rho\, K_t(\rho,\tau)}{K_{uu}(\rho,\tau)}}\left(1 - \frac{t}{\rho}\right)^{1/2}.$$

Since $\tau > 1$, the constant term $1-\tau \neq 0$, so $1-u_1(t)$ is analytic at ρ. Moreover $K(\rho, 1) \neq 0$ (as $\rho_K < \rho$). Expanding $M(t)$ around ρ gives

$$M(t) \;=\; M(\rho) \;+\; \frac{G(\rho,1)\,Y(\rho)}{K(\rho,1)}\sqrt{\frac{2\rho\, K_t(\rho,\tau)}{K_{uu}(\rho,\tau)}}\left(1 - \frac{t}{\rho}\right)^{1/2} + O\!\left(1 - \frac{t}{\rho}\right).$$

By the transfer lemma, giving

$$\boxed{M_n \;\sim\; -\frac{G(\rho,1)\,Y(\rho)}{K(\rho,1)}\sqrt{\frac{\rho\, K_t(\rho,\tau)}{2\pi\, K_{uu}(\rho,\tau)}} \cdot n^{-3/2}\,\rho^{-n}.}$$

Positive drift

In this case we have three different prerequisites. We will discuss them step by step. In all positive drift sub-cases ρ_K is the dominant singularity and the pole there is not cancelled.

The numerator does not vanish at ρ_K.

- One sub case of the positive drift is $\tau < 1$: then $u_1(t) \leq \tau < 1$ for all $t \leq \rho$, so $u_1(\rho_K) < 1$.
- Another sub case of the positive drift is $\tau = 1$ and $\rho_K < \rho$: the first zero of $K(t, 1)$ is caused by a large root; u_1 only reaches 1 at the later time ρ, so $u_1(\rho_K) < 1$.
- For the last sub case we have $\tau > 1$ and $v(\rho_K) = 1$: a large root caused ρ_K, and $u_1(\rho_K) \neq 1$ otherwise we would be in the negative drift case.

In all sub cases $1 - u_1(\rho_K) > 0$ and therefore the pole does not cancel. Since $K(t, 1)$ has a simple zero at ρ_K, we write

$$K(t,1) \;=\; -\rho_K\, K_t(\rho_K,1)\left(1 - \frac{t}{\rho_K}\right) + O\!\left(\left(1 - \frac{t}{\rho_K}\right)^2\right).$$

Expanding the numerator around ρ_K and dividing gives

$$M(t) \;=\; \frac{-(1 - u_1(\rho_K))\,Y(\rho_K)\,G(\rho_K,1)}{\rho_K\, K_t(\rho_K,1)} \cdot \frac{1}{1 - t/\rho_K} \;+\; H(t),$$

where $H(t)$ is holomorphic at ρ_K. Since $[t^n](1 - t/\rho_K)^{-1} = \rho_K^{-n}$, we can conclude

$$\boxed{M_n \sim \frac{-(1 - u_1(\rho_K))\, Y(\rho_K)\, G(\rho_K, 1)}{\rho_K\, K_t(\rho_K, 1)} \cdot \rho_K^{-n}.} \qquad \square$$

6 Applications to Specific Patterns

In this chapter, we illustrate the general theory developed so far by applying the vectorial kernel method to a selection of concrete patterns. The goal is to demonstrate how the abstract framework translates into explicit computations of generating functions and asymptotic formulas.

For each example, we construct the corresponding automaton, derive the transition matrix, and apply the kernel method to obtain a functional equation. From this, we extract the generating function and, where possible, determine the asymptotic behaviour of the associated counting sequence.

The sequences are known and can be found in the Online Encyclopedia of Integer Sequences (OEIS) [The26], which serves as a valuable reference for identifying and verifying combinatorial sequences. Comparing our results with entries in the OEIS highlights both the correctness of the method and its connection to a broader combinatorial context.

While the examples follow the general framework established in the previous chapters, they also demonstrate how the vectorial kernel method can be extended in a more general approach. In particular, they show that the vectorial kernel method can be applied to general automata and even counting the appearance of the forbidden pattern.

6.1 Mock Example: Zero Drift Meanders with Forbidden Pattern `UDUD`

In this example, we illustrate the general method developed in Chapter 5 on a concrete example. As outlined there, the asymptotic behavior of lattice paths depends crucially on the drift of the step set. We therefore consider three representative cases; the present example corresponds to the *zero drift* situation.

We consider lattice paths with step set $\{-2, -1, 1, 2\}$, avoiding the pattern `UDUD`. The step set is symmetric, and hence has drift zero, which we still need to verify. Since the forbidden pattern is a meander and therefore a quasimeander, we can use the theory from theorem 4.6.

$$M(t, u) = \frac{R(t, u)}{u^c K(t, u)} \prod_{i=1}^{c} (u - u_i(t))$$

or if we ignore u and set it 1 we get

$$M(t) = \frac{R(t, 1)}{K(t, 1)} \prod_{i=1}^{c} (1 - u_i(t))$$

So we need to first calculate the kernel. The formula for the kernel is given by 4.3 thus it is

$$K(t,u) = \left(1 - tu^2 - tu - \frac{t}{u} - \frac{t}{u^2}\right)(1+t^2) + t^4,$$

and the corresponding autocorrelation polynomial is

$$R(t,u) = 1 + t^2.$$

Solving the kernel equation $K(t,u) = 0$ yields four algebraic branches

$$u_1(t),\ u_2(t),\ v_1(t),\ v_2(t),$$

where u_1, u_2 are the small roots and v_1, v_2 are the large roots. The explicit expressions are algebraically involved and omitted here, but can be computed using a computer algebra system such as Maple.

The bivariate generating function of meanders is given by

$$M(t,u) = \frac{R(t,u)}{u^2 K(t,u)} \prod_{i=1}^{2} \big(u - u_i(t)\big),$$

where the product runs over the small roots. In practice, the product $\prod_{i=1}^{2}(u - u_i(t))$ can be computed symbolically in Maple via

```
expand((u - u1)*(u - u2)).
```

Substituting the resulting expression yields a closed-form representation of $M(t,u)$, which is algebraic in both variables.

We now turn to the singularity analysis. As explained in Chapter 5, dominant singularities arise from collisions between a small and a large branch of the kernel equation. Solving $u_i(t) = v_j(t)$ numerically yields a unique positive solution

$$\rho \approx 0.2509,$$

which is therefore the dominant singularity. The other equations either yield no solution, a negative solution or a complex solution. We will call the branches that collide u_1 and v_1.

Figure 6.1 illustrates this phenomenon. The plot shows that the collision occurs at height $u = 1$, i.e. $u_1(\rho) = 1 =$. This confirms that we are indeed in the zero drift case. Moreover, the figure indicates that no earlier intersection occurs, so ρ is also the smallest positive solution of $K(t,1) = 0$. Thus $\tau = 1$ is indeed true.

We now verify the genericity conditions required for the asymptotic results stated in Chapter 5.

- The generating function $M(t,u)$ is algebraic, as it is constructed from the algebraic solutions of the kernel equation.
- There exists a unique dominant singularity ρ, which is algebraic and arises from a collision of branches, hence it is not a pole.

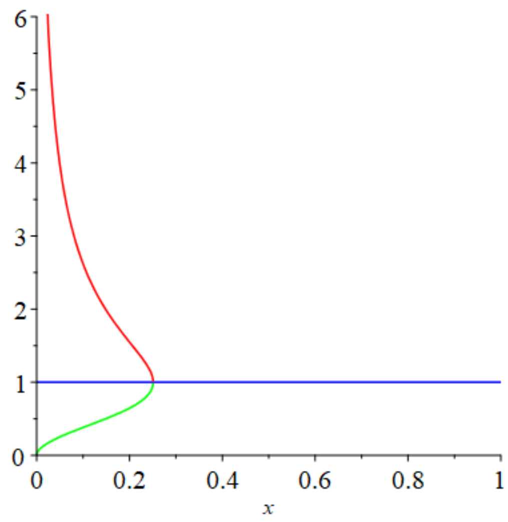


Figure 6.1: Visualization of the root collision at $t = \rho$.

- The function $R(t,u)$ is a polynomial in u, which is clearly satisfied since $R(t,u) = 1+t^2$.
- No negative branch interferes at the dominant singularity. Indeed, the negative branches only meet in the complex plane and therefore do not affect the real dominant singularity.
- Finally, we verify that the smallest positive solution τ of $K(t,1) = 0$ is simple. Substituting $u = 1$ yields

$$K(t,1) = (1-4t) \cdot (t^2+1) + t^4.$$

Solving $K(t,1) = 0$ gives

$$0.2509...,$$

which is ρ as expected and one checks that

$$\frac{\partial}{\partial t} K(t,1)\Big|_{t=\tau} = -4.1905 \neq 0,$$

so the root is simple.

