

EPTCS 403

Proceedings of the
**13th edition of the conference on
Random Generation of Combinatorial
Structures. Polyominoes and Tilings**

Bordeaux, France, 24-28th June 2024

Edited by: Srečko Brlek and Luca Ferrari

Published: 24th June 2024
DOI: 10.4204/EPTCS.403
ISSN: 2075-2180
Open Publishing Association

Table of Contents

Table of Contents	i
Foreword..... <i>Srečko Brlek and Luca Ferrari</i>	iv
Invited Presentation: Snake Tilings, Skeletons Subshift and Self-Avoiding Walks <i>Nathalie Aubrun</i>	1
Invited Presentation: Cutoff for Permuted Markov Chains <i>Anna Ben-Hamou</i>	2
Invited Presentation: q,t -Combinatorics and Sandpiles..... <i>Michele D’Adderio</i>	3
Invited Presentation: Computing Tiling Properties of Polyforms..... <i>Craig S. Kaplan</i>	4
Invited Presentation: Random Automata <i>Cyril Nicaud</i>	5
On the Confluence of Directed Graph Reductions Preserving Feedback Vertex Set Minimality <i>Moussa Abdenbi, Alexandre Blondin Massé, Alain Goupil and Odile Marcotte</i>	6
Power Quotients of Plactic-like Monoids <i>Antoine Abram, Florent Hivert, James D. Mitchell, Jean-Christophe Novelli and Maria Tsalakou</i>	12
Self-descriptive Sequences directed by two Periodic Sequences <i>Shigeki Akiyama, Damien Jamet, Irène Marcovici and Mai-Linh Trân-Công</i>	18
About the Determinant of Complete non-ambiguous Trees <i>Jean-Christophe Aval</i>	23
Progressive and Rushed Dyck Paths..... <i>Axel Bacher</i>	29
Interval Posets and Polygon Dissections <i>Eli Bagno, Estrella Eisenberg, Shulamit Reches and Moriah Sigron</i>	35
Type-B analogue of Bell numbers using Rota’s Umbral calculus approach <i>Eli Bagno and David Garber</i>	43

Dyck Paths Enumerated by the Q-bonacci Numbers	49
<i>Elena Barcucci, Antonio Bernini, Stefano Bilotta and Renzo Pinzani</i>	
Diagram Calculus for the Affine Temperley–Lieb Algebra of Type D	54
<i>Riccardo Biagioli, Giuliana Fatabbi and Elisa Sasso</i>	
Optimal Generation of Strictly Increasing Binary Trees and Beyond	60
<i>Olivier Bodini, Francis Durand and Philippe Marchal</i>	
Uniform Sampling and Visualization of 3D Reluctant Walks	66
<i>Benjamin Buckley and Marni Mishna</i>	
Pop Stacks with a Bypass	73
<i>Lapo Cioni, Luca Ferrari and Rebecca Smith</i>	
Random Generation of Git Graphs	79
<i>Julien Courtiel and Martin Pépin</i>	
Bijjective Enumeration and Sign-Imbalance for Permutation Depth and Excedances	87
<i>Sen-Peng Eu, Tung-Shan Fu and Yuan-Hsun Lo</i>	
Square-Triangle Tilings: Lift & Flip to Sample?	92
<i>Thomas Fernique and Olga Mikhailovna Sizova</i>	
Restricted Permutations Enumerated by Inversions	96
<i>Atli Fannar Franklín, Anders Claesson, Christian Bean, Henning Úlfarsson and Jay Pantone</i>	
Bijections between Variants of Dyck Paths and Integer Compositions	101
<i>Manosij Ghosh Dastidar and Michael Wallner</i>	
Greedy Gray Codes for some Restricted Classes of Binary Words	108
<i>Nathanaël Hassler, Vincent Vajnovszki and Dennis Wong</i>	
Morphic Sequences: Complexity and Decidability	113
<i>Raphael Henry</i>	
Detecting Isohedral Polyforms with a SAT Solver	118
<i>Craig S. Kaplan</i>	
A Symmetry Property of Christoffel Words	123
<i>Yan Lanciault and Christophe Reutenauer</i>	
Combinatorics on Social Configurations	128
<i>Dylan Laplace Mermoud and Pierre Popoli</i>	
Perfectly Clustering Words and Iterated Palindromes over a Ternary Alphabet	134
<i>Mélodie Lapointe and Nathan Plourde-Hébert</i>	

On the Orthogonality of Generalized Pattern Sequences	139
<i>Shuo Li</i>	
Counting Polyominoes in a Rectangle $b \times h$	145
<i>Louis Marin</i>	
LLT Polynomials and Hecke Algebra Traces	150
<i>Alejandro H. Morales, Mark A. Skandera and Jiayuan Wang</i>	
HBS Tilings Extended: State of the Art and Novel Observations	156
<i>Carole Porrier</i>	
Counting Colored Tilings on Grids and Graphs	164
<i>José L. Ramírez and Diego Villamizar</i>	
A Bijection between Stacked Directed Polyominoes and Motzkin Paths with Alternative Catastrophes	169
<i>Florian Schager and Michael Wallner</i>	
Natural Measures on Polyominoes Induced by the Abelian Sandpile Model	177
<i>Andrea Sportiello</i>	
Construction of Minkowski Sums by Cellular Automata	191
<i>Pierre-Adrien Tahay</i>	
Local Limit Theorems for q -Multinomial and Multiple Heine Distributions	196
<i>Malvina Vamvakari</i>	

Random Generation of Combinatorial Structures. Polyominoes and Tilings

Foreword

Srečko Brlek	Luca Ferrari
UQAM, Canada	Università di Firenze, Italy
brlek.srecko@uqam.ca	luca.ferrari@unifi.it

This volume contains the proceedings of the 13th edition of the conference GASCom, held at LABRI, Bordeaux, during the week of June 24-28, 2024.

The very first edition of GASCom took place in Bordeaux on January 18 and 19, 1994, organized by Dominique Gouyou-Beauchamps and the late Jean-Guy Penaud, in conjunction with “Polyominoes and Tilings”, organized by Philippe Aigrin in Toulouse on January 20 and 21, 1994. As reported in the foreword of the dedicated special issue,

“Both meetings investigated the same types of combinatorial objects (polynominoes, paths, animals) but with different motivations. Counting and enumeration were among the interests common to both, but GASCom focused on random generation, Polyominoes and Tilings on the geometrical and topological problems in tiling and the complexity of tiling algorithms.”
(*Theoretical Computer Science*, 159 (1996) 1–2)

The second GASCom was organized by Jean-Guy-Penaud and Renzo Pinzani in Caen, again in conjunction with “Polyominoes and Tilings”, organized by Jacques Mazoyer, under the name “Journées d’hiver à Caen”, on January 30 and February 1, 1997. Again from the foreword, we quote

“They follow on from a previous edition, split between Bordeaux and Toulouse in January 1994, and the size of the audience common to both conferences prompted the organizers to repeat the experience. Readers of these proceedings will appreciate the wisdom of this choice” (translated from French, *Theoretical Computer Science*, 218 vol. 2 (1999) 217–218)

In 2024, GASCom returns to its origins, since it was the Ecole de Combinatoire de Bordeaux that planted the seeds of these joint conferences, reminding us that the link between the two subjects (combinatorial generation and polyominoes) is as strong as ever.

The conference features 5 invited talks and 33 contributed talks carefully selected by the members of the program committee. We warmly thank them for their efforts and dedication in the reviewing process. We also extend our thanks to the members of the organizing committee, who have worked hard to organize the conference in a friendly and stimulating atmosphere. Finally, we thank all the speakers, whose contributions are of course the key ingredients for the success of the conference.

To conclude, we wish to express our sincere gratitude to the staff of EPTCS, whose help and collaboration have been invaluable in the preparation of this volume.

Snake Tilings, Skeletons Subshift and Self-Avoiding Walks

Nathalie Aubrun

LISN, Université Paris-Saclay

nathalie.aubrun@lisn.upsaclay.fr

Wang tiles are squares with coloured edges that can be placed side by side as long as two neighbouring tiles have the same colour on their common side. Given a finite set of such tiles, whether it is possible to create an infinite tiling of the plane is an undecidable problem, known as the domino problem. This problem has also been studied for about fifteen years for groups of finite type other than \mathbb{Z}^2 . In this talk I will focus on a variant of this problem, the snake domino tiling. Instead of looking for a tiling of the whole plane (or the whole group), we only look for a tiling of an infinite path. This problem remains undecidable in the Euclidean plane \mathbb{Z}^2 . I will present recent progress on the snake tiling problem for groups, and make the connection with self-avoiding paths.

Cutoff for Permuted Markov Chains

Anna Ben-Hamou

Sorbonne Université, France

`anna.ben.hamou@sorbonne-universite.fr`

For a given finite Markov chain with uniform stationary distribution, and a given permutation on the state-space, we consider the Markov chain which alternates between random jumps according to the initial chain, and deterministic jumps according to the permutation. In this framework, Chatterjee and Diaconis (2020) showed that when the permutation satisfies some expansion condition with respect to the chain, then the mixing time is logarithmic, and that this expansion condition is satisfied by almost all permutations. We will see that the mixing time can even be characterized much more precisely: for almost all permutations, the permuted chain has cutoff, at a time which only depends on the entropic rate of the initial chain.

q, t -Combinatorics and Sandpiles

Michele D'Adderio

Università di Pisa, Italy

michele.dadderio@unipi.it

In 1987 Bak, Tang and Wiesenfeld introduced their famous sandpile model as a first system showing self-organized criticality. In 1988 Macdonald introduced his famous symmetric polynomials. Each of these two discoveries produced a huge amount of research that is still developing intensely today. But until recently, these two lines of research went on without any relevant interaction. In this talk we show how the combinatorics generated by these two important mathematical objects come together in a surprising way, proving that a synergy between these two topics is inevitable.

Computing Tiling Properties of Polyforms

Craig S. Kaplan

School of Computer Science
University of Waterloo
csk@uwaterloo.ca

Polyforms—shapes constructed by gluing together copies of cells in an underlying grid—are a convenient experimental tool with which to probe problems in tiling theory. They are expressive: in practice, even simple families like polyominoes exhibit many of the tiling-theoretic behaviours we might wish to study. But unlike the general world of shapes, polyforms can be enumerated exhaustively and the behaviour of each one examined using discrete computation. In this way polyforms can provide the raw data from which we might identify patterns, examples, or counterexamples that drive new insights into unsolved problems in tiling theory.

I am particularly interested in using polyforms to explore two open problems. The first asks whether there is a finite upper bound to a shape’s Heesch number, the maximum number of times a non-tiler may be surrounded by rings of copies of itself and yet still fail to tile the plane. The second asks whether there is a finite upper bound to a shape’s isohedral number, the minimum number of transitivity classes needed in any periodic tiling by the shape. I discuss these two problems and the progress made in exploring them using discrete computation. I also discuss the connections from Heesch numbers and isohedral numbers to the question of the existence of an aperiodic monotile, a problem that remained open for over sixty years but that was resolved in 2023 by a polyform called the “hat”.

Random Automata

Cyril Nicaud

Université Gustave Eiffel, Marne-La-Vallée, France

`cyril.nicaud@univ-eiffel.fr`

We will survey several results concerning random finite state automata, including random generation and algorithm analysis. We will place special focus on the subset construction, the standard algorithm for building a deterministic automaton equivalent to a given non-deterministic one.

On the Confluence of Directed Graph Reductions Preserving Feedback Vertex Set Minimality

Moussa Abdenbi

Université du Québec à Montréal
Québec, Canada

abdenbi.moussa@uqam.ca

Alexandre Blondin Massé

Université du Québec à Montréal
Québec, Canada

blondin_masse.alexandre@uqam.ca

Alain Goupil

Université du Québec À Trois-Rivières
Québec, Canada

alain.goupil@uqtr.ca

Odile Marcotte

Université du Québec à Montréal
Québec, Canada

odile.marcotte@videotron.ca

1 Introduction

In graph theory, the minimum directed feedback vertex set (FVS) problem consists in identifying the smallest subsets of vertices in a directed graph whose deletion renders the directed graph acyclic. In other words, a FVS in a directed graph G with vertex set V is a subset of V with a nonempty intersection with every circuit of G . Computing a minimum cardinality FVS (MFVS) is NP-hard [4, 5]. In this extended abstract we investigate graph reductions that preserve all or some minimum cardinality FVS and we focus on their properties, especially the Church-Rosser property, also called confluence. The Church-Rosser property implies the irrelevance of reduction order, leading to a unique digraph [2]. We explore graph reductions proposed for solving the MFVS problem, preserving the collection of MFVS or at least one of them [6, 7, 8]. The study seeks the largest set of reductions with the Church-Rosser property and explores the adaptability of reductions to meet this criterion. Addressing these questions is crucial, as it may have algorithmic implications, including potential parallelization and speeding up sequential algorithms in graph classes with polynomial algorithms [3, 9].

For sake of completeness, we recall some definitions and notation from graph theory.

A *directed graph* (or *digraph*) is an ordered pair $G = (V, E)$ where V is a finite set of *vertices* and $E \subseteq V \times V$ is a set of arcs. Let $G = (V, E)$ be a digraph, $u, v \in V$ and $U \subseteq V$. We denote $N_G^+(u) = \{s \in V \mid (u, s) \in E\}$ and $N_G^-(u) = \{p \in V \mid (p, u) \in E\}$ the set of successors and the set of predecessors of u respectively. For a vertex $u \in V$, $G - u$ denotes the digraph whose set of vertices is $V \setminus \{u\}$ and whose set of arcs is $E \setminus \{(x, y) \in E \mid x = u \text{ or } y = u\}$. Accordingly, for a given arc $(u, v) \in E$, $G - (u, v)$ is the digraph where the set of vertices is V and the set of arcs is $E \setminus \{(u, v)\}$. Similarly, the digraph $G \circ u$ is the digraph where the set of vertices is $V \setminus \{u\}$ and the set of arcs is $(E \setminus \{(x, y) \in E \mid x = u \text{ or } y = u\}) \cup N_G^-(u) \times N_G^+(u)$.

For $e = (u, v) \in E$, we say that e is a *2-way arc* in G if $(v, u) \in E$. The set of all 2-way arcs of G is denoted by $E^{\leftrightarrow} = \{(u, v) \in E \mid (v, u) \in E\}$. Given a digraph $G = (V, E)$ we distinguish two special digraphs $G^{\leftrightarrow} = (V, E^{\leftrightarrow})$ and $G^{\rightarrow} = (V, E^{\rightarrow})$ where $E^{\rightarrow} = E \setminus E^{\leftrightarrow}$.

A (*directed*) *path* of G is a sequence $p = (v_1, v_2, \dots, v_k)$ of vertices $v_i \in V$ for $i = 1, 2, \dots, k$ such that $(v_i, v_{i+1}) \in E$ for $i = 1, 2, \dots, k-1$. Moreover, a path p is called a *circuit* if $v_1 = v_k$. An arc (u, u) is called a *loop*. We say that G is *acyclic*, if there is no circuit in G . Given $U \subseteq V$, we say that U is a *directed clique* or *diclique* of G if for each $u, v \in U$ and $u \neq v$, we have $(u, v) \in E$ and $(u, u) \notin E$.

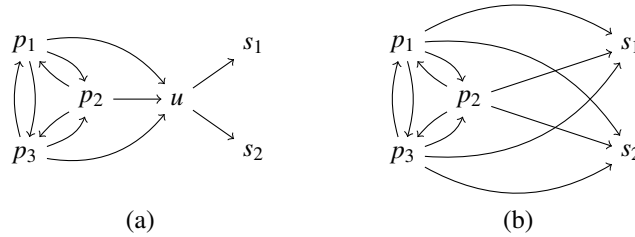


Figure 1: Illustration of the INDICLIQUE reduction. (a) $N_G^-(u)$ is a diclique, so INDICLIQUE(u) is applicable. (b) We remove u and all its incident arcs, and we add the new arcs (p_i, s_j) for $i \in \{1, 2, 3\}$ and $j \in \{1, 2\}$.

A set $U \subseteq V$ is called a *feedback vertex set* if $G' = (V', E')$ where $V' = V \setminus U$ and $E' = E \setminus \{(u, v) \mid u \in U \text{ or } v \in U\}$ is acyclic. The set of all feedback vertex sets of G is denoted by $FVS(G)$. The set of all *minimal feedback vertex sets*, in short $MFVS(G)$, is the set of feedback vertex sets with minimal cardinality.

2 Reductions

Given a digraph $G = (V, E)$, the problem of finding a minimum feedback vertex set is NP-hard [4]. However, in some cases we can use a set of *transformations* by which the size of the input graph can be reduced, with the guarantee that at least one minimum feedback vertex set in G could be constructed from a minimum feedback vertex set in the reduced graph, in polynomial time. These transformations are called *digraph reductions*.

In this context, by digraph reduction we mean a transformation of the digraph $G = (V, E)$ into a digraph $G' = (V', E')$ such that (1) either $|V'| < |V|$, or $|V'| = |V|$ and $|E'| < |E|$, and (2) an MFVS of G can be computed in polynomial time from any MFVS of G' .

In the following, we give a brief description of Levy and Low's [7] simple and straightforward reductions, followed by a generalization of two of their reductions by Lemaic [6], and additional reductions from Lin and Jou [8]. Let $G = (V, E)$ be a digraph and $u, v \in V$.

- The precondition of the reduction LOOP(u) is $(u, u) \in E$. This reduction transforms $G = (V, E)$ in $G - u$ and adds u to the MFVS in construction.
- The precondition of IN0(u) is $N_G^-(u) = \emptyset$. This reduction transforms $G = (V, E)$ in $G - u$.
- The precondition of OUT0(u) is $N_G^+(u) = \emptyset$. This reduction transforms $G = (V, E)$ in $G - u$.
- The precondition of IN1(u) is $(u, u) \notin E$ and $|N_G^-(u)| = 1$. The transformation consists of replacing G by the digraph $G \circ u$. This reduction does not necessarily preserve all the FVS of the original digraph but every MFVS of the reduced digraph is also an MFVS of the original graph.
- The precondition of OUT1(u) is $(u, u) \notin E$ and $|N_G^+(u)| = 1$. The transformation consists of replacing G by the digraph $G \circ u$. This reduction does not necessarily preserve all the FVS of the original digraph but every MFVS of the reduced digraph is also an MFVS of the original digraph.

Lemaic [6] proposed a generalization of IN1 and OUT1 based on the diclique concept.

- The precondition of INDICLIQUE(u) is $(u, u) \notin E$ and $N_G^-(u)$ forms a diclique in G . The transformation consists of replacing G by $G \circ u$. Similar to IN1(u), this reduction, illustrated in Figure 1,

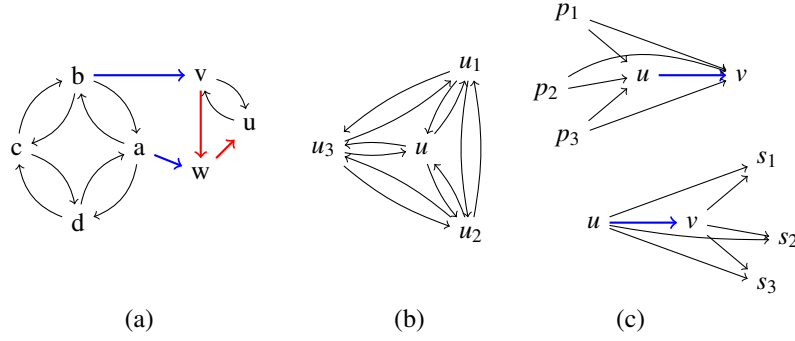


Figure 2: Illustration of Lin and Jou reductions preconditions. (a) PIE is applicable on the blue and red arcs. Indeed, there no circuit going through the blue arcs (b, v) and (a, w) in G , and therefore in G^{\rightarrow} and the same is true for the red arcs (v, w) and (w, u) in G^{\rightarrow} . Therefore we can remove the blue and red arcs from G . (b) $\text{CORE}(u)$ is applicable since u and its neighbors, $\{u, u_1, u_2, u_3\}$ form a diclique. So we can add $\{u_1, u_2, u_3\}$ to the MFVS and remove them from G . (c) In the top the first case of DOME and in the bottom the second case. The arc (u, v) is dominated and we can remove it from G .

does not necessarily preserve all the FVS of the original digraph but every MFVS of the reduced digraph is also an MFVS of the original digraph.

- The precondition of $\text{OUTDICLIQUE}(u)$ is $(u, u) \notin E$ and $N_G^+(u)$ forms a diclique in G . The transformation consists of replacing G by $G \circ u$. This reduction does not necessarily preserve all the FVS of the original digraph but every MFVS of the reduced digraph is also an MFVS of the original digraph.

Lin and Jou extended the work of Levy and Low by proposing the following three reductions [7, 8].

- The precondition of $\text{PIE}(u, v)$, for an arc (u, v) of G that is not a 2-way arc, is the following: there is no circuit in the digraph G^{\rightarrow} going through arc (u, v) . The transformation consists of replacing G with $G - (u, v)$. This reduction preserves all the FVS of the original digraph.
- The precondition of $\text{CORE}(u)$ is $\{u\} \cup N_G^-(u) \cup N_G^+(u)$ is a diclique of G . The transformation consists of removing all vertices $x \in N_G^-(u) \cup N_G^+(u)$ and add them to the MFVS and we replace G with $G - x$. This reduction does not necessarily preserve all the FVS of the original digraph but every MFVS of the reduced digraph is also an MFVS of the original digraph.
- The precondition of $\text{DOME}(u, v)$, for an arc (u, v) of G , is $N_{G^{\rightarrow}}^-(u) \subseteq N_G^-(v)$ (first case) or $N_{G^{\rightarrow}}^+(v) \subseteq N_G^+(u)$ (second case). The transformation consists of replacing G with $G - (u, v)$. This reduction preserves all the FVS of the original digraph.

See Figure 2 for illustrations of the preconditions of the Lin and Jou reductions.

3 The finite Church-Rosser property

Another way to see digraph reductions is to consider them as binary relations on the set of all digraphs \mathcal{G} . More precisely, a reduction R can be seen as a binary relation $\mathcal{R} \subseteq \mathcal{G} \times \mathcal{G}$. Hence for $G, G' \in \mathcal{G}$ if we can reduce G to G' with the reduction R , then we say that $(G, G') \in \mathcal{R}$. For a given reduction relation \mathcal{R} , we say that $G \in \mathcal{G}$ is \mathcal{R} -irreducible (or simply *irreducible* when the context is clear) if there does not exist $G' \in \mathcal{G}$ such that $(G, G') \in \mathcal{R}$.

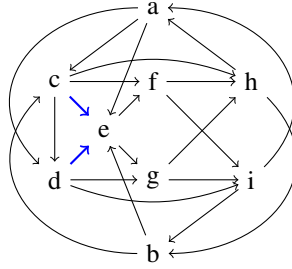


Figure 3: A digraph showing that the reductions in the article by Lin and Jou does not have the Church-Rosser property. If $G = (V, E)$ denotes the displayed digraph, the equalities $N_{G \rightarrow}^-(c) = \{a, b\} \subseteq \{a, b, c, d\} = N_G^-(e)$ and $N_{G \rightarrow}^-(d) = \{a, c\} \subseteq \{a, b, c, d\} = N_G^-(e)$ hold. Hence we can reduce G using $\text{DOME}(c, e)$ and $\text{DOME}(d, e)$. We can apply $\text{DOME}(d, e)$ followed by $\text{DOME}(c, e)$. If we first apply $\text{DOME}(c, e)$, however, the precondition of $\text{DOME}(d, e)$ is not verified. Indeed, we have $N_{G-(c,e)}^-(d) = \{a, c\} \not\subseteq \{a, b, d\} = N_{G-(c,e)}^-(e)$ and the graph $(V, E - (c, e))$ cannot be reduced further.

Now, given some $G \in \mathcal{G}$, one might wish to *reduce* G as much as possible by using the following procedure: (Step 1) if there is no G' such that $(G, G') \in \mathcal{R}$, then stop; (Step 2) otherwise, pick any $G' \in \mathcal{G}$ such that $(G, G') \in \mathcal{R}$; (Step 3) replace G by G' and repeat the previous steps. However, there is no guarantee that the final digraph is unique, since there might be more than one available candidate for G' at Step 2). An important property that could be satisfied by a set of reductions is the *Church-Rosser finiteness* property [2] also called *confluence* [1]. According to this property, the order in which a sequence of reductions is applied does not affect the final reduced graph.

In order to introduce more formally this property, we need some additional definitions. Let $\mathcal{R} \subseteq \mathcal{S} \times \mathcal{S}$ be any binary relation on a set \mathcal{S} and write $x\mathcal{R}y$ whenever $(x, y) \in \mathcal{R}$. The *reflexive closure* of \mathcal{R} , denoted by \mathcal{R}^R , is given by $\mathcal{R}^R = \mathcal{R} \cup \{(x, x) \mid x \in \mathcal{S}\}$. Its *transitive closure* is defined by $\mathcal{R}^T = \bigcup_{i=1}^{\infty} \mathcal{R}^i$ where \mathcal{R}^i is the composition of \mathcal{R} with itself i times. The *reflexive-transitive closure* of \mathcal{R} is then defined by $\mathcal{R}^{RT} = \mathcal{R}^R \cup \mathcal{R}^T$. The *completion* of \mathcal{R} is given by $\mathcal{R}^C = \{(x, y) \in \mathcal{R}^{RT} \mid \text{there does not exist } z \in \mathcal{S} \text{ such that } (y, z) \in \mathcal{R}\}$. A pair $(\mathcal{S}, \mathcal{R})$ is called *finite* if for $x, y \in \mathcal{S}$, there is a constant k such that if $x\mathcal{R}^i y$, then $i \leq k$.

We say that $(\mathcal{S}, \mathcal{R})$ has the *Church-Rosser finiteness property*, if $(\mathcal{S}, \mathcal{R})$ is finite and for $x, y, z \in \mathcal{S}$, if $(x, y) \in \mathcal{R}^C$ and $(x, z) \in \mathcal{R}^C$, then $y = z$. The following theorem proved in [10] gives a simpler test for Church-Rosser finiteness property.

Theorem 1 (Sethi [10]). *Let \mathcal{R} be a relation on a set \mathcal{S} . Then $(\mathcal{S}, \mathcal{R})$ is Church-Rosser finite if and only if $(\mathcal{S}, \mathcal{R})$ is finite and, for all $x, y, z \in \mathcal{S}$, the conditions $x\mathcal{R}y$ and $x\mathcal{R}z$ imply that there exists $w \in \mathcal{S}$ such that $y\mathcal{R}^T w$ and $z\mathcal{R}^T w$.*

The Church-Rosser finiteness property has been equivalently called *confluence* [1]. From now on, for the sake of making the text shorter, we shall use that word as well.

Levy and Low have shown that the set of reductions $\{\text{LOOP}, \text{IN0}, \text{OUT0}, \text{IN1}, \text{OUT1}\}$ is confluent [7], and Lemaic has shown that the set of reductions $\{\text{LOOP}, \text{INDICLIQUE}, \text{OUTDICLIQUE}\}$ is also confluent [6]. However, Lin and Jou in their article [8] did not investigate whether the confluence is preserved if one includes their three additional reductions (namely, $\text{PIE}(u, v)$, $\text{CORE}(u)$ and $\text{DOME}(u, v)$) in the family of reductions.

The digraph displayed in Figure 3 is a counter-example to the (false) claim that the family consisting of $\{\text{LOOP}, \text{IN0}, \text{OUT0}, \text{IN1}, \text{OUT1}, \text{PIE}, \text{CORE}, \text{DOME}\}$ is confluent.

Moreover, for practical purposes, when proving confluence, it is convenient to exclude the reductions subsumed by other reductions. For example, IN1 is subsumed by INDICLIQUE which means that if IN1 is applicable on a given vertex u , then INDICLIQUE is also applicable on u . Therefore, if $\{\text{INDICLIQUE}, \mathcal{R}\}$ is confluent, then $\{\text{IN1}, \text{INDICLIQUE}, \mathcal{R}\}$ is also confluent. The following proposition formalizes this property.

Proposition 1. *Let \mathcal{S} be a set and $\mathcal{R}_1, \mathcal{R}_2$ and \mathcal{R}_3 three relations on \mathcal{S} . If $\mathcal{R}_1 \subseteq \mathcal{R}_2$ and $\{\mathcal{R}_2, \mathcal{R}_3\}$ is confluent, then $\{\mathcal{R}_1, \mathcal{R}_2, \mathcal{R}_3\}$ is also confluent.*

So according to Propostion 1, proving that $\{\text{LOOP}, \text{INDICLIQUE}, \text{OUTDICLIQUE}, \text{PIE}\}$ is confluent implies that $\{\text{LOOP}, \text{IN0}, \text{OUT0}, \text{IN1}, \text{OUT1}, \text{PIE}, \text{CORE}, \text{INDICLIQUE}, \text{OUTDICLIQUE}\}$ is also confluent. Indeed, Lemaic has proved that IN1 and OUT1 are subsumed by INDICLIQUE and OUTDICLIQUE. The same is true for IN0 and OUT0 if we consider an empty set as a diclique. For the CORE reduction we can subsume it with INDICLIQUE/OUTDICLIQUE followed by LOOP. Indeed, if $u \in V$ is a core, then $N_G^-(u) \cup N_G^+(u) \cup \{u\}$ forms a diclique. In particular, $N_G^-(u) \cup N_G^+(u) = N_G^-(u) = N_G^+(u)$ forms a diclique, so we can apply INDICLIQUE(u) or OUTDICLIQUE(u). Hence, the neighbors of u will all have loops in $G \circ u$, which means that they have to be added to the minimum FVS which is equivalent to what CORE(u) should do, except that CORE(u) will isolate u and add its neighbors to the minimum FVS. On the other hand, INDICLIQUE(u) or OUTDICLIQUE(u) and LOOP will remove u and vertices in $N_G^-(u) \cup N_G^+(u)$ are added to the minimum FVS.