This is therefore a generic model and the general asymptotic result from Chapter 5 applies. In particular, the number M_n of meanders satisfies

$$M_n \sim G(\rho,1)\,Y(\rho)\sqrt{\frac{2}{\pi\rho K_t(\rho,1)K_{uu}(\rho,1)}}\; n^{-1/2}\rho^{-n},$$

where

$$Y(t) = \prod_{i=2}^{c}(1-u_i(t)).$$

Evaluating the prefactor numerically yields

$$C \approx 0.6999,$$

such that

$$M_n \sim C \cdot \rho^{-n} n^{-1/2}.$$

This example demonstrates concretely how the kernel method, combined with singularity analysis, yields explicit algebraic expressions for generating functions and precise asymptotic estimates. This example was constructed to illustrate the zero drift case. The author is not aware of this specific model appearing in the literature; the asymptotic constant $C \approx 0.6999$ is believed to be new.

6.2 Mock Example: Positive drift with Forbidden Pattern `UdUd`

We consider meanders with step set $S = \{-1, +2\}$ and forbidden pattern $p = (+2, -1, +2, -1)$ of length $|p| = 4$ and altitude $\mathrm{alt}(p) = 2 - 1 + 2 - 1 = 2$. Here `U` denotes the step $+2$ and `d` the step -1; the capitalisation reflects the relative magnitude of the steps. Since the only negative step has magnitude $c = 1$, the kernel has exactly one small root. We note that the forbidden pattern is a quasimeander and thus the formula from 4.6 applies. First we calculate the autocorrelation polynomial. The pattern p has a unique proper self-overlap: its length-2 prefix $(+2, -1)$ equals its length-2 suffix $(+2, -1)$, with altitude 1. Hence

$$R(t, u) \;=\; 1 + t^2 u.$$

Next we look at the kernel and the step polynomial. The step polynomial is $P(u) = u^{-1} + u^2$. Substituting into the general formula gives

$$K(t, u) \;=\; \big(1 - t\,P(u)\big)\,R(t, u) + t^4 u^2 \;=\; \Big(1 - t\big(u^{-1} + u^2\big)\Big)(1 + t^2 u) + t^4 u^2.$$

Multiplying through by u yields the polynomial

$$K(t, u) \cdot u \;=\; -t^3 u^4 + (t^4 - t)\,u^3 + t^2\,u^2 + (1 - t^3)\,u - t,$$

a degree-4 polynomial in u with one small root $u_1(t)$.

To apply the Theorem 5.12 we verify the fife genericity conditions. The meander generating function is algebraic due to u_1, so its singularities are not poles. The function $R(t, u) = 1 + t^2 u$ is a polynomial in t, hence has no negative real singularity at ρ. Evaluating at $u = 1$ gives

$$K(t, 1) \;=\; (1 - 2t)(1 + t^2) + t^4 \;=\; t^4 - 2t^3 + t^2 - 2t + 1,$$

which has a simple zero at ρ_K. The unique algebraic dominant singularity stems from a collision between a small branch and a large branch. The genericity conditions are therefore satisfied.

The smallest positive real zero of $K(t, 1)$ is

$$\rho_K \;\approx\; 0.531.$$

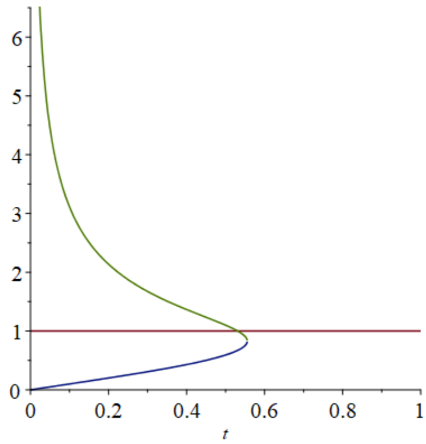


Figure 6.2: Visualization of the root collision at $t = \rho$.

Maple computes the four roots $u_1(t), v_1(t), v_2(t), v_3(t)$ of the kernel explicitly. As illustrated in Figure 6.2, the small root $u_1(t)$ and one large root collide at

$$\rho \;\approx\; 0.556, \qquad \tau \;=\; u_1(\rho) \;\approx\; 0.829.$$

Since $\tau < 1$, we are indeed in the positive drift case. The dominant singularity of $M(t)$ is the simple pole at $\rho_K < \rho$.

For $c = 1$ the meander generating function is

$$M(t) \;=\; \frac{(1 - u_1(t))\, R(t,1)}{K(t,1)}.$$

Now we have everything prepared to calculate the asymptotic constant. The positive drift asymptotic formula then yields

$$M_n \;\sim\; \frac{-(1 - u_1(\rho_K))\, R(\rho_K, 1)}{\rho_K\, K_t(\rho_K, 1)} \cdot \rho_K^{-n} \;\approx\; 0.390 \cdot (0.531...)^{-n}.$$

6.3 Mock Example: Negative Drift with Forbidden Pattern `uuDu`

We consider lattice paths with step set $S = \{-2, +1\}$ avoiding the pattern `uuDu` $= (+1, +1, -2, +1)$. Here `u` denotes the step $+1$ and `D` the step -2; the capitalisation reflects the relative magnitude of the steps, with the dominant downward step written in uppercase. The step set has drift $\frac{-2+1}{2} = -\frac{1}{2} < 0$. Since the forbidden pattern is a quasimeander, the formula from Theorem 4.6 applies.

Autocorrelation polynomial. The pattern `uuDu` has a unique proper self-overlap: its length-1 prefix u equals its length-1 suffix u. The complement of this overlap has length 3 and height 0, giving

$$R(t,u) = 1 + t^3.$$

Kernel and step polynomial. The step polynomial is $P(u) = u^{-2} + u$. The pattern has length 4 and final height +1, so the kernel term is $t^4 u$. Substituting into the general formula gives

$$K(t,u) = \big(1 - t\,P(u)\big)\,R(t,u) + t^4\,u = \left(1 - \frac{t}{u^2} - tu\right)(1+t^3) + t^4\,u.$$

Since the largest negative power of u in $P(u)$ is u^{-2}, there are $c = 2$ small roots $u_1(t)$ and $u_2(t)$, and one large root $v_1(t)$.

Derivatives of the kernel. By direct computation:

$$\begin{aligned} K_t(t,u) &= 3t^2 - \frac{1+4t^3}{u^2} - u(1-t^3), \\ K_u(t,u) &= \frac{2t(1+t^3)}{u^3} - t(1+t^3) + t^4, \\ K_{uu}(t,u) &= -\frac{6t(1+t^3)}{u^4}. \end{aligned}$$

Meander generating function. By Theorem 4.6, the bivariate generating function is

$$M(t,u) = \frac{R(t,u)}{u^2\,K(t,u)} \prod_{i=1}^{2} (u - u_i(t)) = \frac{(1+t^3)(u-u_1(t))(u-u_2(t))}{u^2\,K(t,u)}.$$

Setting $u = 1$ gives the univariate generating function

$$M(t) = \frac{R(t,1)(1-u_1(t))(1-u_2(t))}{K(t,1)}.$$

Dominant singularity. The branches $u_1(t)$ and $v_1(t)$ collide at the unique positive real solution of $u_1(t) = v_1(t)$, giving

$$\rho \approx 0.605, \qquad \tau = u_1(\rho) \approx 1.346.$$

One verifies numerically that $K(\rho,\tau) = 0$ and $K_u(\rho,\tau) \approx 0$, confirming that (ρ,τ) is a genuine branch point of the kernel curve. Since $\tau > 1$, this is the negative drift case.

Genericity conditions. We verify the five conditions of Chapter 5.

- The generating function $M(t,u)$ is algebraic, constructed from the algebraic solutions of the kernel equation, hence not rational.
- The dominant singularity $\rho = 0.605$ is unique and algebraic, arising from a branch collision rather than a pole.
- The function $R(t,u) = 1 + t^3$ is a polynomial in t, so the third condition is satisfied.

- The only negative branch is u_2, and there is no large negative branch for it to collide with at ρ. The negative branches only meet in the complex plane.
- Substituting $u = 1$ into $K(t, 1)$ and solving gives the smallest positive root

$$\rho_K \approx 0.5357,$$

and one verifies that $K_t(\rho_K, 1) \neq 0$, so the root is simple.