So in order to prove that $\{\text{LOOP}, \text{INDICLIQUE}, \text{OUTDICLIQUE}, \text{PIE}\}$ is confluent, we use Lemma 1.

Lemma 1. *Given a digraph $G = (V, E)$, an arc $(u, v) \in E$, $s \in N_G^+(v)$ and $p \in N_G^-(u)$, if (u, v) is acyclic in G , then (u, s) and (p, v) are also acyclic.*

We can now state the following Theorem for the confluence of the set of binary relations $\{\mathcal{R}_{\text{LOOP}}, \mathcal{R}_{\text{INDICLIQUE}}, \mathcal{R}_{\text{OUTDICLIQUE}}, \mathcal{R}_{\text{PIE}}\}$.

Theorem 2. *Let \mathcal{G} be the set of all digraphs and $\mathcal{R} = \mathcal{R}_{\text{LOOP}} \cup \mathcal{R}_{\text{INDICLIQUE}} \cup \mathcal{R}_{\text{OUTDICLIQUE}} \cup \mathcal{R}_{\text{PIE}}$ a binary relation on \mathcal{G} . Then, $(\mathcal{G}, \mathcal{R})$ is confluent.*

Proof. According to Theorem 1 it is enough to prove that $(\mathcal{G}, \mathcal{R})$ is finite and that for $G, G_1, G_2 \in \mathcal{G}$ if $(G, G_1) \in \mathcal{R}$ and $(G, G_2) \in \mathcal{R}$, then there exist $G' \in \mathcal{G}$ such that $(G_1, G') \in \mathcal{R}^T$ and $(G_2, G') \in \mathcal{R}^T$. Thanks to Proposition 1, it is sufficient to prove this only for \mathcal{R}_{PIE} and the other relations, since $\{\mathcal{R}_{\text{LOOP}}, \mathcal{R}_{\text{INDICLIQUE}}, \mathcal{R}_{\text{OUTDICLIQUE}}\}$ was proved to be confluent [6].

Let $G = (V, E) \in \mathcal{G}$ be a digraph, $x \in V$ and $(u, v) \in E$. PIE being a reduction, then its successive application are bounded by $|V|^2$, hence $(\mathcal{G}, \mathcal{R})$ is finite. Now, assume that PIE(u, v) is applicable.

If LOOP(x) is applicable, then it remains applicable after applying PIE(u, v), even if $x = u$ or $x = v$. So it is enough to consider $G' = G - x$. Otherwise, the two reductions can be applied in any order and $G' = (G - x) - (u, v)$.

Thanks to Lemma 1, if INDICLIQUE(x) (resp. OUTDICLIQUE(x)) is applied first, and $x = u$ or $x = v$, then PIE(p, v) or PIE(u, s) is applicable, $\forall p \in N_G^-(x = u)$ or $\forall s \in N_G^+(x = v)$ (the same goes if we first apply OUTDICLIQUE(x)). Otherwise, if we apply PIE(u, v) first, the applicability of INDICLIQUE(u) (resp. OUTDICLIQUE(u)) remains valid. In both cases, we can get the same digraph $G' = (G \circ x) - \{\bigcup_{p \in N_G^-(x)} (p, v)\} = (G - (u, v)) \circ x$ if $x = u$, or $G' = (G \circ x) - \{\bigcup_{s \in N_G^+(x)} (u, s)\} = (G - (u, v)) \circ x$ if $x = v$. Obviously, this remains true if $x \neq u$ and $x \neq v$, with $G' = (G \circ x) - (u, v) = (G - (u, v)) \circ x$.

Finally, it is easy to see that $(G - (u, v)) - (x, y) = (G - (x, y)) - (u, v)$, if PIE(x, y) is applicable for a given $(x, y) \in E$. Therefore, we can conclude that $(\mathcal{G}, \mathcal{R})$ is confluent. \square

4 Concluding remarks

In this extended abstract we focus on reductions for the minimum feedback vertex set problem, exploring their properties with an emphasis on confluence. By identifying a subset of reductions with confluence property and considering their adaptability, this work contributes to the understanding of graph reductions and their potential impact on algorithmic advancements. The exploration of the confluence property not only enhances our comprehension of algorithmic strategies but also opens avenues for parallelization and speed improvements in sequential algorithms. In future work, we will investigate the DOME reduction and explore how it can be modified so that it can be included in a confluent set of reductions considered in this extended abstract.

References

- [1] Franz Baader & Tobias Nipkow (1998): *Term rewriting and all that*. Cambridge University Press, USA, doi:10.5555/280474.
- [2] Alonzo Church & J. B. Rosser (1936): *Some properties of conversion*. *Transactions of the American Mathematical Society* 39(3), pp. 472–482, doi:10.1090/S0002-9947-1936-1501858-0.
- [3] Hartmut Ehrig & Barry K. Rosen (1980): *Parallelism and concurrency of graph manipulations*. *Theoretical Computer Science* 11(3), pp. 247–275, doi:10.1016/0304-3975(80)90016-X.
- [4] Michael R. Garey & David S. Johnson (1990): *Computers and Intractability; A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., USA, doi:10.5555/574848.
- [5] Richard M. Karp (1972): *Reducibility among Combinatorial Problems*, pp. 85–103. Springer US, Boston, MA, doi:10.1007/978-1-4684-2001-2_9.
- [6] Mile Lemaic (2008): *Markov-Chain-Based Heuristics for the Feedback Vertex Set Problem for Digraphs*. Ph.D. thesis, Universität zu Köln. Available at <https://kups.ub.uni-koeln.de/2547/>.
- [7] Hanoch Levy & David W Low (1988): *A contraction algorithm for finding small cycle cutsets*. *Journal of Algorithms* 9(4), pp. 470–493, doi:10.1016/0196-6774(88)90013-2.
- [8] Hen-Ming Lin & Jing-Yang Jou (2000): *On computing the minimum feedback vertex set of a directed graph by contraction operations*. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 19(3), pp. 295–307, doi:10.1109/43.833199.
- [9] Barry K Rosen (1976): *Correctness of parallel programs: The Church-Rosser approach*. *Theoretical Computer Science* 2(2), pp. 183–207, doi:10.1016/0304-3975(76)90032-3.
- [10] Ravi Sethi (1974): *Testing for the Church-Rosser Property*. *J. ACM* 21(4), p. 671–679, doi:10.1145/321850.321862.

Power Quotients of Plactic-like Monoids

Antoine Abram*

LACIM, Université du Québec à Montréal,
Montréal, Québec

abram.antoine@courrier.uqam.ca

Florent Hivert†

LISN, Université Paris-Saclay,
Orsay, France

florent.hivert@lri.fr

James D. Mitchell‡

School of Mathematics and Statistics, University of St Andrews,
St Andrews, Scotland

jdm3@st-andrews.ac.uk

Jean-Christophe Novelli§

LIGM, Université Gustave-Eiffel,
Marne-la-Vallée, France

novelli@univ-mlv.fr

Maria Tsalakou,

School of Mathematics and Statistics, University of St Andrews,
St Andrews, Scotland

mt200@st-andrews.ac.uk

In this paper we describe the quotients of several plactic-like monoids by the least congruences containing the relations $a^{\sigma(a)} = a$ with $\sigma(a) \geq 2$ for every generator a . The starting point for this description is the recent paper of Abram and Reutenauer about the so-called *stylic monoid* which happens to be the quotient of the plactic monoid by the relations $a^2 = a$ for every letter a . The plactic-like monoids considered are the plactic monoid itself, the Chinese monoid, and the sylvester monoid. In each case we describe: a set of normal forms, and the idempotents; and obtain formulae for their size.

1 Introduction

The plactic monoid $\mathbf{Plax}(\mathcal{A})$ over an ordered alphabet $(\mathcal{A}, <)$, can be defined as the quotient of the free monoid \mathcal{A}^* on \mathcal{A} by identifying words that produce the same Young tableau using Robinson-Schensted insertion algorithm [9, 10].

Knuth [6] found an explicit presentation as the quotient of \mathcal{A}^* by the relations:

$$acb = cab \quad \text{if } a \leq b < c \quad \text{and} \quad bac = bca \quad \text{if } a < b \leq c, \quad \text{with } a, b, c \in \mathcal{A}.$$

For more details see [7], or [8, Chapter 5]. Due to its link with symmetric functions and representation theory, the plactic monoid is a central object in algebraic combinatorics that has been widely studied in the literature.

Other monoids, whose relations are delineated in terms of insertion algorithms on certain combinatorial objects, are often referred to as "plactic-like" monoids. They exhibit a rich combinatorial structure and have applications in several topics including geometry and representation theory.

Among others, this family contains the Chinese monoid $\mathbf{Ch}(\mathcal{A})$ [3], that has applications on Hecke atoms and the Bruhat order (see [4]), and the sylvester monoid $\mathbf{Sylv}(\mathcal{A})$ [5], which is related to the associahedra and the Loday-Ronco algebra of trees.

* Antoine Abram was partially supported by CRSNG BESC D grant,

† Florent Hivert was partially supported by OpenDreamKit Horizon 2020 ERI (#676541)

‡ James D. Mitchell was partially supported by OpenDreamKit Horizon 2020 ERI (#676541)

§ Jean-Christophe Novelli was partially supported by the ANR program CARPLO.

Recently, Reutenauer and the first author [1] discovered that the quotient of the plactic monoid by the relations $a^2 = a$, for every letter a , has several interesting properties. Inspired by their results, we investigate more general finite quotients of plactic-like monoids. For such monoid $\mathbf{M}(\mathcal{A})$ defined over an alphabet \mathcal{A} and a function $\sigma : \mathcal{A} \rightarrow \mathbb{N}_{\geq 2}$, we study the quotients $\mathbf{M}(\mathcal{A}, \sigma)$ by the relations $a^{\sigma(a)} = a$ for every $a \in \mathcal{A}$. It turns out that the monoids are of two different non-disjoint types. For the first type, that includes the plactic, Chinese, and hypoplactic monoids, $\mathbf{M}(\mathcal{A}, \sigma)$ can be naturally embedded in the cartesian product of $\mathbf{M}(\mathcal{A}, 2)$ with the commutative monoid $\mathbf{Com}(\mathcal{A}, \sigma)$. The second type, that includes the hypoplactic and the sylvester monoids, have words with a particular property in every equivalence class, that provides a set of normal forms. In both types, $\mathbf{M}(\mathcal{A}, 2)$ plays an important role in the structure of $\mathbf{M}(\mathcal{A}, \sigma)$ for any σ . In addition, $\mathbf{M}(\mathcal{A}, 2)$ has a rich combinatorial structure usually related to the one of $\mathbf{M}(\mathcal{A})$.

For the first type, our knowledge of the stylic monoid helps us in understanding the structure of $\mathbf{Plax}(\mathcal{A}, \sigma)$. We then consider another example, namely the Chinese monoid, by studying its 2-quotient, which involves rich combinatorial objects, and transpose this to the study of its general σ -quotient.

For the second type, we focus on the sylvester monoid and more particularly on its 2-quotient.

2 Words, Monoids and σ -Quotients

Let \mathcal{A} be a finite ordered alphabet, \mathcal{A}^* the free monoid over \mathcal{A} , and $\mathbf{Com}(\mathcal{A})$ the free commutative monoid over \mathcal{A} .

For $W \in \mathcal{A}^*$, the *content* of W is the set of distinct letters occurring in W , and is denoted by $\text{cont}(W)$. We call the natural surjection $\text{ev} : \mathcal{A}^* \rightarrow \mathbf{Com}(\mathcal{A})$ the *evaluation map* and say $\text{ev}(W)$ is the *evaluation* of the word W , for all $W \in \mathcal{A}^*$. For a word $W = w_1 \cdots w_n \in \mathcal{A}^*$, an *inflation* of W is any word of the form $w_1^{\varepsilon_1} \cdots w_n^{\varepsilon_n} \in \mathcal{A}^*$ for some $\varepsilon_1, \dots, \varepsilon_n \in \mathbb{N}_{\geq 1}$.

Let $\mathbf{M}(\mathcal{A})$ be a monoid defined by the presentation $\langle \mathcal{A} | \mathcal{R} \rangle$ for some alphabet \mathcal{A} and some relations $\mathcal{R} \subseteq \mathcal{A}^* \times \mathcal{A}^*$. If $U, V \in \mathcal{A}^*$ have the same image under the natural surjective homomorphism $\pi_{\mathbf{M}} : \mathcal{A}^* \rightarrow \mathbf{M}$, we say that U and V are *equivalent* in $\mathbf{M}(\mathcal{A})$, and we write $U \equiv_{\mathbf{M}(\mathcal{A})} V$.

If $\mathcal{R} \subseteq \mathcal{A}^* \times \mathcal{A}^*$ has the property that $\text{ev}(U) = \text{ev}(V)$ for all $(U, V) \in \mathcal{R}$, then we say that the presentation $\langle \mathcal{A} | \mathcal{R} \rangle$ is *evaluation-preserving*, and, by extension, that $\mathbf{M}(\mathcal{A})$ is an *evaluation-preserving monoid*.

Definition 2.1. *Let $\mathbf{M}(\mathcal{A})$ be an evaluation preserving monoid and $\sigma : \mathcal{A} \rightarrow \mathbb{N}_{\geq 2}$. We define $\mathbf{M}(\mathcal{A}, \sigma)$ to be the quotient of $\mathbf{M}(\mathcal{A})$ obtained by adding the extra relations $(a^{\sigma(a)}, a)$ for every $a \in \mathcal{A}$ to the presentation $\langle \mathcal{A} | \mathcal{R} \rangle$. If $\sigma : \mathcal{A} \rightarrow \mathbb{N}_{\geq 2}$ is constant with value n , then we write $\mathbf{M}(\mathcal{A}, n)$ instead of $\mathbf{M}(\mathcal{A}, \sigma)$.*

Two types of monoids arise from the study of these quotients. These two types are not mutually exclusive; the hypoplactic monoid is of both types.

3 Monoids of Type 1

Let $\mathbf{M}(\mathcal{A})$ be an evaluation-preserving monoid. For any $\sigma : \mathcal{A} \rightarrow \mathbb{N}_{\geq 2}$, let $\theta : \mathbf{M}(\mathcal{A}, \sigma) \rightarrow \mathbf{M}(\mathcal{A}, 2)$ and $\text{ev}_{\sigma} : \mathbf{M}(\mathcal{A}, \sigma) \rightarrow \mathbf{Com}(\mathcal{A}, \sigma)$ be the natural surjective morphisms. We let ϕ_{σ} be the product map $\theta \times \text{ev}_{\sigma}$.

Definition 3.1. *An evaluation-preserving monoid $\mathbf{M}(\mathcal{A})$ is of type 1 if for any $\sigma : \mathcal{A} \rightarrow \mathbb{N}_{\geq 2}$, ϕ_{σ} is an embedding.*

3.1 The σ -Plactic Monoids

In [1], the monoid which we define here as $\mathbf{Plax}(\mathcal{A}, 2)$ was introduced as the stylic monoid. We recall some necessary definitions and refer the reader to [1] for more details.

An N -tableau is a semi-standard tableau such that its rows are strictly increasing; and each row is contained in the row underneath. The *row reading* of a tableau T is the word $R(T)$ obtained by reading each row from left to right and top to bottom. The N -tableaux have an insertion algorithm, denoted $T \leftarrow W$, called the N -insertion that is similar to the Robinson-Schensted insertion in Young tableaux.

Using these N -tableaux and their insertion algorithm, one can prove that the plactic monoid is of type 1.

Given $(T, e) \in \mathbf{Plax}(\mathcal{A}, \sigma) \times \mathbf{Com}(\mathcal{A}, \sigma)$ such that $\text{cont}(T) = \text{cont}(e)$, we define the *row reading* $R_\sigma(T, e) \in \mathcal{A}^*$ via the following algorithm starting with $R(T)$. For every $a \in \text{cont}(T)$, if α is the number of occurrences of a in $R(T)$ and β is the exponent of a in e , then we replace the last occurrence of a in $R(T)$ by a^γ where $\gamma \in \mathbb{N}_{\geq 0}$ is the least value such that $\gamma + \alpha = \beta \pmod{\sigma(a) - 1}$. For example, if $\sigma(x) = 4$ for all $x \in \mathcal{A}$ then

$$R_\sigma \left(\begin{array}{|c|c|} \hline b & \\ \hline a & b \\ \hline \end{array}, a^1 b^1 \right) = bab^3 \quad \text{and} \quad R_\sigma \left(\begin{array}{|c|c|c|} \hline b & c & \\ \hline a & b & c \\ \hline \end{array}, a^3 b^2 c^1 \right) = bca^3 bc^3.$$

Such a row reading $R_\sigma(T, e)$ is a preimage of (T, e) under ϕ ; and these row readings constitute a set of normal forms for $\mathbf{Plax}(\mathcal{A}, \sigma)$.

3.2 The σ -Chinese Monoids

The *Chinese monoid* $\mathbf{Ch}(\mathcal{A})$ is defined by the presentation with generating set \mathcal{A} and relations [3]: for $a, b, c \in \mathcal{A}$,

$$cba = cab = bca \quad \text{if } a < b < c, \quad aba = baa, \quad bba = bab \quad \text{if } a < b.$$

As for the plactic monoid, the elements of this monoid can be represented using a combinatorial object, the *Chinese staircase*, and its insertion algorithm. It has a particular *row reading* (defined in [3]) which is the shortlex normal form of its class. For a chinese staircase S , one can also associate a Dyck path $\mathbf{Dyck}(S)$ of length $2|\text{cont}(S)|$; see Fig. 1.

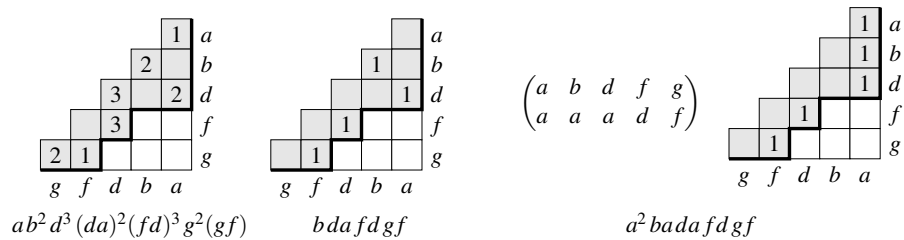


Figure 1: The left diagram is a Chinese staircase with the associated Dyck path, and its reading below; the middle its 2-Chinese staircase and reading; and the right its 2-Chinese function, its reading and its equivalent staircase.

The following result gave us a nice description of the $\mathbf{Ch}(\mathcal{A}, 2)$ -equivalence.

Theorem 3.2. *If S and T are Chinese staircases, then $R(S) \equiv_{\mathbf{Ch}(\mathcal{A}, 2)} R(T)$ if and only if $\text{cont}(R(S)) = \text{cont}(R(T))$ and $\mathbf{Dyck}(S) = \mathbf{Dyck}(T)$.*

Using this, we define two different combinatorial objects, the 2-Chinese staircases and the 2-Chinese functions, that both represent $\mathbf{Ch}(\mathcal{A}, 2)$ -classes, in order to have a better understanding of $\mathbf{Ch}(\mathcal{A}, 2)$.

Let $\mathcal{B} \subseteq \mathcal{A}$ and D be a Dyck path of length $2|\mathcal{B}|$. The 2-Chinese staircase associated to \mathcal{B} and D is the Chinese staircase S such that $\mathbf{Dyck}(S) = D$, that has 1s in all of the peaks of D and in every boxes of the diagonal having an empty hook.

If S is a 2-Chinese staircase, then the *row reading* $R(S)$ of S is simply the row-reading of S as a Chinese staircase which is also the shortlex normal form of its $\mathbf{Ch}(\mathcal{A}, 2)$ -class; see Fig. 1.

A 2-Chinese function on \mathcal{A} is a function $\begin{smallmatrix} \textcircled{a} \\ \textcircled{b} \end{smallmatrix} : \mathcal{B} \rightarrow \mathcal{B}$ for some $\mathcal{B} \subseteq \mathcal{A}$ which for all $x, y \in \mathcal{B}$ satisfies: $x \leq y$ implies $\begin{smallmatrix} \textcircled{a} \\ \textcircled{b} \end{smallmatrix}(x) \leq \begin{smallmatrix} \textcircled{a} \\ \textcircled{b} \end{smallmatrix}(y)$; and $\begin{smallmatrix} \textcircled{a} \\ \textcircled{b} \end{smallmatrix}(x) \leq x$. We denote by $\begin{smallmatrix} \square \\ \square \end{smallmatrix}$ the unique function whose domain is empty.

The *insertion* of $y \in \mathcal{A}$ in a 2-Chinese function $\begin{smallmatrix} \textcircled{a} \\ \textcircled{b} \end{smallmatrix}$ is the function $\begin{smallmatrix} \textcircled{a} \\ \textcircled{b} \end{smallmatrix} \leftarrow y$ whose domain is $\text{dom}(\begin{smallmatrix} \textcircled{a} \\ \textcircled{b} \end{smallmatrix}) \cup \{y\}$ and the image of any x is given by $(\begin{smallmatrix} \textcircled{a} \\ \textcircled{b} \end{smallmatrix} \leftarrow y)(x) := \min\{y\} \cup \begin{smallmatrix} \textcircled{a} \\ \textcircled{b} \end{smallmatrix}(\hat{x})$, for $\hat{x} = \min\{z \in \text{dom}(\begin{smallmatrix} \textcircled{a} \\ \textcircled{b} \end{smallmatrix}) \mid z \geq x\}$. It is routine to verify that $\begin{smallmatrix} \textcircled{a} \\ \textcircled{b} \end{smallmatrix} \leftarrow y$ is also a 2-Chinese function, as seen in the following example:

$$\begin{pmatrix} a & b & c & e & f \\ a & b & c & c & f \end{pmatrix} \leftarrow g = \begin{pmatrix} a & b & c & e & f & g \\ a & b & c & c & f & g \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} a & b & c & e & f \\ a & b & c & c & f \end{pmatrix} \leftarrow d = \begin{pmatrix} a & b & c & d & e & f \\ a & b & c & c & c & d \end{pmatrix}$$

We define the reading word of $\begin{smallmatrix} \textcircled{a} \\ \textcircled{b} \end{smallmatrix}$ to be $R(\begin{smallmatrix} \textcircled{a} \\ \textcircled{b} \end{smallmatrix}) := a_1 \begin{smallmatrix} \textcircled{a} \\ \textcircled{b} \end{smallmatrix}(a_1) \cdots a_n \begin{smallmatrix} \textcircled{a} \\ \textcircled{b} \end{smallmatrix}(a_n)$, where $\text{dom}(\begin{smallmatrix} \textcircled{a} \\ \textcircled{b} \end{smallmatrix}) = \{a_1 < a_2 < \cdots < a_n\}$.

Using properties of both combinatorial representatives of $\mathbf{Ch}(\mathcal{A}, 2)$, we proved that the Chinese monoid is of type 1.

These objects also allowed us prove that $\mathbf{Ch}(\mathcal{A}, 2)$ is \mathcal{J} -trivial. Unlike the stylic monoid, its \mathcal{J} -order is surprisingly not graded.

Similar to the plactic case, we define a set of normal forms the following way: for any $(\begin{smallmatrix} \textcircled{a} \\ \textcircled{b} \end{smallmatrix}, e) \in \mathbf{Ch}(\mathcal{A}, 2) \times \mathbf{Com}(\mathcal{A}, \sigma)$ such that $\text{cont}(\begin{smallmatrix} \textcircled{a} \\ \textcircled{b} \end{smallmatrix}) = \text{cont}(e)$, we define the *row reading* $R_\sigma(\begin{smallmatrix} \textcircled{a} \\ \textcircled{b} \end{smallmatrix}, e)$ of $(\begin{smallmatrix} \textcircled{a} \\ \textcircled{b} \end{smallmatrix}, e)$ by inflating $R(\begin{smallmatrix} \textcircled{a} \\ \textcircled{b} \end{smallmatrix})$ putting a suitable exponent on the last occurrence of each letter. For example, if $\mathcal{A} = \{a, b, c\}$ and σ is constant with value 4, then $R_\sigma \left[\begin{pmatrix} a & b & c & e & f \\ a & b & c & c & f \end{pmatrix}, a^2 b^3 c^2 e f^3 \right] = a^2 b^3 c^2 e c^3 f^3$.

3.3 Cardinality and Idempotents of Monoids of Type 1

From the definition of a type 1 monoid, one only has to know the combinatorial structure of the 2-quotient in order to compute the cardinality of the σ -quotient for any σ and to find its idempotents.

Theorem 3.3. *Let $\sigma : \mathcal{A} \rightarrow \mathbb{N}_{\geq 2}$ be arbitrary. Then the cardinality of $\mathbf{M}(\mathcal{A}, \sigma)$ is given by*

$$|\mathbf{M}(\mathcal{A}, \sigma)| = \sum_{\mathcal{B} \subseteq \mathcal{A}} \left(s_{|\mathcal{B}|} \prod_{b \in \mathcal{B}} (\sigma(b) - 1) \right) \quad (3.1)$$

where s_k is:

- (i) the k -th Bell number if $\mathbf{M} = \mathbf{Plax}$;
- (ii) the k -th Catalan number if $\mathbf{M} = \mathbf{Ch}$;

In particular, $|\mathbf{M}(\mathcal{A}, 2)| = \sum_{k=0}^n \binom{n}{k} s_k$ is the binomial transform of the sequence s_k in both cases.

Proposition 3.4. *The monoids $\mathbf{Plax}(\mathcal{A}, \sigma)$ and $\mathbf{Ch}(\mathcal{A}, \sigma)$ contain exactly $2^{|\mathcal{A}|}$ idempotents, one for each $\mathcal{B} = \{b_1 < b_2 < \cdots < b_k\} \subseteq \mathcal{A}$. These elements are inflations I of, respectively, $\min_{\mathcal{J}}(\mathbf{Plax}(\mathcal{B}, 2))$ and $\min_{\mathcal{J}}(\mathbf{Ch}(\mathcal{B}, 2))$, such that $\text{ev}_\sigma(I) = \prod_{i=1}^k b_i^{\sigma(b_i)-1}$.*

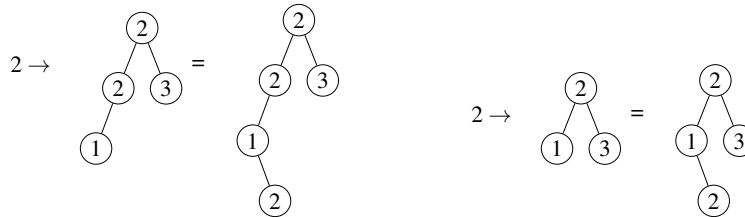


Figure 2: On the left, the insertion of 2 in the binary search tree that has reading word 3122 resulting to the tree having reading word 32122. On the right, the 2-sylvester insertion in its 2-reduced binary search tree.

4 Monoids of Type 2

Let $\mathbf{M}(\mathcal{A})$ be an evaluation-preserving monoid. We say that $W \in \mathcal{A}^*$ is a *gathered element* if whenever $W' \in \mathcal{A}^*$ is such that $W' \equiv_{\mathbf{M}(\mathcal{A})} W$, then $W' = U'a^2V'$ implies $W = Ua^2V$, for some $U, V \in \mathcal{A}^*$ where U and U' have the same number of a 's.

If an $\equiv_{\mathbf{M}(\mathcal{A})}$ -class contains a gathered element, we refer to the lexicographically largest gathered word $G(W)$ in the class as the *canonical gathered element*.

We define the (a, i) -*expansion* of $W \in \mathcal{A}^*$ to be the word obtained from W where the i -th occurrence of a is duplicated.

Definition 4.1. *An evaluation-preserving monoid $\mathbf{M}(\mathcal{A})$ is of type 2 if:*

- (a) *each $\equiv_{\mathbf{M}(\mathcal{A})}$ -class has a gathered element; and*
- (b) *the (a, i) -expansion of $G(W)$ equals the canonical gathered element of the (a, i) -expansion of W .*

If $W \in \mathcal{A}^*$, then we define $G_\sigma(W)$, the σ -*reduced word* of W , to be the word obtained from $G(W)$ by repeatedly replacing any factor $a^{\sigma(a)}$ of $G(W)$ by a , until there are no such factors remaining. The set of σ -reduced word constitutes a set of normal forms for $\mathbf{M}(\mathcal{A}, \sigma)$.