Cancellation at ρ_K. Since $u_1(\rho_K) = 1$, the factor $(1 - u_1(t))$ in the numerator of $M(t)$ vanishes at ρ_K, cancelling the simple pole of $1/K(t, 1)$ there. Hence $M(t)$ is holomorphic at ρ_K, and the dominant singularity is ρ.

Asymptotic formula. We are in the negative drift case with $\tau > 1$. Here $G(\rho, 1) = R(\rho, 1) = 1 + \rho^3 \approx 1.2214$ and

$$Y(\rho) = 1 - u_2(\rho) = 1.6735,$$

since u_2 is the small branch not involved in the collision. The asymptotic formula from Chapter 5 gives

$$M_n \sim -\frac{G(\rho, 1)\, Y(\rho)}{K(\rho, 1)} \sqrt{\frac{\rho\, K_t(\rho, \tau)}{2\pi\, K_{uu}(\rho, \tau)}} \cdot n^{-3/2} \rho^{-n}.$$

Evaluating numerically yields

$$M_n \sim 4.7466 \cdot n^{-3/2} \cdot (0.605...)^{-n}.$$

6.4 A094507 Dyck paths avoiding the pattern UDUD

The sequence A094507 in the On-Line Encyclopedia of Integer Sequences [The26, A094507] is a triangle $T(n, k)$, read by rows, where $T(n, k)$ counts the number of Dyck paths of semilength n containing exactly k occurrences of the consecutive pattern UDUD. Here, as throughout, U denotes an up-step $(+1)$ and D a down-step (-1).

A systematic study of pattern occurrences in Dyck paths was carried out by Sapounakis, Tasoulas and Tsikouras [STT07], who develop a general method based on functional equations and Chebyshev polynomials of the second kind. Their framework handles arbitrary consecutive patterns and, in particular, yields a bivariate generating function for UDUD whose specialisation to $k = 0$ counts Dyck paths *avoiding* the pattern entirely. This first column of the triangle, which corresponds to the sequence A078481 [The26, A078481], is where our analysis begins.

We treat this example in two stages. In the first stage we work with the standard kernel method for forbidden patterns, as developed in the preceding sections, to derive a closed-form generating function for paths avoiding UDUD and to extract the asymptotic behaviour of these counts. In the second stage we pass to the full triangle. Tracking the number of pattern occurrences introduces an additional variable v, and the transition structure of the problem is no longer captured by a single kernel equation. To handle this we model the problem as a finite automaton whose states record the longest suffix of the current path that matches a prefix of UDUD, augmented with a fifth state that registers a completed occurrence. The generating function for the automaton's weighted path counts is then extracted via the *vectorial kernel method*: we form the transfer matrix A of the automaton, write the kernel as $K(t,u,v) = \det(I - tA)$, and use the adjugate $\mathrm{adj}(I - tA)$ together with the boundary conditions on meanders to determine all unknown generating functions simultaneously. This approach yields the trivariate generating function $M(t,u,v)$, from which $E(t,v) = M(t,0,v)$ is recovered by setting the altitude variable u to zero, giving the full bivariate generating function for the triangle A094507. Several classical sequences appear as specialisations, including the Catalan numbers (A000108) as row sums and the sequence A244236 as the column $v = 1$.

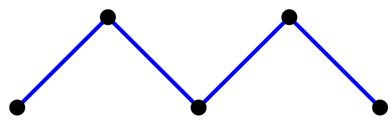
To the best of the author's knowledge, this is the first derivation of an explicit closed form for the bivariate generating function of the triangle A094507 via the vectorial kernel method. The OEIS records only an implicit quadratic equation for this generating function, whereas the trivariate generating function $M(t,u,v)$ derived here, which additionally tracks the final altitude, appears to be new. The asymptotic analysis of the avoidance case independently recovers the formula via the vectorial kernel method.

As usual, we let t mark the length of the path and u mark the altitude. The step polynomial associated with the step set $\mathcal{S} = \{+1, -1\}$ is

$$P(u) = u + u^{-1}.$$

The forbidden pattern $p =$ UDUD has length $\ell = 4$ and total altitude change

$$\mathrm{alt}(p) = +1 - 1 + 1 - 1 = 0.$$



Forbidden pattern

The effect of self-overlaps of the forbidden pattern is captured by the autocorrelation polynomial $R(t,u)$. In the present case, the pattern $UDUD$ admits a nontrivial self-overlap of length 2 and altitude 0, since its prefix $(+1,-1)$ coincides with its suffix $(+1,-1)$. Consequently, the autocorrelation polynomial is

$$R(t,u) = 1 + t^2,$$

where the term t^2 corresponds to this unique proper overlap.

Following the general construction for forbidden patterns, the kernel is defined by

$$K(t,u) = (1 - tP(u))R(t,u) + t^{\ell} u^{\mathrm{alt}(p)}.$$

Substituting the explicit expressions yields

$$K(t,u) = (1 - t(u + u^{-1}))(1 + t^2) + t^4.$$

Clearing denominators and solving the resulting equation for u gives two algebraic roots,

$$u_1(t) := \frac{(t^4 + t^2 + 1 + \sqrt{t^8 - 2t^6 - 5t^4 - 2t^2 + 1})}{(2t(t^2 + 1))}$$
$$u_2(t) := -\frac{(-t^4 - t^2 + \sqrt{t^8 - 2t^6 - 5t^4 - 2t^2 + 1} - 1)}{(2t(t^2 + 1))}$$

with

$$\lim_{t \to 0} u_2(t) = 0.$$

Thus, $u_2(t)$ is the unique *small root* of the kernel. Since the largest negative step is -1, the general theory predicts exactly one small root, in agreement with this computation.

We now have all the necessary ingredients to compute the excursion generating function.

Let $E(t)$ denote the generating function of excursions, i.e., Dyck paths that start and end at height 0. The kernel method with forbidden patterns yields the explicit formula

$$E(t) = \frac{(-1)^c}{t^\ell - t} \prod_{i=1}^{c} u_i(t),$$

where $c = 1$ is the absolute value of the minimal step and $u_1(t)$ denotes the unique small root of the kernel. In the present case, this gives

$$E(t) = \frac{t^4 + t^2 + 1 - \sqrt{t^8 - 2t^6 - 5t^4 - 2t^2 + 1}}{2t^2(t^2 + 1)}.$$

Expanding $E(t)$ as a power series in t yields

$$E(t) = 1 + t^2 + t^4 + 3t^6 + 7t^8 + 19t^{10} + 53t^{12} + 153t^{14} + \cdots.$$

Only even powers of t appear, as expected. The coefficient of t^{2n} counts Dyck paths of semilength n avoiding the pattern `UDUD`.

These values agree with the entries of the triangle A094507 for $k = 0$. Which are given at [The26, A078481] as

$$1, 1, 1, 3, 7, 19, 53, 153, 453, 1367, 4191$$

This confirms the correctness of our kernel construction and provides a kernel method derived generating function for the first column.

6.4.1 Asymptotic Analysis of Excursions

We now derive the asymptotic behaviour of the coefficients E_n, where E_n counts Dyck paths of length n avoiding the pattern `UDUD`.

Parity structure and variable substitution. Since every Dyck path has even length, $E_n = 0$ for all odd n, and the generating function satisfies $E(t) = E(-t)$. This means singularities come in symmetric pairs $\pm\rho$, so the model is not generic in the standard sense. We resolve this by substituting $s = t^2$, defining

$$\tilde{E}(s) := E(\sqrt{s}) = \frac{s^2 + s + 1 - \sqrt{s^4 - 2s^3 - 5s^2 - 2s + 1}}{2s(s+1)} = \sum_{m=0}^{\infty} E_{2m}\, s^m.$$

The coefficients of $\tilde{E}(s)$ are $\tilde{E}_m = E_{2m}$, the number of Dyck paths of length $2m$ avoiding `UDUD`, and these are nonzero at every index $m \geq 0$. The substituted system has kernel

$$\tilde{K}(s,u) = \left(1 - \sqrt{s}\left(u + \frac{1}{u}\right)\right)(1+s) + s^2,$$

autocorrelation polynomial $\tilde{R}(s,u) = 1 + s$, and a unique small root $\tilde{u}_1(s)$ with $\tilde{u}_1(0) = 0$. The substituted model is generic and the standard asymptotic theory applies directly.