4.1 The σ -Sylvester Monoids

The *sylvester monoid* [5] is the quotient of \mathcal{A}^* by the following infinite set of relations:

$$acWb = caWb \quad \text{if } a \leq b < c, \quad \text{for all } W \in \mathcal{A}^*.$$

A *binary search tree* $T = (L, r, R)$ is a binary tree labelled by \mathcal{A} such that the label of each node is greater than or equal to all labels in its left subtree L and strictly smaller than all labels in its right subtree R . Binary search trees are endowed with a well-known left insertion $a \rightarrow T$ of letters $a \in \mathcal{A}$; see [5]. Given a binary search tree T , denote $R(T)$ its *right to left postfix reading*. This word is the lexicographic largest word in its $\equiv_{\text{Sylv}(\mathcal{A})}$ -class [5, Proposition 14].

Using this set of normal forms, one can prove that the sylvester monoid is of type 2.

The 2-reduced words are the readings of the binary trees where parents have different label than their left children. If the number of nodes i , and the number of nodes k having at least one ancestor with the same label are fixed, then, thanks to [2] and a standard involution among trees, *recursive reversal of the left branch subtrees*, one can prove that the number of such trees is $B_{i-k-1, k}$, an element of the Borel triangle (A234950 in [11]).

Theorem 4.2. *Let \mathcal{A} be an alphabet of size $n \in \mathbb{N}$. Then*

$$|\mathbf{Sylv}(\mathcal{A}, 2)| = 1 + \sum_{i=1}^{2n-1} \sum_{k=0}^{\lfloor i/2 \rfloor} B_{i-k-1,k} \binom{n}{i-k}. \quad (4.1)$$

One can easily adapt the enumeration formula for $\mathbf{Sylv}(\mathcal{A}, p)$ but, not being of type 1 makes the general formula significantly more complicated.

Proposition 4.3. *The monoid $\mathbf{Sylv}(\mathcal{A}, 2)$ contains exactly $\sum_{k=0}^n \binom{n}{k} S_k$ idempotents, where S_n is the Schröder numbers (A006318 in [11]), shifted by one: $S_0 = S_1 = 1$, $S_2 = 2$, $S_3 = 6$, etc.*

These idempotents are the postfix reading of 2-reduced binary search trees T such that, for all $x \in \text{cont}(T)$, the deepest node labelled x in T does not have any left subtree; see the rightmost tree of Fig. 2.

Using these trees, one can describe the idempotents of $\mathbf{Sylv}(\mathcal{A}, \sigma)$ for arbitrary σ but the enumeration formula can only be easily adapted for σ constant.

References

- [1] A. Abram & C. Reutenauer (2022): *The stylic monoid*. *Semigroup Forum* 105(1), pp. 1–45, doi:10.1007/s00233-022-10285-3. arXiv:2106.06556.
- [2] Yue Cai & Catherine Yan (2019): *Counting with Borel’s triangle*. *Discrete Math.* 342(2), pp. 529–539, doi:10.1016/j.disc.2018.10.031. arXiv:1804.01597.
- [3] Julien Cassaigne, Marc Espie, Daniel Krob, Jean-Christophe Novelli & Florent Hivert (2001): *The Chinese monoid*. *Internat. J. Algebra Comput.* 11(3), pp. 301–334, doi:10.1142/S0218196701000425. Available at <https://hal.science/hal-00018547/document>.
- [4] Zachary Hamaker, Eric Marberg & Brendan Pawlowski (2017): *Involution words II: braid relations and atomic structures*. *J. Algebraic Combin.* 45(3), pp. 701–743, doi:10.1007/s10801-016-0722-6. arXiv:1601.02269.
- [5] F. Hivert, J.-C. Novelli & J.-Y. Thibon (2005): *The algebra of binary search trees*. *Theoret. Comput. Sci.* 339(1), pp. 129–165, doi:10.1016/j.tcs.2005.01.012. arXiv:math/0401089.
- [6] Donald E. Knuth (1970): *Permutations, matrices, and generalized Young tableaux*. *Pacific J. Math.* 34, pp. 709–727. Available at <http://projecteuclid.org/euclid.pjm/1102971948>.
- [7] Alain Lascoux & Marcel-P. Schützenberger (1981): *Le monoïde plaxique*. In: *Noncommutative structures in algebra and geometric combinatorics (Naples, 1978)*, Quad. “Ricerca Sci.” 109, CNR, Rome, pp. 129–156. Available at <http://igm.univ-mlv.fr/~berstel/Mps/Travaux/A/1981-1PlaxiqueNaples.pdf>.
- [8] M. Lothaire (2002): *Algebraic combinatorics on words*. *Encyclopedia of Mathematics and its Applications* 90, Cambridge University Press, Cambridge, doi:10.1017/CBO9781107326019. Available at <http://tomlr.free.fr/Math%E9matiques/Fichiers%20Claude/Auteurs/aaaDivers/Lothaire%20-%20Algebraic%20Combinatorics%200n%20Words.pdf>.
- [9] G. de B. Robinson (1938): *On the Representations of the Symmetric Group*. *Amer. J. Math.* 60(3), pp. 745–760, doi:10.2307/2371609.
- [10] C. Schensted (1961): *Longest increasing and decreasing subsequences*. *Canadian J. Math.* 13, pp. 179–191, doi:10.4153/CJM-1961-015-3. Available at <https://sites.math.washington.edu/~billey/classes/561.fall.2019/articles/schensted.1961.pdf>.
- [11] Neil J. A. Sloane & The OEIS Foundation Inc. (2024): *The on-line encyclopedia of integer sequences*. Available at <https://oeis.org>.

Self-descriptive Sequences directed by two Periodic Sequences

Shigeki Akiyama

Institute of Mathematics, University of Tsukuba
1-1-1 Tennodai, Tsukuba, Ibaraki, 305-8571 Japan
akiyama@math.tsukuba.ac.jp

Irène Marcovici

Univ Rouen Normandie, CNRS,
Normandie Univ LMRS UMR 6085,
F-76000 Rouen, France
irene.marcovici@univ-rouen.fr

Damien Jamet

Univ. Lorraine, Loria, UMR 7503
Vandœuvre-lès-Nancy, F-54506, France
damien.jamet@loria.fr

Mai-Linh Trần Công

École Normale Supérieure de Lyon
15 parvis René Descartes, F-69342 Lyon, France
mai-linh.tran_cong@ens-lyon.fr

1 Introduction

A self-descriptive sequence $(u_n)_{n \in \mathbb{N}}$ is an infinite concatenation of finite powers of a letter (usually called runs) $(w_n)_{n \in \mathbb{N}}$ such that $|w_n| = u_n$ where $|x|$ denotes the length of the finite word x . The best known self-descriptive sequence is certainly the Oldenburger word $\mathbb{O}_{1,2} = (k_n)_{n \in \mathbb{N}}$ [9, 8] defined by $u_0 = 1$, $w_{2n} = 1^{u_{2n}}$ and $w_{2n+1} = 2^{u_{2n+1}}$ for all $n \in \mathbb{N}$. Until recently, the Oldenburger word was still called the Kolakoski word in reference to [8], but actually, it first appeared in [9].

The Oldenburger word is a special case of a self-descriptive sequence. Indeed, the run w_n is entirely determined by knowledge of its index n : its length is equal to u_n and its single letter is determined by the parity of n .

In [3], the authors focus on a larger family of self-descriptive sequences where the w_n 's are determined not only by their index n but also by another sequence, namely the directing sequence of u . In practice, given a sequence $t = (t_n)_{n \in \mathbb{N}}$ on the alphabet $\mathcal{A} \in \{1, 2, \dots\}$, the sequence directed by t is the sequence u defined by: $u = t_0^{u_0} t_1^{u_1} \dots t_n^{u_n} \dots$. For example, the Oldenburger word $\mathbb{O}_{1,2} = 1^{u_0} 2^{u_1} 1^{u_2} 2^{u_4} \dots$ is directed by the sequence $t = (12)^\omega$.

One of the most fascinating questions about the sequence $\mathbb{O}_{1,2}$ concerns the existence and possible value of the frequencies of occurrences of each of its letters [7]: *Do the letters 1 and 2 have frequencies of occurrences f_1 and f_2 in $\mathbb{O}_{1,2}$? If so, does $f_1 = f_2 = \frac{1}{2}$?* Recall that the frequency of occurrences of the letter a in the sequence u is the limit, when n tends to $+\infty$, of the average number of a in the prefix $u_0 \dots u_{n-1}$ of u .

The notion of self-descriptive sequence is related to that of differentiable word and smooth word [2, 4]. A sequence over \mathcal{A} is differentiable over \mathcal{A} if it is the infinite concatenation of runs whose lengths have values in \mathcal{A} . More precisely, a sequence $(u_n)_{n \in \mathbb{N}}$ over a finite alphabet $\mathcal{A} \subset \mathbb{N}$ is differentiable if there exist two sequences $(x_n)_{n \in \mathbb{N}}$ and $(\alpha_n)_{n \in \mathbb{N}}$ over \mathcal{A} , such that $u = x_0^{\alpha_0} x_1^{\alpha_1} x_2^{\alpha_2} \dots$ with $x_n \neq x_{n+1}$ and $\alpha_n \neq 0$ for all $n \in \mathbb{N}$. The sequence $(\alpha_n)_{n \in \mathbb{N}}$ is the derivative sequence of u . Finally, the sequence u is smooth if it is infinitely differentiable.

The sequence $\mathbb{O}_{1,2}$ is a fixed point for differentiation. It is self-descriptive, differentiable and smooth over $\{1, 2\}$. As with the sequence $\mathbb{O}_{1,2}$, the question of the existence of frequencies of occurrences and their values in smooth words on the alphabet $\{1, 2\}$ is still open. The first significant result on this

question is due to V. Chvátal [6]: for n large enough,

$$0.49916 \leq \frac{|k_0 \cdots k_{n-1}|_1}{n} \leq 0.50084,$$

where $|w|_1$ denotes the number of occurrences of the letter 1 in w . These bounds have been slightly improved by M. Rao [10] using a method quite similar to Chvátal's but with greater computing power:

$$0.49992 \leq \frac{|k_0 \cdots k_{n-1}|_1}{n} \leq 0.50008.$$

This is the best currently known bound for $\mathbb{O}_{1,2}$ and for the set of smooth words on the alphabet $\{1, 2\}$. In other words, neither the existence nor the values of the frequencies of occurrences are known for any smooth word on the $\{1, 2\}$ alphabet. On the other hand, over the $\{a, b\}$ alphabets where a and b have the same parity, it is possible to determine the frequencies of certain smooth words. This is the case at least for the Oldenburger word $\mathbb{O}_{1,3}$ (resp. $\mathbb{O}_{3,1}$) defined on the alphabet $\{1, 3\}$ directed by $(13)^\omega$ (resp. by $(31)^\omega$) [1] and for the extreme smooth words (in the sense of lexicographic order) [5].

Since the work of V. Chvátal [6] and M. Rao [10], it is reasonable to expect that the frequencies of occurrences of each letter in $\mathbb{O}_{1,2}$ are equal to $\frac{1}{2}$. In other words, it is reasonable to assume that the frequencies of occurrence in $\mathbb{O}_{1,2}$ and in its directing sequence are identical.

As far as we know, none of the works on smooth words or on the Oldenburger word has shown the existence or the non-existence of frequencies of occurrences in a non-trivial deterministic self-descriptive sequence over the alphabet $\{1, 2\}$.

Similarly, none of these works has proved the existence of a non-trivial deterministic self-descriptive sequence that shares (resp. does not share) its frequencies of occurrences with its directing sequence.

In the present work, we exhibit a class of self-descriptive sequences that can be explicitly computed and whose frequencies are known. In particular, as a corollary of our main result, we prove that the sequence introduced in [3] has the expected frequencies of occurrences.

2 Definitions and basic notions

Let \mathcal{A} be a finite alphabet. The set of finite words over \mathcal{A} is denoted by \mathcal{A}^* . If $w = w_0 \cdots w_k \in \mathcal{A}^*$ is a finite word over the alphabet \mathcal{A} with $w_i \in \mathcal{A}$ for $i = 0, 1, \dots, k$. Let $|w|$ stand for the **length** of w , that is the number of letters occurring in w . If $w = w_0 \cdots w_k$, then $|w| = k + 1$. In particular, $|\varepsilon| = 0$. Let $w \in \mathcal{A}^*$ and let $a \in \mathcal{A}$. We set $|w|_a = \#\{i \in \{0, 1, \dots, |w| - 1\} \mid w_i = a\}$.

Definition 1 (Self-descriptive sequence). Let $\mathcal{A} \subset \mathbb{N}^*$ be a finite alphabet. The infinite sequence $u = (u_n)_{n \in \mathbb{N}} \in \mathcal{A}^{\mathbb{N}}$ is said to be **self-descriptive** if there exists a sequence $\delta = (\delta_n)_{n \in \mathbb{N}}$ over \mathcal{A} such that

$$u = \delta_0^{u_0} \delta_1^{u_1} \delta_2^{u_2} \cdots \delta_n^{u_n} \cdots \quad (1)$$

The sequence δ is called the **directing sequence** of u and one says that u is directed by x .

In other words, the sequence u is self-descriptive if it is the concatenation of runs of size u_0, u_1, u_2, \dots respectively. Note that in the definition of self-descriptive sequences, unlike that of differentiable or smooth words, it is not necessary that $x_n \neq x_{n+1}$. Furthermore, if $0 \notin \mathcal{A}$, then the sequence $(x_n)_{n \in \mathbb{N}}$ is entirely determined by u . In other words, there exists a canonical bijection between sequences over \mathcal{A} and self-descriptive sequences over \mathcal{A} .

Let $u = \delta_0^{u_0} \delta_1^{u_1} \delta_2^{u_2} \dots \delta_n^{u_n} \dots$ be a self-descriptive sequence over the alphabet $\{1, 2\}$. Let $(m_k)_{k \in \mathbb{N}}$ (resp. $(n_k)_{k \in \mathbb{N}}$) be the increasing sequence over \mathbb{N} such that $u_i = 1$ (resp. $u_i = 2$) if and only if there exists $k \in \mathbb{N}$ such that $i = m_k$ (resp. $i = n_k$). In other words, $(m_k)_{k \in \mathbb{N}}$ (resp. $(n_k)_{k \in \mathbb{N}}$) is exactly the ordered sequences of the indices where u is equal to 1 (resp. 2). Let $T_1 = (\delta_{m_k})_{k \in \{1, 2\}^{\mathbb{N}}}$ and $T_2 = (\delta_{n_k})_{k \in \{1, 2\}^{\mathbb{N}}}$.

In the present work, since we are only interested in frequencies of letters, we assume, without loss of generality, that $u_0 = u_1 = 2$. The sequence u and its directing sequence δ are then computable from T_1 and T_2 as follows:

```
def OK(T1, T2):
    u = [2, 2]
    delta = [2]
    k = 1
    while len(T1) > 0 and len(T2) > 0:
        if u[k] == 1:
            c = T1.pop(0)
            u += [c]
        else:
            c = T2.pop(0)
            u += [c] * u[k]
        delta += [1]
        k += 1
    return u, delta
```

Program 1: Python function computing u and δ from T_1 and T_2 .

One then says that u is also directed by sequences T_1 and T_2 . For instance, if $T_1 = 121\dots$ and $T_2 = 12\dots$, then

$$u = \frac{22}{2} \cdot \frac{11}{2} \cdot \frac{11}{1} \frac{2}{1} \cdot \frac{1}{1} \frac{22}{2} \dots \quad (2)$$

$$= 2^2 \cdot 1^2 \cdot 1^1 2^1 \cdot 1^1 2^2 \dots \quad (3)$$

$$= \frac{22}{2} \cdot \frac{11}{w_0} \cdot \frac{12}{w_1} \cdot \frac{122}{w_2} \dots = 22 \cdot w_0 \cdot w_1 \cdot w_2 \dots \quad (4)$$

with $w_i \in \{a, b, c, d\}^*$, for $i \in \mathbb{N}$, $a = 1$, $b = 2$, $c = 1$ and $d = 1$.

3 Main result

The main result of the present work is:

Theorem 1 (Main result). *Let $u \in \{1, 2\}^{\mathbb{N}}$ be a sequence over $\{1, 2\}$ directed by two periodic sequences $T_1 = (x_1)^\omega \in \{1, 2\}^{\mathbb{N}}$ and $T_2 = (x_2)^\omega \in \{1, 2\}^{\mathbb{N}}$, with $x_1, x_2 \in \{1, 2\}^*$. Let $p_1 = \frac{|x_1|_1}{|x_1|}$ and $q_2 = \frac{|x_2|_2}{|x_2|}$. One has*

$$f_1 := \lim_{n \rightarrow \infty} \frac{|u_0 \dots u_{n-1}|_1}{n} = \frac{(1 - q_2)(p_1 + 2q_2 + \sqrt{\Delta})}{2 + \sqrt{\Delta} - p_1}$$

with $\Delta = (p_1 + 2q_2)^2 - 8(p_1 + q_2 - 1)$.

If $\delta = (\delta_n)_{n \in \mathbb{N}} \in \{1, 2\}$ is directing u , then

$$\lim_{n \rightarrow \infty} \frac{|\delta_0 \dots \delta_{n-1}|_1}{n} = p_1 f_1 + p_2 (1 - f_1).$$

Sketch of proof. Let us recode T_1, T_2 and u over $\{a, b, c, d\}$ as follows: rewrite T_1 (resp. T_2) as the image of T_1 (resp. T_2) by the morphism $1 \mapsto a, 2 \mapsto b$ (resp. $1 \mapsto c, 2 \mapsto d$). In u , let us substitute a (resp. b) for isolated 1's (resp. isolated 2's) and cc (resp. dd) for double 1's (resp. double 2's).

1. Let $u = (u_n)_{n \in \mathbb{N}} = 22 \cdot w_0 \cdot w_1 \cdots w_n \cdots$, where w_{n+1} is the image of w_n by the recoding rule along T_1 and T_2 (see (4) for an example).
2. Let $p_2 = 1 - p_1$ and $q_1 = 1 - q_2$. Let

$$A = \begin{pmatrix} p_1 & 0 & p_1 & 0 \\ p_2 & 0 & p_2 & 0 \\ 0 & 2q_1 & 0 & 2q_1 \\ 0 & 2q_2 & 0 & 2q_2 \end{pmatrix} \text{ and } v_n = \begin{pmatrix} |w_n|_a \\ |w_n|_b \\ |w_n|_c \\ |w_n|_d \end{pmatrix} \implies v_{n+1} = A \cdot v_n + e_n,$$

where e_n is an "error" vector and is bounded.

3. A is primitive with exactly two eigenvalues $0 < |\alpha_2| \leq 1 < \alpha_1$. By the Perron-Frobenius theorem, there exists a right (resp. left) eigenvector vectors \mathbf{r} (resp. ℓ) of A such that: $\ell \mathbf{r} = 1$ and $\lim_{n \rightarrow \infty} \alpha_1^{-n} A^n = \mathbf{r} \cdot \ell$.
4. One cuts the sequence u into words $(g_n)_{n \in \mathbb{N}}$ following Algorithm 1 and shows:
 - $|w_0 \cdots w_{\ell_n}| = o(|g_n|)$ and $\lim_{n \rightarrow \infty} |g_n| = +\infty$
 - $\frac{|u_0 \cdots u_n|_a}{|u_0 \cdots u_n|} = \frac{|w_0 \cdots w_{\ell_{n-1}}|_a + |g_n|_a}{|w_0 \cdots w_{\ell_{n-1}}| + |g_n|} = \frac{\frac{|g_n|_a}{|g_n|} + o(1)}{1 + o(1)} \xrightarrow{n \rightarrow \infty} r_0$

Algorithm 1: Cutting the sequence \mathcal{O} into $(g_n)_{n \in \mathbb{N}}$

Input: $u = w_0 w_1 \cdots w_n \cdots = u_0 u_1 \cdots$

```

1  $\ell_0 \leftarrow 0$  // initial left index
2 for each  $n \in \mathbb{N}$  do
3    $g_n \leftarrow u_{\ell_n} \cdots u_n$  //  $|g_n| = (n+1) - |w_0 \cdots w_{\ell_{n-1}}|$ 
4   if  $|g_n| + 1 > |w_0 \cdots w_{\ell_n}|^2$  then
5      $\ell_{n+1} \leftarrow \ell_n + 1$  // increment left index
6   else
7      $\ell_{n+1} \leftarrow \ell_n$  // keep left index
```

□

As a direct consequence of Theorem 1, we prove the existence of the frequencies in the sequence introduced in Section 5 of [3]:

Definition 2 (BJM sequence [3]). Let $U = (u_n)_{n \in \mathbb{N}}$ be the self-descriptive sequence $U = x_0^{u_0} x_1^{u_1} x_2^{u_2} \cdots$ defined by $x_0 = u_0 = u_1 = 2$, and for all $n \in \mathbb{N}^*$:

- i) if $u_n = 1$, then $x_n = 1$ (resp. $x_n = 2$) if $|u_0 \cdots u_n|_1$ is odd (resp. even),
- ii) if $u_n = 2$ then $x_n = 1$.

In other words, the runs of size 2 (except the first one) are filled by 1, and the runs of size 1 are filled alternatively by 1 and 2. The sequence $X = (x_n)_{n \in \mathbb{N}}$ is the directed sequence of U .

In [3], the authors showed that the frequencies of occurrence in U cannot be equal to those of its directing sequence X . However, the authors did not prove the existence of the frequencies but only that they cannot be identical... if they exist. Since U is directed by $T_1 = (12)^\omega$ and $T_2 = 1^\omega$, it directly follows from Theorem 1 that:

Corollary 1. *Let U be the sequence directed by $T_1 = (12)^\omega$ and $T_2 = 1^\omega$. Then*

$$\lim_{n \rightarrow \infty} \frac{|U_0 \cdots U_{n-1}|_1}{n} = \frac{7 - \sqrt{17}}{4} \text{ and } \lim_{n \rightarrow \infty} \frac{|X_0 \cdots X_{n-1}|_1}{n} = \frac{1 + \sqrt{17}}{8}.$$

Proof. In that present case, $p_1 = 0.5$ and $q_2 = 0$. □

4 Conclusion and perspectives

In the present work, we have shown that self-descriptive sequences directed by two periodic sequences have frequencies. We have also given an explicit expression for these frequencies.

In future work, it will be interesting to extend this result to non-periodic sequences. For example, Sturmian words, namely the aperiodic sequences with the least number of finite factors, are good candidates for directing sequences.

References

- [1] M. Baake & B. Sing (2004): *Kolakoski-(3, 1) Is a (Deformed) Model Set*. *Can. Math. Bull.* 47(2), p. 168–190, doi:10.4153/CMB-2004-018-6.
- [2] V. Berthé, S. Brlek & P. Choquette (2005): *Smooth words over arbitrary alphabets*. *Theor. Comput. Sci.* 341(1-3), pp. 293–310, doi:10.1016/j.tcs.2005.04.010.
- [3] C. Boisson, D. Jamet & I. Marcovici (2024): *On a probabilistic extension of the Oldenburger-Kolakoski sequence*. *RAIRO Theor. Informatics Appl.* 58, p. 11, doi:10.1051/ITA/2024005.
- [4] S. Brlek, S. Dulucq, A. Ladouceur & L. Vuillon (2006): *Combinatorial properties of smooth infinite words*. *Theor. Comput. Sci.* 352(1-3), pp. 306–317, doi:10.1016/j.tcs.2005.12.003.
- [5] S. Brlek, D. Jamet & G. Paquin (2008): *Smooth words on 2-letter alphabets having same parity*. *Theor. Comput. Sci.* 393(1-3), pp. 166–181, doi:10.1016/j.tcs.2007.11.019.
- [6] V. Chvátal (1993): *Notes on the Kolakoski sequence*. Technical Report, DIMACS Technical Report 93-84. Available at <http://users.encs.concordia.ca/~chvatal/93-84.pdf>.
- [7] M.S. Keane (1991): *Ergodic theory and subshifts of finite type*. In: *Ergodic theory, symbolic dynamics, and hyperbolic spaces. Lectures given at the workshop "Hyperbolic geometry and ergodic theory", held at the International Centre for Theoretical Physics in Trieste, Italy, 17-28 April, 1989*, Oxford etc.: Oxford University Press, pp. 35–70.
- [8] W. Kolakoski (1966): *Self-generating runs, Problem 5304*. *Amer. Math. Monthly* 73(6), pp. 681–682, doi:10.2307/2314839.
- [9] R. Oldenburger (1939): *Exponent Trajectories in Symbolic Dynamics*. *Trans. Amer. Math. Soc.* 46(3), pp. 453–466, doi:10.2307/1989933.
- [10] M. Rao: *Trucs et bidules sur la séquence de Kolakoski*. <https://www.arthy.org/kola/kola.php#Xchvatal>. Accessed: 2023-10-23.

About the Determinant of Complete non-ambiguous Trees

Jean-Christophe Aval

LaBRI, Université de Bordeaux

aval@labri.fr

Complete non-ambiguous trees (CNATs) are combinatorial objects which appear in various contexts. Recently, Chen and Ohlig studied the notion of leaf permutation on these objects, and proposed a series of nice conjectures. Most of them were proved by Selig and Zhu, through a connexion with the abelian sandpile model. But one conjecture remained open, about the distribution of a natural statistic named determinant. We prove this conjecture, in a bijective way.

1 Introduction

Non-ambiguous trees (NATs) were first defined in [1] and may be seen as a proper way to draw a binary tree on a square grid (see Definition 2.1). They were put to light as a special case of tree-like tableaux, which have been found to have applications in the PASEP model of statistical mechanics [5, 3]. The initial study of NATs revealed nice properties, mostly in an enumerative context [1, 2]. This includes enumeration formulas with respect to fixed constraints (hook formula), and new bijective proofs of combinatorial identities. When the underlying binary tree is *complete*, we are led to complete non-ambiguous trees (CNATs). These objects were first considered in [1], where it was proved that their enumerating sequence is related to the formal power series of the logarithm of the Bessel function of order 0. To end this early study of NATs, an extension to higher dimension was proposed in [9].

Recent works have revealed new facets of these objects. In [6], some striking mathematical cross-connections were obtained, such as a bijection between CNATs and fully-tiered trees of weight 0. In [7], CNATs were linked to the abelian sandpile model. In the same article, it was noticed that if we restrict a CNAT to its leaf dots, we obtain a permutation. This link was investigated in [4], where nice properties were derived, and several conjectures proposed. By using the connection with the abelian sandpile model, a large number of conjectures were proved very recently [10]. But a conjecture remained open. It asserts that when considering the set of CNATs of a fixed odd size, the number of them with an underlying permutation with even and odd determinant (signature) are equal. We give a bijective proof of this (Theorem 2.7), and include the case of the even size, which was suggested in [4].

In this extended abstract, some technical proofs are omitted.

2 Definitions and statement of the result

We first recall the definition of (complete) non-ambiguous trees, as in [1].

Definition 2.1. A *non-ambiguous tree* (NAT) T is a filling of an $m \times n$ rectangular grid, where each cell is either dotted or not, satisfying the following conditions:

(Existence of a root) The top-left cell is dotted; we call it the *root* of T .

(Non ambiguity) Aside from the root, every dotted cell of T has either a dotted cell above it in the same column, or a dotted cell to its left in the same row, but not both.

(Minimality) Every row and every column of T contains at least one dotted cell.

Remark 2.2. The use of the word *tree* to describe these objects comes from the following observation. Given a NAT T , we connect every dot d different from the root to its *parent* dot $p(d)$, which is the dot immediately above it in the same column, or to its left in the same row (because of the condition of non ambiguity, exactly one of these must exist).

A NAT is said to be complete if the underlying tree is complete, *i.e.* every internal dot has exactly two children.

Definition 2.3. A *complete non-ambiguous tree* (CNAT) is a NAT in which every dot has either both a dot below it in the same column and a dot to its right in the same row (in which case the dot is said to be an *internal dot*), or neither of these (in which case the dot is said to be a *leaf*).

The *size* of a CNAT is its number of leaf dots, or equivalently one more than its number of internal dots.

We denote by \mathcal{T}_n the set of CNATs of size n and $T_n = |\mathcal{T}_n|$.

Figure 1 gives an example of this notion.

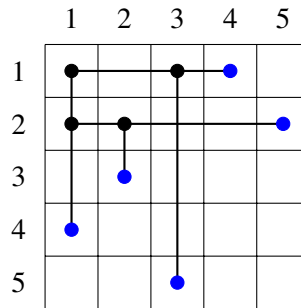


Figure 1: A CNAT of size 5. Leaf dots are represented in blue, and internal dots in black.

As in this figure, it will be convenient to label by integers the rows and columns respectively from top to bottom and from left to right (in such a way that the root appears in the cell $(1, 1)$). Moreover, given a dot d in a CNAT, we denote by $c(d)$ and $r(d)$ the (label of) its column and row. For a given internal dot, its child in the same row is called its *right child* and its child in the same column is called its *left child*.

Remark 2.4. We may observe that any right leaf l in a CNAT T is the only dot in its column: there is no dot above l because this would contradict the minimality condition of Definition 2.3, and there is no dot below l because l is a leaf. In the same way, any left leaf l in T is the only dot in its row.