Dominant singularity. The singularities of $\tilde{E}(s)$ arise where the discriminant

$$\widetilde{\Delta}(s) = s^4 - 2s^3 - 5s^2 - 2s + 1$$

vanishes. The roots are

$$s \approx 0.2820, \quad s \approx 3.5465, \quad s \approx -0.9142 \pm 0.4052\, i,$$

so $\tilde{E}(s)$ has a unique positive real dominant singularity

$$\rho_s \approx 0.2820 = \rho^2, \qquad \rho \approx 0.5310.$$

The collision height is $\tau = \tilde{u}_1(\rho_s) = 1$, confirming that the substituted model is in the zero drift case. One verifies numerically that $\tilde{K}(\rho_s, \tau) \approx 0$ and $\tilde{K}_u(\rho_s, \tau) \approx 0$, confirming a genuine branch point of the kernel curve.

Genericity conditions. We verify the five conditions of Chapter 5 for the substituted system.

- The generating function $\tilde{E}(s)$ is algebraic, involving a square root singularity, hence not rational.
- The dominant singularity $\rho_s = 0.2819716801$ is unique and algebraic, arising from a branch collision rather than a pole.
- The autocorrelation polynomial $\tilde{R}(s,u) = 1 + s$ is a polynomial in s, so the third condition is satisfied.
- There are no negative branches interfering at ρ_s. The step set $S = \{+1, -1\}$ is symmetric, so any negative branches only meet in the complex plane.
- Substituting $u = 1$ into $\tilde{K}(s, 1)$ and solving gives smallest positive root ρ_s, and one verifies that $\tilde{K}_s(\rho_s, 1) \neq 0$, so the root is simple.

Asymptotic constant via Theorem 5.7. We evaluate the ingredients of the theorem for the substituted system. With $e = 1$ and $(-1)^e = -1$:

$$\begin{aligned}
\tilde{G}(\rho_s, 0) &= \tilde{R}(\rho_s, 0) = 1 + \rho_s \approx 1.2820, \\
\tilde{Y}(\rho_s) &= 1 \quad \text{(empty product since } e = 1), \\
\tilde{D}(\rho_s) &= [u^0]\, u\, \tilde{K}(\rho_s, u) \approx -0.6807, \\
\tilde{K}_s(\rho_s, \tau) &\approx -1.9123, \\
\tilde{K}_{uu}(\rho_s, \tau) &\approx -1.3615.
\end{aligned}$$

Substituting into the formula gives

$$C_s = (-1)^e \, \frac{\tilde{G}(\rho_s, 0)\, \tilde{Y}(\rho_s)}{\tilde{D}(\rho_s)} \sqrt{\frac{\rho_s\, \tilde{K}_s(\rho_s, \tau)}{2\pi\, \tilde{K}_{uu}(\rho_s, \tau)}} \approx (-1) \cdot (-0.4728) = 0.4728...$$

Independent verification via direct singularity analysis. As a consistency check we compute C_s directly from $\tilde{E}(s)$. Near ρ_s the discriminant vanishes linearly since $\widetilde{\Delta}(\rho_s) = 0$ and $\widetilde{\Delta}'(\rho_s) = -5.207... \neq 0$. Writing $s - \rho_s = -\rho_s(1 - s/\rho_s)$:

$$\sqrt{\widetilde{\Delta}(s)} \;\approx\; \sqrt{-\widetilde{\Delta}'(\rho_s)\, \rho_s} \cdot \sqrt{1 - s/\rho_s} \;\approx\; 1.2117... \cdot \sqrt{1 - s/\rho_s}.$$

The singular part of $\tilde{E}(s)$ comes entirely from this square root. Evaluating the smooth prefactor $\frac{1}{2s(s+1)}$ at $s = \rho_s$:

$$\tilde{E}(s) \;\sim\; \tilde{E}(\rho_s) \;-\; \frac{\sqrt{-\widetilde{\Delta}'(\rho_s)\, \rho_s}}{2\rho_s(\rho_s + 1)} \cdot \sqrt{1 - s/\rho_s}.$$

By the transfer theorem [FS09, Theorem VI.3],

$$[s^m]\sqrt{1 - s/\rho_s} \;\sim\; -\frac{1}{2\sqrt{\pi}} \cdot m^{-3/2} \cdot \rho_s^{-m},$$

giving

$$E_{2m} \;\sim\; \frac{\sqrt{-\widetilde{\Delta}'(\rho_s)\, \rho_s}}{4\sqrt{\pi}\, \rho_s(\rho_s + 1)} \cdot m^{-3/2} \cdot \rho_s^{-m} \;\approx\; 0.47280 \cdot m^{-3/2} \cdot \rho_s^{-m},$$

in agreement with the constant obtained from the theorem.

Translating back to E_n. Setting $n = 2m$, so that $m = n/2$, $m^{-3/2} = 2^{3/2}\, n^{-3/2}$, and $\rho_s^{-m} = \rho^{-n}$:

$$\boxed{E_n \;\sim\; 2^{3/2}\, C_s \cdot n^{-3/2} \cdot \rho^{-n} \;\approx\; 1.3373 \cdot n^{-3/2} \cdot (0.53101...)^{-n}, \qquad \text{if n is even.}}$$

For odd n we have $E_n = 0$ exactly.

6.4.2 General case

Since pattern occurrences are allowed for all of the other columns the standard vectorial kernel method no longer suffices. To study the remaining entries $T(n,k)$, corresponding to Dyck paths containing a prescribed number of occurrences of UDUD, we now apply the vectorial kernel method in its more general form. This requires extending the underlying automaton so that occurrences of the pattern are allowed and explicitly counted. To record occurrences of the pattern, we introduce an additional marking variable v.

We introduce the following states, which encode the longest suffix of the path that matches a prefix of the pattern:

$$\varepsilon, \quad \mathtt{U}, \quad \mathtt{UD}, \quad \mathtt{UDU}, \quad \mathtt{UDUD}.$$

The first four states coincide with those used in the avoidance model. The fifth state corresponds to having just completed an occurrence of the pattern, which was previously forbidden and is now allowed but must be marked.

Let A denote the transition matrix of this automaton. Each entry of the matrix encodes the weight associated with when transitioning from the state of the row to the state of the column. Each row corresponds to one of the states above, and each transition is weighted by the corresponding step and, when appropriate, by an additional variable v marking an occurrence of the pattern.

From the empty state, a down-step leads to the empty state again, while an up-step moves the automaton to the state U. From the state U, an up-step preserves the prefix U, while a down-step leads to the state UD. From the state UD, a down-step breaks the prefix and returns to the empty state, whereas an up-step advances to UDU. From the state UDU, an up-step breaks the pattern and returns to U, while a down-step completes the pattern UDUD; this transition is therefore marked by the variable v. Finally, from the state UDUD, a down-step breaks the pattern completely and returns to the empty state, while an up-step leads to the state UDU, reflecting the overlap structure of the word UDUDUD, which contains two overlapping occurrences of $UDUD$. Encoding the transitions of the automaton as a transfer matrix, we obtain

$$A = \begin{pmatrix} \frac{1}{u} & u & 0 & 0 & 0 \\ 0 & u & \frac{1}{u} & 0 & 0 \\ \frac{1}{u} & 0 & 0 & u & 0 \\ 0 & u & 0 & 0 & \frac{v}{u} \\ \frac{1}{u} & 0 & 0 & u & 0 \end{pmatrix}.$$

Note that down steps are marked $1/u$ and upstep with u going along with the height of the step taken. Once the transition matrix A is established, the kernel is obtained as $\det(I - tA)$, and the vector

$$\mathbf{v} = \operatorname{adj}(I - tA)\mathbf{1}$$

is needed for further calculations. We now study first the matrix $\operatorname{adj}(I - tA)$

adj(I-tA)=

$$\begin{pmatrix}
(v-1)ut^3 - vt^2 - ut + 1 & tu - vt^3u & t^2 - vt^4 & t^3u & vt^4 \\
\frac{t^2}{u^2} & \frac{vt^3}{u} - vt^2 - \frac{t}{u} + 1 & \frac{vt^4}{u^2} - \frac{vt^3}{u} - \frac{t^2}{u^2} + \frac{t}{u} & t^2 - \frac{t^3}{u} & \frac{vt^3}{u} - \frac{vt^4}{u^2} \\
\frac{t}{u} - t^2 & -ut^3 + u^2t^2 + t^2 & (vt^2-1)(ut - t^2 - 1 + \frac{t}{u}) & ut^3 - u^2t^2 - t^2 + ut & vt^4 - vut^3 - \frac{vt^3}{u} + vt^2 \\
-\frac{vt^3}{u} + \frac{t^3}{u} + \frac{vt^2}{u^2} & \frac{vt^3}{u} - t^2 + ut & \frac{vt^4}{u^2} - \frac{t^3}{u} + t^2 & -\frac{t^3}{u} + t^2 - ut - \frac{t}{u} + 1 & -\frac{vt^4+vt^2}{u^2} + \frac{vt^3+vt}{u} - vt^2 \\
\frac{t}{u} - t^2 & -ut^3 + u^2t^2 + t^2 & -t^4 + ut^3 + \frac{t^3}{u} & ut^3 - u^2t^2 - t^2 + ut & t(t^2+1)(t - u - \frac{1}{u}) + 1
\end{pmatrix} \quad (6.1)$$

where we sum up the rows to get the vector $\mathbf{v}$

$$\mathbf{v} = \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{pmatrix} = \begin{pmatrix} t^2 - vt^2 + 1 \\ t^2 - \dfrac{t^3}{u} + \dfrac{vt^3}{u} - vt^2 \\ 1 \\ t^2 - \dfrac{t}{u} + 1 - vt^2 + \dfrac{vt}{u} + \dfrac{vt^3}{u} - \dfrac{t^3}{u} \\ 1 \end{pmatrix}.$$

The determinant from the matrix above (6.1) gives us the kernel for this problem, which is,

$$K(t,u,v) = -vt^4 + vt^3u + \frac{vt^3}{u} - vt^2 + t^4 - t^3u - \frac{t^3}{u} + t^2 - tu - \frac{t}{u} + 1.$$

We can now solve for the roots of $K(t,u,v) = 0$ which are then given by

$$\begin{aligned} u_1(t,v) =& -\frac{-vt^4 + t^4 - vt^2 + t^2 + 1}{2t(vt^2 - t^2 - 1)} \\ & - \frac{\sqrt{v^2t^8 - 2vt^8 - 2v^2t^6 + t^8 + 4vt^6 + v^2t^4 - 2t^6 + 4vt^4 - 5t^4 - 2vt^2 - 2t^2 + 1}}{2t(vt^2 - t^2 - 1)} \\ u_2(t,v) =& \frac{vt^4 - t^4 + vt^2 - t^2 - 1}{2t(vt^2 - t^2 - 1)} \\ & + \frac{\sqrt{v^2t^8 - 2vt^8 - 2v^2t^6 + t^8 + 4vt^6 + v^2t^4 - 2t^6 + 4vt^4 - 5t^4 - 2vt^2 - 2t^2 + 1}}{2t(vt^2 - t^2 - 1)} \\ & . \end{aligned}$$

Taking the limit for $t \to 0$, we can see that u_2 is the small root and u_1 is a large root. So, for this calculation we are going to need u_2. From Lemma 4.4, the meander generating function is $M(t,u,v) = \Phi(t,u,v)/(u^e \cdot K(t,u,v))$, where

$$\Phi(t,u,v) := u^e \left(F_1(t,u,v), \ldots, F_5(t,u,v)\right) \mathbf{v}(t,u,v).$$

From Theorem 4.4 we also know the structure of $\mathbf{F} = F_1, \ldots, F_5$ which consists of the vector difference $(1,0,0,0,0)$ minus the forbidden transitions that make the meander go negative. Thinking about our states we can see that since `UDUD` is a meander in itself once we are on a legal step and append the states of the pattern we will not go into the negative height. The only possible danger comes from the first state, the empty state. If in the previous step we had a legal meander at height zero and take another step into the negative direction we violate the non negativity constraint. So from here we conclude that $\mathbf{F}$ is of the form

$$\mathbf{F} = \begin{pmatrix} 1 - \frac{t}{u}[u^0]M(t,u,v) \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}.$$

The first entry contains the 1 that was previously mentioned and the other term we have described above as taking a legal meander at height zero $[u^0]M(t,v,u)$ and going one step into the negative t/u. This must be subtracted since we cannot count it as a possibility. Even though, we do not yet know the explicit form of this term $[u^0]M(t,u,v)$ we can continue with our calculations. This unknown term is only dependent on t and v since the operator $[u^0]$ implies that there is no mention of u anywhere after that. For brevity, we write $[u^0]M(t,u,v) =: F(t,v)$, where the notation reflects that this quantity depends only on t and v. We know from the proof of Lemma 4.4 the following equation holds

$$(F_1(t,u_i(t,v),v), F_2(t,u_i(t,v),v), \ldots, F_5(t,u_i(t,v),v)) \cdot \mathbf{v}(t,u_i(t,v),v) = 0.$$

for all u_i, such that u_i is a small root of $K(t,u,v)$. So in our case we need to plug in $u_2(t,v)$. So we can rearrange this equation and express $F(t,v)$ via this relation. So, we get then this equation

$$\begin{aligned}
\begin{pmatrix} 1-\frac{t}{u_2(t,v)}F(t,v) \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \cdot v(t,u_2(t,v)) = \left(1-\frac{t}{u_2(t,v)}F(t,v)\right)\cdot v_1 = 0 \\
\iff \qquad \left(1-\frac{t}{u_2(t,v)}F(t,v)\right) = 0 \\
\iff \qquad \frac{t}{u_2(t,v)}F(t,v) = 1 \\
\iff \qquad \frac{u_2(t,v)}{t} = F(t,v)
\end{aligned}$$

This gives us an explicit description of $F(t,v)$ which then gives us a representation of Φ

$$\begin{aligned}
\Phi(t,u,v) = u\cdot(1-\frac{t}{u}\cdot F(t,v))\cdot(t^2-vt^2+1) &= u\cdot(1-\frac{t}{u}\cdot\frac{u_2(t,v)}{t})\cdot(t^2-vt^2+1) \\
&= (u-u_2(t,u))\cdot(t^2-vt^2+1)
\end{aligned}$$

Substituting this expression into the formula of the meander generating function, we obtain the trivariate generating function

$$M(t,u,v) = \frac{\phi_1(t,u,v)}{uK(t,u,v)} = \frac{(u-u_2(t,u))\cdot(t^2-vt^2+1)}{u\cdot K(t,u,v)}.$$

This means that we get

$$G(t,v,u) = t^2 - vt^2 + 1$$

We now have found the meander generating function, since we are looking at Dyck paths we have to have a final altitude of zero. In accordance with Theorem 4.4 we must let u go to zero. This gives us the following generating function

$$M(t,0,v) = E(t,v) = \frac{u_2(t,v)(t^2-vt^2+1)}{(1-v)t^3+t}$$

The correctness of this generating function can be seen when we do a multivariate Taylor expansion around $t = 0$. This expansion was computed with Maple and yields

$$\begin{aligned}
E(t, v) = \\
&1 + t^2 + t^4(v + 1) + t^6(v^2 + v + 3) + t^8(v^3 + v^2 + 5v + 7) \\
&+ t^{10}(v^4 + v^3 + 7v^2 + 14v + 19) + t^12(v^5 + v^4 + 9v^3 + 22v^2 + 46v + 53) \\
&+ t^{14}(v^6 + v^5 + 11v^4 + 31v^3 + 82v^2 + 150v + 153) \\
&+ t^{16}(v^7 + v^6 + 13v^5 + 41v^4 + 127v^3 + 299v^2 + 495v + 453) \\
&+ t^{18}(v^8 + v^7 + 15v^6 + 52v^5 + 181v^4 + 507v^3 + 1087v^2 + 1651v + 1367) \\
&+ O(t^{20}).
\end{aligned}$$

These values can be compared with the values from A094507 and it can be seen that the function we found is indeed correct. As expected we only find even powers of t and the polynomial associated with each power of t^{2n} gives the number of possible Dyck paths in the $n-$th row. The numbers must be read in reverse order. The power gives us the position in the table and the coefficient the value.

The bivariate generating function obtained above refines the class of Dyck paths by the number of (possibly overlapping) occurrences of the consecutive pattern `UDUD`, where $U = (1, 1)$ and $D = (1, -1)$. As a consequence, several classical integer sequences arise naturally as specializations of this generating function.

Fixing the parameter v and extracting the coefficient of t^{2n} yields the number of Dyck paths of semilength n with exactly k occurrences of the pattern `UDUD`.

The column $[v^1]$, thus $k = 1$, counting Dyck paths with exactly one occurrence of the pattern, corresponds to A244236 and can be recovered directly from the explicit form of the generating function.

On the other hand, summing over all values of k for fixed n amounts to forgetting the pattern statistic and therefore counts all Dyck paths of semilength n. This explains why the row sums recover the Catalan numbers (A000108), a fact which is consistent with the combinatorial interpretation.

Finally, the diagonal extraction, corresponding to Dyck paths of semilength $2n$ with exactly n occurrences of `UDUD`, gives rise to the sequence A304361. This illustrates how a single bivariate generating function encapsulates a rich family of enumerative results, each obtained by a different specialization or coefficient extraction.

To the best of the author's knowledge, the results of this section contain the following new contributions.