We may see a CNAT T as a matrix $M(T)$ where dotted cells are 1's and undotted cells are 0's. For example, the CNAT of Figure 1 is encoded matricially as

$$\begin{pmatrix} 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix}.$$

The numbers T_n of CNATs of size n appear as the series A002190 in [11]. As proved in [1], these numbers give a combinatorial interpretation for the development of the Bessel function J_0 .

Let us now introduce the notion of permutation associated to a CNAT.

Definition 2.5. Let T be a CNAT of size n . It is clear that in any column of T the bottom-most dot of is a leaf, as well as the right-most dot of any row. Thus every row and every column must have exactly one leaf dot. As such, the set of leaf dots of a CNAT T of size n forms the graphical representation of an n -permutation $\pi(T)$. We say that $\pi(T)$ is the permutation associated with the CNAT T .

For example, the CNAT of Figure 1 has associated permutation $\pi(T) = 45312$.

A careful study of permutations associated to CNATs was initiated in [4], where the following proposition is proved.

Proposition 2.6. *Let T be a CNAT. We have:*

$$\det M(T) = \operatorname{sgn} \pi(T).$$

Let us denote by $T(n; \varepsilon)$ the number of CNATs of size n with determinant equal to ε . We are now in a position to state the main result of this article.

Theorem 2.7. *If $n > 1$ is odd:*

$$T(n; +1) = T(n; -1) = \frac{T_n}{2}. \quad (1)$$

If n is even (let us set $n = 2p$):

$$T(2p; +1) = \frac{T_{2p} + (-1)^p T_p}{2} \text{ and } T(2p; -1) = \frac{T_{2p} - (-1)^p T_p}{2}. \quad (2)$$

The odd case corresponds to Conjecture 2.6 in [4], the even case to Remark 2.7 in the same paper.

3 A bijective proof of Theorem 2.7

This section is devoted to proving our main result. This proof is bijective. More precisely, we shall:

1. introduce a subset $\mathcal{A}_{2p} \subset \mathcal{T}_{2p}$ of CNATs of even size, with $\mathcal{A}_{2p} = T_p$, and such that for any $T \in \mathcal{A}_{2p}$ we know that $\operatorname{sgn} \pi(T) = (-1)^p$;
2. construct an *involution* Φ on the set of CNATs such that if T is not in any of the sets \mathcal{A}_{2p} we have:

$$\operatorname{sgn} \pi(\Phi(T)) = -\operatorname{sgn} \pi(T).$$

We first introduce a useful notion on the leaves of CNATs.

Definition 3.1. A leaf in a CNAT is said to be *short* if its parent is in a cell adjacent to it. Otherwise the leaf is said to be *long*. Moreover, we denote by \mathcal{A}_n the set of CNATs of size n with only short leaves.

Figure 2 illustrates this notion.

The elements of \mathcal{A}_p are designed to be the fixed points of our involution Φ . We treat their case with two propositions, whose proofs are omitted.

Proposition 3.2. *When the size $n = 2p + 1$ is odd, the set \mathcal{A}_{2p+1} is empty. When the size $n = 2p$ is even, the set \mathcal{A}_{2p} is in bijection with \mathcal{T}_p .*

Proposition 3.3. *Let T be an element of \mathcal{A}_{2p} , its determinant is given by:*

$$\operatorname{sgn} \pi(T) = (-1)^p.$$

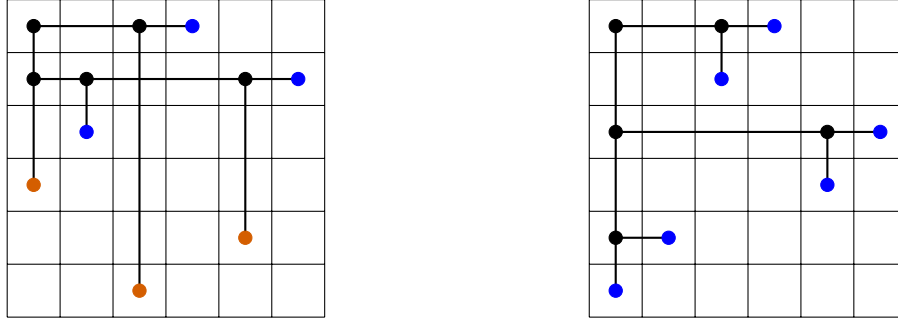


Figure 2: (Left) A CNAT with short leaves in blue and long leaves in red.
(Right) An element of \mathcal{A}_6 .

We now come to the definition of a function Φ on \mathcal{T}_n , which is the key construction of this work. We first introduce the following notion.

Definition 3.4. Let T be a CNAT, and l_1 and l_2 two leaves in T with respective parent p_1 and p_2 . If l_1 and l_2 are both left leaves, they are said to be *interacting* if

$$r(p_1) < r(l_2) < r(l_1) \text{ or } r(p_2) < r(l_1) < r(l_2).$$

The definition is similar for right leaves.

This notion is illustrated by Figure3.

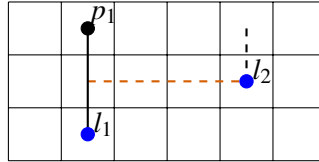


Figure 3: Interacting leaves.

The interest of this notion of interacting leaves is put to light by the following operation.

Definition 3.5. For two interacting left leaves l_1 and l_2 in a CNAT T , we define the *switch* of these two leaves as the exchange of the row labels of l_1 and l_2 . More precisely:

- we erase l_1 and we put a new leaf l'_1 in the same column, in row $r(l_2)$;
- we erase l_2 and we put a new leaf l'_2 in the same column, in row $r(l_1)$.

By doing this, we obtain an object $T' = S(T, l_1, l_2)$.

We have the same operation for right (interacting) leaves.

This notion is illustrated by Figure 4

Proposition 3.6. Let T be in \mathcal{T}_n . For two interacting leaves l_1 and l_2 in T , $S(T, l_1, l_2)$ is in \mathcal{T}_n .

Proof. The only condition in Definition 2.1 that is not trivially satisfied in non ambiguity. This is a direct consequence of Remark 2.4. \square



Figure 4: Switching interacting leaves.

The technical part of the construction of Φ now relies on the following lemma, whose proof is omitted.

Lemma 3.7. *Any CNAT with a long leaf has at least two interacting leaves.*

Let us now precise the construction of $\Phi : \mathcal{T}_n \rightarrow \mathcal{T}_n$. First of all, we set that for any $T \in \mathcal{A}_n$, $\Phi(T) = T$. This case done, we are reduced to the case where T has at least one long leaf. By Lemma 3.7, T contains interacting leaves. To define Φ , we want to *choose* a pair of interacting leaves. Since the set of interacting leaves may change when we switch leaves, we have to choose in a way such that we create an involution. If T contains left interacting leaves, we consider the (non-empty) set $\{(r(l_1), r(l_2)); l_1 \text{ and } l_2 \text{ interacting}\}$ and choose l_1 and l_2 which correspond to the lexicographical maximum of this set. Let us call these interacting leaves *active*. This done, we set $\Phi(T) = S(T, l_1, l_2)$. And if T contains only right interacting leaves, we consider the lexicographical maximum of the set $\{(c(l_1), c(l_2)); l_1 \text{ and } l_2 \text{ interacting}\}$ to choose the pair of active leaves.

Proposition 3.8. *The function Φ is an involution on \mathcal{T}_n . Moreover, if $\mathcal{T} \notin \mathcal{A}_n$ then*

$$\text{sgn } \pi(\Phi(T)) = -\text{sgn } \pi(T).$$

Proof. Omitted. □

Figure 5 shows an example of the application of Φ .

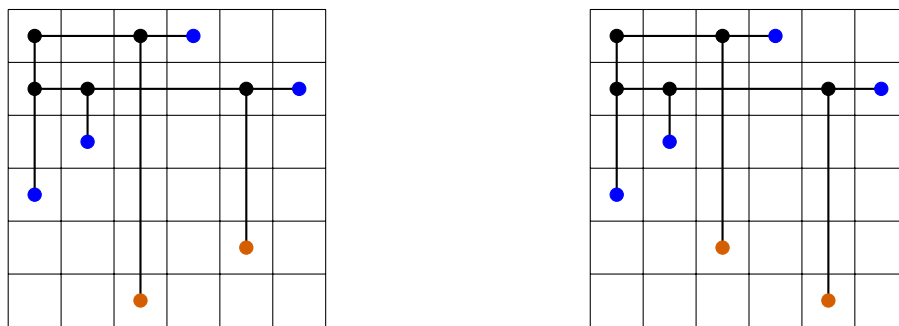


Figure 5: A CNAT and its image under Φ . Active leaves appear in red.

We can now conclude the proof of our main result.

Proof of Theorem 2.7. It is a consequence of Propositions 3.2, 3.3 and 3.8. □

References

- [1] Jean-Christophe Aval, Adrien Boussicault, Mathilde Bouvel & Matteo Silimbani (2014): *Combinatorics of non-ambiguous trees*. *Advances in Applied Mathematics* 56, pp. 78–108, doi:10.1016/j.aam.2013.11.004.
- [2] Jean-Christophe Aval, Adrien Boussicault, Bérénice Delcroix-Oger, Florent Hivert & Patxi Laborde-Zubieta (2021): *Non-ambiguous trees: new results and generalization*. *European Journal of Combinatorics* 95, p. 103331, doi:10.1016/j.ejc.2021.103331.
- [3] Jean-Christophe Aval, Adrien Boussicault & Philippe Nadeau (2011): *Tree-like Tableaux*. *FPSAC'11 - 23rd International Conference on Formal Power Series and Algebraic Combinatorics* 20, doi:10.37236/3440.
- [4] Daniel Chen & Sebastian Ohlig (2023): *Associated Permutations of Complete Non-Ambiguous Trees*, doi:10.48550/arXiv.2210.11117. arXiv:2210.11117.
- [5] Sylvie Corteel & Lauren Williams (2006): *A Markov Chain on Permutations which Projects to the PASEP*. *International Mathematics Research Notices* 17, doi:10.1093/imrn/rnm055.
- [6] William Dugan, Sam Glennon, Paul E. Gunnells & Einar Steingrímsson (2019): *Tiered trees, weights, and q -Eulerian numbers*. *Journal of Combinatorial Theory, Series A* 164, pp. 24–49, doi:10.1016/j.jcta.2018.12.002.
- [7] Mark Dukes, Thomas Selig, Jason P. Smith & Einar Steingrímsson (2019): *Permutation graphs and the abelian sandpile model, tiered trees and non-ambiguous binary trees*. *Electronic Journal of Combinatorics* 26(3), p. p3.29, doi:10.37236/8225.
- [8] Patxi Laborde Zubieta (2015): *Occupied corners in tree-like tableaux*. *Seminaire Lotharingien de Combinatoire* 74, p. 74b, doi:10.48550/arXiv.1505.06098.
- [9] Patxi Laborde-Zubieta (2017): *Combinatoire de l'ASEP, arbres non-ambigus et polyminos parallélogrammes périodiques*. Ph.D. thesis, Université de Bordeaux.
- [10] Thomas Selig & Haoyue Zhu (2023): *Complete non-ambiguous trees and associated permutations: connections through the Abelian sandpile model*. *arXiv e-prints:arXiv:2303.15756*, doi:10.48550/arXiv.2303.15756. arXiv:2303.15756.
- [11] Neil J. A. Sloane & The OEIS Foundation Inc. (2022): *The on-line encyclopedia of integer sequences*. <http://oeis.org/?language=english>.

Progressive and Rushed Dyck Paths

Axel Bacher

LIPN, Université Sorbonne Paris Nord, France

bacher@lipn.fr

We call *progressive paths* and *rushed paths* two families of Dyck paths studied by Asinowski and Jelínek, which have the same enumerating sequence (OEIS entry A287709). We present a bijection proving this fact. Rushed paths turn out to be in bijection with *one-sided trees*, introduced by Durhuus and Ûnel, which have an asymptotic enumeration involving a stretched exponential. We conclude by presenting several other classes of related lattice paths and directed animals that may have similar asymptotic properties.

1 Introduction

Dyck paths—sequences of up and down steps starting and ending at 0 and staying at nonnegative height—are probably the most famous family of lattice paths, one of the many combinatorial classes enumerated by the Catalan numbers (OEIS entry A000108). The aim of this article is to study two families of Dyck paths defined in the OEIS entry A287709 and studied by Asinowski in relation to certain rectangulation models [1]. We paraphrase their definitions below and illustrate them in Figure 1. Throughout the paper, we call *height* of the path P and denote by $h(P)$ the maximum height visited by P .

Definition 1. Let P be a Dyck path of height h . We say that P is *progressive* if, for $i = 2, \dots, h$, it visits the height $i - 1$ at least twice before the first visit at height i . We say that P is *rushed* if it starts with h up steps and then never again visits the height h .



Figure 1: Left: a progressive path of height 4 with the first and second visits at each height marked with a red and blue dot, respectively. Right: a rushed path of height 4. Note that progressive paths are allowed to visit their maximal height multiple times, while rushed paths are not.

A rushed path can be described as a run of up steps followed by a right-to-left *culminating path*, as defined by Bousquet-Mélou and Ponty [4] (a culminating path is a path that visits only nonnegative heights and visits its final height only once). This means that the enumeration of rushed paths of a given height is easily derived from that of culminating paths. The addition of a run of up steps can seem like a trivial difference; however, when considering paths irrespective of height, it makes a significant difference in the asymptotic enumeration.

A remarkable fact, due to Asinowski and Jelínek and stated in the OEIS entry mentioned above, is that progressive and rushed paths, when counted according to just length, have the same enumerating sequence (this is not true when taking height into account). This calls for a bijection, but such a bijection

has apparently remained elusive. We present such a bijection and explain how it behaves with respect to the height statistic.

Rushed paths, as it turns out, have already been studied by Durhuus and Ünel in the guise of *one-sided trees*, defined as trees with a height equal to the length of their leftmost branch [9]; such trees map to rushed paths via standard bijections (Figure 2). Their enumerating sequence exhibit a remarkable asymptotic estimate of the form $4^n e^{-\nu n^{1/3}} n^{-5/6}$. This sort of *stretched exponential* has attracted recent attention in many combinatorial contexts [11, 6, 10, 8], but it highly unusual for lattice paths, which usually have a form $\mu^n n^\gamma$, with γ normally 0, $-1/2$ or $-3/2$ in one dimension [2]. Culminating paths, for their part, asymptotically number $2^n/(4n)$ [4, Proposition 4.1]. Pushed Dyck paths also have the stretched exponential, but involve weights on paths rather than being a proper subset (see [11], or [8] for their tree avatars—remarkably, the asymptotic form is the same as rushed paths when the parameter is $1/2$).