First, the vectorial kernel method has not previously been applied to the problem of counting pattern occurrences in Dyck paths. While Sapounakis, Tasoulas and Tsikouras [STT07] study this problem using Chebyshev polynomials and functional equations, the present derivation is the first fully automatic treatment via the vectorial kernel method.

Second, the trivariate generating function $M(t, u, v)$, where t marks the length, u the final altitude, and v the number of pattern occurrences, appears to be new. It is more general than the bivariate function recorded in [The26, A094507], which tracks only length and pattern occurrences.

Finally, the asymptotic constant for the avoidance case,

$$E_n \sim 1.3373 \cdot n^{-3/2} \cdot (0.53101\ldots)^{-n},$$

was derived here.

6.5 A023432: Cornerless Motzkin Paths

The sequence A023432 from the OEIS [The26, A023432] counts Motzkin paths of length $n-1$ avoiding the consecutive patterns `UD`, `UU`, and `DD`, where $U = (1,1)$, $D = (1,-1)$, and $H = (1,0)$. The sequence begins

$$1, 1, 1, 1, 2, 4, 7, 12, 22, 42, 80, 152, 292, 568, 1112, \ldots$$

Equivalently, it counts Motzkin paths in which every down-step is immediately followed by a horizontal step. These paths appear naturally in RNA secondary structure models and the study of constrained plane trees [ABR20].

6.5.1 Automaton Description

Since the admissibility of each step depends only on the immediately preceding step, the model is naturally encoded by a finite automaton whose states record the last step taken. We introduce four states:

ε (initial), U (last step up), H (last step horizontal), D (last step down).

The forbidden patterns `UD`, `UU`, and `DD` are enforced by removing the corresponding transitions: from state U neither an up-step nor a down-step is permitted, from state D neither a down-step is permitted, and from state ε we allow all steps.

The transition matrix with states ordered as (ε, U, H, D) is

$$A = \begin{pmatrix} 0 & u & 1 & \frac{1}{u} \\ 0 & 0 & 1 & 0 \\ 0 & u & 1 & \frac{1}{u} \\ 0 & u & 1 & 0 \end{pmatrix}.$$

6.5.2 Kernel and Adjugate Matrix

Introducing the length variable t, the kernel matrix is $I - tA$:

$$I - tA = \begin{pmatrix} 1 & -tu & -t & -\frac{t}{u} \\ 0 & 1 & -t & 0 \\ 0 & -tu & 1-t & -\frac{t}{u} \\ 0 & -tu & -t & 1 \end{pmatrix}.$$

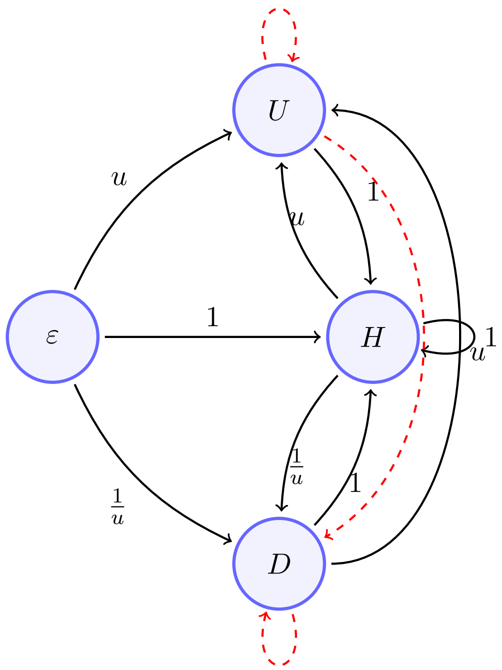


Figure 6.3: Automaton for Motzkin paths avoiding `UU`, `UU`, and `DD`. Solid arrows show permitted transitions weighted by height change. Dashed red arrows indicate forbidden transitions.

Computing the adjugate matrix $\operatorname{adj}(I - tA)$ gives

$$\begin{pmatrix} -t^3 - ut^2 - \frac{t^2}{u} - t + 1 & t^2 + ut & t^3 + ut^2 + \frac{t^2}{u} + t & \frac{t}{u} \\ 0 & -\frac{t^2}{u} - t + 1 & t & \frac{t^2}{u} \\ 0 & t^2 + ut & 1 & \frac{t}{u} \\ 0 & tu & ut^2 + t & -ut^2 - t + 1 \end{pmatrix}$$

and multiplying by the all-ones vector $\mathbf{1} = (1, 1, 1, 1)^\top$ yields

$$\mathbf{v} = \operatorname{adj}(I - tA)\,\mathbf{1} = \begin{pmatrix} 1 + t^2 + ut + \frac{t}{u} \\ 1 \\ 1 + t^2 + ut + \frac{t}{u} \\ tu + 1 \end{pmatrix}.$$

The kernel is

$$K(t, u) = \det(I - tA) = 1 - t - \frac{t^2(u^2 + 1)}{u} - t^3.$$

Multiplying through by u gives the polynomial

$$u\,K(t, u) = u - tu - t^2u^2 - t^2 - t^3u,$$

which has degree $c = 1$ in u^{-1}, so there is exactly one small root $u_0(t)$ satisfying $u_0(0) = 0$.

Among the four states, only state D can cause the path to violate the non-negativity constraint, since a down-step at height zero would take the path below the axis. The correction vector is therefore

$$\mathbf{F} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ -\frac{t}{u}[u^0]M(t,u) \end{pmatrix},$$

where $[u^0]M(t,u) = E(t)$ is the excursion generating function.

Applying the vectorial kernel method via the condition $\mathbf{F}(t,u_0) \cdot \mathbf{v}(t,u_0) = 0$ gives

$$1 \cdot \left(1 + t^2 + u_0 t + \frac{t}{u_0}\right) - \frac{t}{u_0} E(t) \cdot (tu_0 + 1) = 0,$$

which yields the excursion generating function

$$E(t) = \frac{1 + t^2 + u_0 t + \frac{t}{u_0}}{tu_0 + 1} \cdot \frac{u_0}{t}.$$

Expanding as a power series in t gives

$$E(t) = 1 + t + t^2 + t^3 + 2t^4 + 4t^5 + 7t^6 + 12t^7 + 22t^8 + \cdots,$$

which agrees with the values of A023432, confirming the correctness of the construction. We get the meander generating function by by taking $u \cdot \mathbf{F}(t,u)\mathbf{v}(t,u)$ and assigning this the name φ. This we can split into a part with $u - u_0(t)$ and into the rest giving us the following form

$$\varphi(t,u) = (u - u_0(t))(1 + tu).$$

So we call the polynomial $G(t,u) = 1 + tu$. And then we can pull up the general Theorem of meander generating functions (4.4) which gives us

$$M(t,u) = \frac{(u - u_0(t))(1 + tu)}{u\,K(t,u)}.$$

6.5.3 Asymptotic Analysis

By similar calculations as in the previous examples, we verify that the model is generic with dominant singularity $\rho \approx 0.4656$ and collision height $\tau = 1$, confirming zero drift. The kernel conditions $K(\rho,\tau) \approx 0$ and $K_u(\rho,\tau) \approx 0$ confirm a genuine branch point, and one checks that $K_{uu}(\rho,\tau) \approx -0.4335 \neq 0$ and $K_t(\rho,1) \approx -3.5126 \neq 0$, so the model satisfies all genericity conditions. Applying Theorem 5.7 with $G(\rho,0) = 1$, $Y(\rho) = 1$, $D(\rho) = -\rho^2 \approx -0.2168$, yields the asymptotic constant

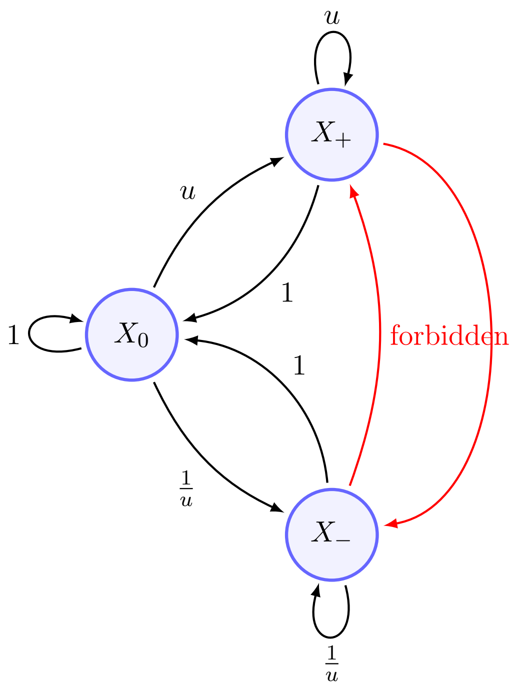
$$C \approx 3.5747,$$

so that the number of cornerless Motzkin paths of length n satisfies

$$E_n \;\approx\; 3.5747... \cdot n^{-3/2} \cdot (0.4655712319...)^{-n}.$$

The asymptotic behaviour of the coefficients E_n is known; an explicit formula appears in [The26, A023432], due to Kotesovec (2014). The derivation presented here recovers this result via the vectorial kernel method, providing an independent confirmation via a different approach.

A308435 Motzkin Paths Without Peaks and Valleys



We now revisit Example 8.2 of [ABBG20], which concerns Motzkin paths without peaks and valleys. Although this example has already been treated in the literature, we reproduce the derivation independently in order to further illustrate the vectorial kernel method.

A Motzkin path is a lattice path starting at height 0, never going below the horizontal axis, and consisting of steps

$$+1 \quad \text{(up)}, \qquad 0 \quad \text{(level)}, \qquad -1 \quad \text{(down)}.$$

A *peak* is an occurrence of the pattern $(+1, -1)$, while a *valley* is an occurrence of the pattern $(-1, +1)$. We are interested in Motzkin paths avoiding both patterns.

Avoidance of peaks and valleys can be enforced by remembering only the last step taken. Accordingly, we introduce three states:

$$X_0 \quad \text{(neutral)}, \qquad X_+ \quad \text{(last step was } +1), \qquad X_- \quad \text{(last step was } -1).$$

From state X_+, a down-step is forbidden, as it would create a peak. Similarly, from state X_-, an up-step is forbidden, as it would create a valley.

We now create the transition matrix. As always, let u mark the altitude and t mark the length of the path. With the states ordered as (X_0, X_+, X_-), the transition matrix A is given by

$$A = \begin{pmatrix} & X_0 & X_+ & X_- \\ X_0 & 1 & u & \frac{1}{u} \\ X_+ & 1 & u & 0 \\ X_- & 1 & 0 & \frac{1}{u} \end{pmatrix}.$$

The kernel matrix is obtained in the usual way as

$$I - tA = \begin{pmatrix} 1-t & -tu & -\frac{t}{u} \\ -t & 1-tu & 0 \\ -t & 0 & 1-\frac{t}{u} \end{pmatrix},$$

where we can now apply the adjugate operator. Let

$$\operatorname{adj}(I - tA) = \begin{pmatrix} t^2 - tu - \frac{t}{u} + 1 & tu - t^2 & \frac{t}{u} - t^2 \\ t - \frac{t^2}{u^2} & -\frac{t}{u} - t + 1 & \frac{t^2}{u} \\ t - t^2 u & t^2 u & -ut - t + 1 \end{pmatrix}.$$

As in the general theory, we define the vector

$$\mathbf{v} = \operatorname{adj}(I - tA)\,\mathbf{1} = \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} = \begin{pmatrix} 1 - t^2 \\ 1 - \frac{t}{u} \\ 1 - tu \end{pmatrix}, \tag{6.2}$$

where $\mathbf{1} = (1,1,1)^{\mathsf{T}}$. The kernel is defined as

$$K(t,u) := \det(I - tA). \tag{6.3}$$

A direct computation yields

$$K(t,u) = \frac{t^3 u + t^2 u - tu^2 - tu - t + u}{u}, \tag{6.4}$$

which agrees with the expression given in the reference. Solving the kernel equation $K(t,u) = 0$ for u gives us a unique small root $u_0(t)$ satisfying

$$\lim_{t \to 0} u_0(t) = 0.$$

Its explicit form is

$$u_0(t) = \frac{1 - t + t^2 + t^3 - \sqrt{(1-t^4)(1-2t-t^2)}}{2t}. \tag{6.5}$$

Next we want to consider the vector $\mathbf{F}$. In the present situation, only transitions that end in the negative state X_- can lead to a violation of the non-negativity constraint. So we can already fill in the first two entries of the vector $\mathbf{F}$ as 1 for the empty path and 0. In the last entry we must now think how a legal path can become negative with one step downwards. This means that the previous height was zero. This gives us the following form

$$\mathbf{F} = \begin{pmatrix} 1 \\ 0 \\ -\frac{t}{u}[u^0]M(t,u) \end{pmatrix}. \tag{6.6}$$

By the general vectorial kernel method for meanders in theorem 4.4, we have the following relation

$$\mathbf{F}(t,u_i) \cdot \mathbf{v}(t,u_i) = 0, \tag{6.7}$$

where u_i can be any of the small roots of the kernel. Since there is exactly one small root $u_0(t)$, this equation uniquely determines

$$[u^0]M(t,u). \tag{6.8}$$

Interestingly this also happens to be the generating function for excursions. Which we will later see. The following calculations give us the construction of $[u^0]M(t,u)$

$$\begin{aligned} 1 \cdot (1-t^2) - \frac{t}{u_0(t)}[u^0]M(t,u) \cdot (1 - tu_0(t)) = 0 \\ 1 - t^2 = \frac{t}{u_0(t)}[u^0]M(t,u) \cdot (1 - tu_0(t)) \\ \frac{1-t^2}{1-tu_0(t)} \cdot \frac{u_0(t)}{t} = [u^0]M(t,u). \end{aligned}$$

Now that the form of $[u^0]M(t,u)$ is known, we can calculate $\Phi(t,u)$ in the following fashion

$$\Phi(t,u) = u^e \, \mathbf{F}(t,u) \cdot \mathbf{v}(t,u), \tag{6.9}$$

where $e = 1$ is the absolute value of the minimal step. The bivariate generating function for Motzkin meanders without peaks and valleys is then given by

$$M(t,u) = \frac{\Phi(t,u)}{u\,K(t,u)}. \tag{6.10}$$

Plugging all of the information in gives

$$M(t,u) = \frac{(1-t^2)(u-u_0(t))}{(1-tu_0(t))uK(t,u)}$$

as [ABBG20] predicted.

Excursions. Finally, setting $u = 0$ yields the generating function for excursions,

$$E(t) := M(t, 0) \tag{6.11}$$

$$= \frac{1-t^2}{1-tu_0(t)} \cdot \frac{u-u_0(t)}{uK(t,u)}\bigg|_{u=0}, \tag{6.12}$$

which simplifies to

$$E(t) = \frac{(1-t^2)u_0(t)}{t-t^2u_0(t)} = \frac{(1+t)(1-t)^2 - \sqrt{(1-t^4)(1-2t-t^2)}}{2t^2}. \tag{6.13}$$

This term is already known to us, as it is the same as $[u^0]M(t, u)$. This generating function corresponds to the sequence A308435 in the OEIS, thereby confirming the correctness of the construction.

Asymptotic Analysis We now derive the asymptotic behaviour of the coefficients E_n, where E_n counts Motzkin paths of length n avoiding peaks and valleys.

Dominant singularity. The singularities of $E(t)$ arise where the discriminant

$$\Delta(t) = (1-t^4)(1-2t-t^2)$$

vanishes. The factor $1-t^4$ contributes roots at $t = \pm 1$ and $t = \pm i$, while the factor $1-2t-t^2$ contributes roots at

$$t = -1 \pm \sqrt{2}.$$

The smallest positive real root is

$$\rho = \sqrt{2} - 1 \approx 0.4142,$$

which is the dominant singularity of $E(t)$. At ρ we have $1 - \rho^4 \approx 0.9706 \neq 0$, so the singularity arises entirely from the factor $1 - 2t - t^2$ vanishing, giving a simple square root singularity.

Collision height and drift. The small root of the kernel evaluated at ρ gives

$$\tau = u_0(\rho) = 1.$$

Since $\tau = 1$ we are in the zero drift case, consistent with the symmetry of the Motzkin step set $\{+1, 0, -1\}$.

Genericity conditions. We verify the five conditions of Chapter 5.

- The generating function $E(t)$ is algebraic, involving a square root singularity, hence not rational.
- The dominant singularity $\rho = \sqrt{2} - 1$ is unique, algebraic, and arises from a branch collision rather than a pole.

- The relevant factor in the meander generating function is $G(t,u) = (1-t^2)/(1-tu_0(t))$, which is a polynomial in t. Also it is analytic at ρ with $G(\rho,0) = \sqrt{2} \approx 1.41421$.
- There are no large negative branches interfering at ρ.
- One verifies that $K_t(\rho,1) \approx -1.6569 \neq 0$, so the root of $K(t,1)$ at ρ is simple.

Additionally we verify the branch point conditions:

$$K(\rho,\tau) \approx 0, \qquad K_u(\rho,\tau) \approx 0, \qquad K_{uu}(\rho,\tau) = -0.8284 \neq 0,$$

confirming a genuine branch point of the kernel curve at (ρ,τ).

Asymptotic constant. We apply Theorem 5.7 with $e = 1$ and $(-1)^e = -1$. The ingredients are

$$\begin{aligned} G(\rho,0) &= \sqrt{2} \approx 1.41421, \\ Y(\rho) &= 1 \quad \text{(empty product since } e = 1\text{)}, \\ D(\rho) &= [u^0]\, u\, K(\rho,u) = -(\sqrt{2}-1) \approx -0.4142, \\ K_t(\rho,\tau) &\approx -1.6569, \\ K_{uu}(\rho,\tau) &\approx -0.8284. \end{aligned}$$

Substituting into the formula gives

$$\begin{aligned} C &= (-1)^e \frac{G(\rho,0)\, Y(\rho)}{D(\rho)} \sqrt{\frac{\rho\, K_t(\rho,\tau)}{2\pi\, K_{uu}(\rho,\tau)}} \\ &\approx (-1)\cdot(-1)\cdot \frac{\sqrt{2}}{(\sqrt{2}-1)} \sqrt{\frac{(\sqrt{2}-1)\cdot(-1.6569)}{2\pi\cdot(-0.8284)}} = 1.2397. \end{aligned}$$

Result. The number of Motzkin paths of length n avoiding peaks and valleys satisfies

$$\boxed{E_n \;\approx\; 1.2397\ldots \cdot n^{-3/2} \cdot (\sqrt{2}-1)^{-n}.}$$

Note that $E(t)$ has a single dominant singularity at $\rho = \sqrt{2}-1$, with no parity issue, so no variable substitution is needed.

7 Conclusion and Outlook

7.1 Summary of Results

This thesis has provided a detailed and self-contained exposition of the vectorial kernel method, following and extending the work of Asinowski, Bacher, Banderier, and Gittenberger [ABBG20]. Where the original paper presents results concisely, we have developed the arguments in full, supplied all intermediate steps, and added the analytic and combinatorial context needed to make the method accessible.

The main contributions of this thesis are as follows.

Detailed proofs. We have provided complete proofs of all central results: the generating function of walks avoiding a generic pattern (Theorem 4.2), the generating functions of bridges, meanders, and excursions (Theorems 4.4 and 4.5), the asymptotic formulas for all four path families (Theorems 5.4, 5.7, 5.8, and 5.12), and the quasi-meander subcase Theorem 4.6. In several places we have identified and corrected minor errors in the formulas of [ABBG20].

Worked examples covering all cases. Chapter 6 illustrates the theory through concrete examples. We derived explicit generating functions and asymptotics for zero-drift, positive-drift, and negative-drift meanders, verified the results numerically, and identified all examples with sequences in the OEIS.

Extension beyond the standard setting. The vectorial kernel method as formulated in [ABBG20] handles patterns encoded by the canonical pattern-matching automaton. In this thesis we have shown that the method extends to any finite automaton: the transition matrix, kernel, and adjugate construction apply verbatim, and the boundary correction vector $\mathbf{F}$ can be determined from the automaton's structure by identifying which transitions may violate non-negativity. This is demonstrated concretely in the examples A094507 and A023432, where the automaton is augmented with an additional counting variable or uses a non-standard state space.

New results. For the sequence A094507 (Dyck paths by number of occurrences of `UDUD`), we derived the trivariate generating function $M(t, u, v)$ and the explicit closed form

$$E(t, v) = \frac{u_2(t, v)(1 - vt^2 + t^2)}{(1 - v)t^3 + t},$$

which is new. The OEIS records only an implicit quadratic equation for the bivariate specialisation. Furthermore, this appears to be the first fully automatic treatment of this problem via the vectorial kernel method. The asymptotic constant for this sequence does not appear in the existing literature and is therefore new. For the three mock examples, the asymptotic constants were computed explicitly and are believed to be new, though no claim of novelty beyond the numerical evaluation is made.

7.2 Outlook

The results of this thesis naturally lead to the following open problems.

Multiple forbidden patterns. The framework of [ABBG20] and the present thesis handles a single forbidden pattern. The paper [ABR20] extends the vectorial kernel method to sets of forbidden patterns, where the automaton tracks partial matches with each pattern simultaneously. A natural next step would be to develop the asymptotic theory for several forbidden patterns and to apply it to specific combinatorial sequences that are currently out of reach of the single-pattern theory.

Asymptotics of bivariate generating functions. In the A094507 example, we obtained the bivariate generating function $E(t, v)$ tracking the number of pattern occurrences, but performed the asymptotic analysis only at $v = 0$. A more complete analysis would determine the asymptotic behaviour of the coefficients $T(n, k)$ for fixed k, or for k growing with n. This requires understanding the singularity structure of $E(t, v)$ as a function of both variables, and falls within the theory of multivariate generating functions [FS09].

Non-generic models. The asymptotic results in Chapter 5 require the genericity conditions of Definition 5.1. Understanding the asymptotic behaviour when one or more of these conditions fails, for instance when the dominant singularity is a higher-order branch point, or when two singularities of equal modulus coincide, would broaden the scope of the theory considerably. Some of these degenerate cases are discussed briefly in [ABBG20]; a systematic treatment remains open.

Weighted step sets. Throughout this thesis, each step is given weight 1. Allowing steps to carry positive real weights would model random walks with non-uniform step distributions and is directly relevant to probabilistic applications such as queuing theory. The kernel method extends to this setting, but the asymptotic analysis requires additional care, since the dominant singularity and the drift depend on the weights in a more complex way.

Bibliography

[ABBG20] A. Asinowski, A. Bacher, C. Banderier, and B. Gittenberger. Analytic combinatorics of lattice paths with forbidden patterns, the vectorial kernel method, and generating functions for pushdown automata. *Algorithmica*, 82(2):386–428, 2020. DOI: 10.1007/s00453-019-00623-3.

[ABR20] A. Asinowski, C. Banderier, and V. Roitner. Generating functions for lattice paths with several forbidden patterns. In *Séminaire Lotharingien de Combinatoire*, 2020.

[BF02] C. Banderier and P. Flajolet. Basic Analytic Combinatorics of Directed Lattice Paths. *Theoretical Computer Science*, 281(1–2):37–80, 2002.

[FO90] P. Flajolet and A. M. Odlyzko. Singularity analysis of generating functions. *SIAM Journal on Discrete Mathematics*, 3(2):216–240, 1990. DOI: 10.1137/0403019.

[FS09] P. Flajolet and R. Sedgewick. *Analytic Combinatorics*. Cambridge University Press, Cambridge, 2009. URL: https://doi.org/10.1017/CBO9780511801655.

[GO81] L. J. Guibas and A. M. Odlyzko. String overlaps, pattern matching, and nontransitive games. *Journal of Combinatorial Theory, Series A*, 30(2):183–208, 1981. DOI: 10.1016/0097-3165(81)90058-9.

[HU79] J. Hopcroft and J. Ullman. *Introduction to Automata Theory, Languages, and Computation*. Addison-Wesley, 1979. URL: https://books.google.at/books?id=G_BQAAAAMAAJ.

[Jän06] K. Jänich. *Funktionentheorie: Eine Einführung*. Springer Berlin Heidelberg, 2006. URL: https://books.google.de/books?id=lRX0BQAAQBAJ.

[Knu68] D. E. Knuth. *The Art of Computer Programming, Volume 1: Fundamental Algorithms*. Addison-Wesley, Boston, 1968.

[Mat67] G. H. Matthews. N. Chomsky and M. P. Schützenberger. The algebraic theory of context-free languages. Computer programming and formal systems, edited by P. Braffort and D. Hirschberg, Studies in logic and the foundations of mathematics, North-Holland Publishing Company, Amsterdam1963, pp. 118–161. *Journal of Symbolic Logic*, 32(3):388–389, 1967. DOI: 10.2307/2270782.

[Ste98] G. Stewart. On the adjugate matrix. *Linear Algebra and its Applications*, 283:151–164, 1998.

[STT07] A. Sapounakis, I. Tasoulas, and P. Tsikouras. Counting strings in Dyck paths. *Discrete Mathematics*, 307:2909–2924, 2007. DOI: 10.1016/j.disc.2007.03.005.

[The26] The OEIS Foundation Inc. The On-Line Encyclopedia of Integer Sequences, 2026. URL: `https://oeis.org`.

[Wal62] R. J. Walker. *Algebraic Curves*. Dover Publications, New York, 1962.