Figure 2: Left: a rushed path of height 4 and semilength 11. Right: the corresponding one-sided tree of height 3 with 10 edges (the leftmost edge generated by the classical bijection is omitted).

The paper is organized as follows. In Section 2, we describe our bijection between rushed and progressive paths. In Sections 3 and 4, we give exact and asymptotic enumeration results, respectively. Finally, in Section 5, we give some perspectives, including classes of directed animals linked to progressive paths.

2 Bijections

In this section, we present our bijection from rushed to progressive paths. First, we need another bijection, which goes from Dyck paths to *progressive culminating paths* (culminating paths satisfying the same constraint as progressive paths). Let P be a Dyck path of height h . Decompose it as:

$$P = A_0 u \cdots A_{h-1} u \cdot A_h \cdot dB_{h-1} \cdots dB_0 \quad (1)$$

where the A_i 's and B_i 's are downward Dyck paths of height at most i , possibly empty (the factors A_0 and B_0 are always empty in the Dyck setting, but including them makes it easier to describe the bijections without special cases). Let $F(P)$ be the path:

$$F(P) = A_0 u \cdot dB_0 u A_1 u \cdots dB_{h-1} u A_h u. \quad (2)$$

The decompositions of P and $F(P)$ are illustrated in Figure 3.

Theorem 2. *The mapping F is a bijection between Dyck paths of length $2n$ and height h and progressive culminating paths of length $2n + h + 1$ and height $h + 1$.*

Proof. First, we check that $F(P)$ is a culminating progressive path. By construction, we see that the factor $dB_{i-2} u A_{i-1} u$ goes from the first visit at height $i - 1$ to the first visit at height i ; it never visits negative height and visits the height $i - 1$ at least twice, so $F(P)$ is culminating and progressive.



Figure 3: Left: a Dyck path of height 3 decomposed as in (1). Right: its image by F , a progressive culminating path of height 4 decomposed as in (2).

Second, the decomposition (2) can be recovered from $F(P)$ by cutting at the first and second visits at height i , for $i = 1, \dots, h - 1$, which allows us to rebuild the path P according to (1). This shows that F is a bijection. \square

We are now ready to define our main bijection. Let P be a rushed path of height h . Decompose it as:

$$P = u^h \cdot dA_0 \cdots dA_{h-1} \tag{3}$$

where the A_i 's are Dyck paths of height at most i . Let m be the maximum height of A_i for $i = 0, \dots, h - 1$ and let j be the smallest index with $h(A_j) = m$ (necessarily, we have $j \geq m$). Let $G(P)$ be the path:

$$G(P) = F(A_j) \cdot A_0 d \cdots A_{m-1} d \cdot uA_m d \cdots uA_{j-1} d \cdot d \cdot uA_{j+1} d \cdots uA_{h-1} d. \tag{4}$$

The decompositions (3) and (4) are illustrated in Figure 4.

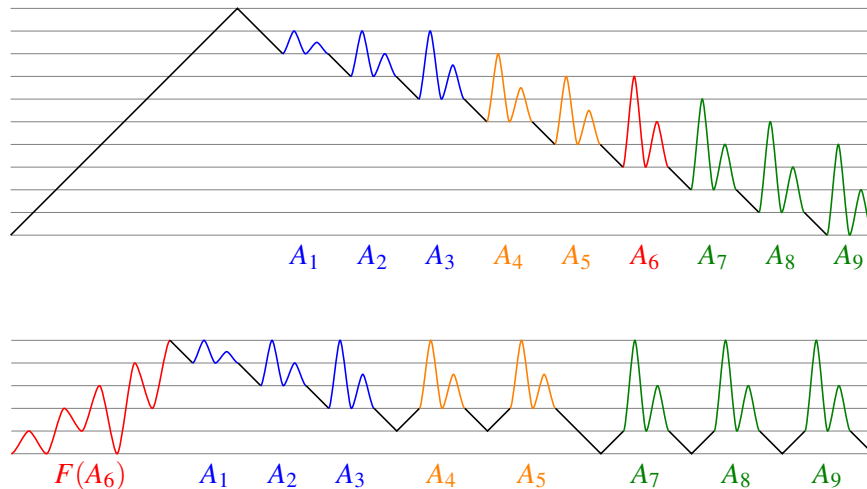


Figure 4: Above: a rushed Dyck path of height 10 decomposed as in (3), where $m = 4$ and $j = 6$ (so A_6 has height exactly 4, while A_7, A_8 and A_9 have height at most 4). Below: its image by G , a progressive Dyck path of height 5 decomposed as in (4).

Theorem 3. *The mapping G is a bijection from rushed paths of length $2n$ to progressive paths of length $2n$.*

Proof. First, we check that $G(P)$ is a progressive path of height $m + 1$ and length $2n$. The factor $F(A_k)$ is a progressive culminating path of height $m + 1$ by Theorem 2; since the factors A_i have height at most i

for $i = 0, \dots, m-1$, at most $m-1$ for $i = m, \dots, k-1$ and at most m for $i = k, \dots, h-1$, the rest of the path never visits heights above $m+1$. Finally, the length of $G(P)$ is $|P| + |F(A_j)| - |A_j| - m - 1$, which is $2n$ by Theorem 2.

Second, the decomposition (4) and thus the path P can be recovered from the path $G(P)$ by cutting at the first visit at height $m+1$, at the first visit at height i thereafter for $i = m, \dots, 2$, at every visit at height 1 thereafter before the first visit at height 0 and at every visit at height 0 thereafter. This shows that G is a bijection. \square

3 Enumeration and generating functions

Here we state exact enumeration results for our paths, based on classical results on Dyck paths of bounded height [7] and culminating paths [4]. In the following, $F_h(z)$ are the Fibonacci polynomials defined by $F_0(z) = F_1(z) = 1$ and $F_h(z) = F_{h-1}(z) - zF_{h-2}(z)$ for $h \geq 2$, while $q := q(z)$ is the series of Catalan numbers satisfying $q = z(1+q)^2$. The proof will appear in a longer version.

Theorem 4. *The generating functions of rushed and progressive paths with height h are:*

$$R_h(z) = \frac{z^h}{F_h(z)} \quad \text{and} \quad P_h(z) = \frac{z^{2h-1}}{F_{h-1}(z)F_h(z)F_{h+1}(z)}. \quad (5)$$

The generating function of rushed (or progressive) paths is:

$$R(z) = P(z) = \sum_{h \geq 0} \left(\frac{q}{1+q} \right)^h \frac{1-q}{1-q^{h+1}}. \quad (6)$$

Finally, the number of rushed paths of semilength n and height $h-1$ is given by:

$$r_{n,h-1} = \frac{4^{n+1}}{2^h h} \sum_{j=1}^{\lfloor \frac{h-1}{2} \rfloor} (-1)^{j+1} \sin^2 \frac{j\pi}{h} \cos^{2n-h} \frac{j\pi}{h}. \quad (7)$$

4 Asymptotics

Below is our result for asymptotic enumeration, stated without proof in this extended abstract.

Theorem 5. *The number of rushed (or progressive) Dyck paths of semilength n satisfies, as n tends to infinity:*

$$r_n = p_n = \lambda 4^n e^{-\nu n^{1/3}} n^{-5/6} [1 + O(n^{-1/3})] \quad (8)$$

where $\lambda = (4\pi)^{5/6} (\log 2)^{1/3} / \sqrt{3}$ and $\nu = 3(\pi \log 2 / 2)^{2/3}$.

Moreover, let \mathfrak{R}_n be a uniformly distributed rushed path of semilength n . The height of \mathfrak{R}_n , properly rescaled, tends to a normal law:

$$\frac{h(\mathfrak{R}_n) - \mu n^{1/3}}{\sigma n^{1/6}} \xrightarrow{d} \mathcal{N}(0, 1) \quad (9)$$

where $\mu = (2\pi^2 / \log 2)^{1/3}$ and $\sigma = (2\pi^2)^{1/6} / ((\log 2)^{2/3} \sqrt{3})$. We also have:

$$h(\mathfrak{R}_n) - h(G(\mathfrak{R}_n)) \xrightarrow{d} \mathcal{D} \quad \text{where} \quad \mathbf{E}[u^{\mathcal{D}}] = \frac{1}{u} \left(1 + \frac{u-1}{2-u} 2^{\frac{u}{2-u} + 1} \right). \quad (10)$$

In particular, the height of a random progressive path of length $2n$ has the same Gaussian limit law as $h(\mathfrak{R}_n)$.

The estimate (8) and the limit law (9) are directly in [9]. The limit law (10) is also derived using a local limit law found in the same reference.

5 Perspectives

Many interesting problems remain on this topic. First, classes of Dyck paths similar to progressive paths may exhibit similar asymptotical behavior: OEIS entry A287776 describes paths one could name *doubly progressive*—both left-to-right and right-to-left progressive. The definition of progressive paths also naturally extends to unconstrained paths (not necessarily staying positive or ending at 0): an unconstrained path is progressive if, before its first visit at every height $i \notin \{0, 1\}$, it visits twice either $i - 1$ or $i + 1$. There are many possible variations and it would be interesting to see which can be enumerated and if results like (8) and (9) hold.

Dyck paths are also linked to directed animals on the triangular lattice, via bijections with heaps of dimers [12, 5]. Adopting the convention that animals grow towards the right side, we say that a directed animal is *acute* if, for every site at ordinate $y \neq 0$, there are at least two sites to the left with ordinate $y - 1$ or $y + 1$. Progressive paths are in bijection with acute half-animals, or acute animals with all sites above the x -axis (in this case, for every site at ordinate $y > 0$, there are two sites to the left at ordinate $y - 1$; see Figure 5). Full acute animals seem harder to enumerate.

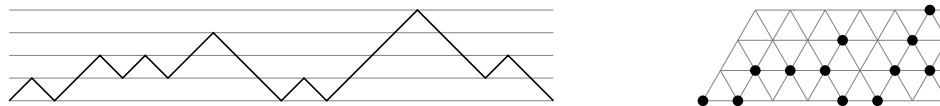


Figure 5: Left: a progressive path. Right: the corresponding acute half-animal: for every site not on the bottom row, there are at least two sites to the left in the row just below.

One may want to go beyond Dyck paths, to Motzkin paths, Schröder paths, m -Dyck paths, or more general models [2, 3], as well as directed animals on different lattices. It is not clear, however, which of the many possibilities is the “right” extension of the definition of progressive paths. The adaptation of the asymptotic results may also prove challenging.

A final problem is the efficient random generation of our paths. Bousquet-Mélou and Ponty gave several beautiful algorithms for the generation of uniform culminating paths [4], but they are difficult to adapt to our case, because the distribution of the height is different. Using the recursive method [13] and several laziness ideas to reduce computation, it is possible to achieve a subquadratic algorithm. Details will appear in a longer version.

References

- [1] Andrei Asinowski (2024): Private communication.
- [2] Cyril Banderier & Philippe Flajolet (2002): *Basic analytic combinatorics of directed lattice paths*. *Theoretical Computer Science* 281(1-2), pp. 37–80, doi:10.1016/S0304-3975(02)00007-5. Selected papers in honour of Maurice Nivat.
- [3] Mireille Bousquet-Mélou (2006/08): *Discrete excursions*. *Séminaire Lotharingien de Combinatoire* 57, pp. Art. B57d, 23.

- [4] Mireille Bousquet-Mélou & Yann Ponty (2008): *Culminating paths*. *Discrete Mathematics and Theoretical Computer Science* 10(2), pp. 125–152, doi:10.46298/dmtcs.438.
- [5] Mireille Bousquet-Mélou & Andrew Rechnitzer (2002): *Lattice animals and heaps of dimers*. *Discrete Mathematics* 258, pp. 235–274, doi:10.1016/S0012-365X(02)00352-7.
- [6] Yu-Sheng Chang, Michael Fuchs, Hexuan Liu, Michael Wallner & Guan-Ru Yu (2022): *Enumeration of d -Combining Tree-Child Networks*. In: *33rd International Conference on Probabilistic, Combinatorial and Asymptotic Methods for the Analysis of Algorithms (AofA 2022)*, *Leibniz International Proceedings in Informatics (LIPIcs)* 225, pp. 5:1–5:13, doi:10.4230/LIPIcs.AofA.2022.5. Available at <https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.AofA.2022.5>.
- [7] N.G. de Bruijn, D.E. Knuth & S.O. Rice (1972): *The Average Height of Planted Plane Trees*. In: *Graph Theory and Computing*, Academic Press, pp. 15–22, doi:10.1016/B978-1-4832-3187-7.50007-6.
- [8] Bergfinnur Durhuus & Meltem Unel (2023): *Trees with exponential height dependent weight*. *Probability Theory and Related Fields* 186, pp. 1–45, doi:10.1007/s00440-023-01188-7.
- [9] Bergfinnur Durhuus & Meltem Unel (2024): *Local Limits of One-Sided Trees*. *La Matematica* 3, pp. 131–165, doi:10.1007/s44007-023-00080-z.
- [10] Andrew Elvey Price, Wenjie Fang & Michael Wallner (2021): *Compacted binary trees admit a stretched exponential*. *Journal of Combinatorial Theory, Series A* 177, p. 105306, doi:10.1016/j.jcta.2020.105306.
- [11] Anthony J. Guttmann (2015): *Analysis of series expansions for non-algebraic singularities*. *Journal of Physics A: Mathematical and Theoretical* 48(4), p. 045209, doi:10.1088/1751-8113/48/4/045209. Available at <https://dx.doi.org/10.1088/1751-8113/48/4/045209>.
- [12] Gérard Xavier Viennot (1986): *Heaps of pieces. I. Basic definitions and combinatorial lemmas*. In: *Combinatoire énumérative (Montreal, Que., 1985/Quebec, Que., 1985)*, *Lecture Notes in Mathematics* 1234, Springer, Berlin, pp. 321–350, doi:10.1007/BFb0072524.
- [13] Herbert Wilf (1977): *A unified setting for sequencing, ranking, and selection algorithms for combinatorial objects*. *Advances in Mathematics* 24, pp. 281–291, doi:10.1016/0001-8708(77)90059-7.

Interval Posets and Polygon Dissections

Eli Bagno, Estrella Eisenberg, Shulamit Reches and Moriah Sigron

Jerusalem College of Technology, 21 HaVaad HaLeumi St., Jerusalem, Israel

bagnoe@g.jct.ac.il

The Interval poset of a permutation is an effective way of capturing all the intervals of the permutation and the inclusions between them and was introduced recently by Tenner. This paper explores the geometric interpretation of interval posets of permutations. We present a bijection between tree interval posets and convex polygons with non-crossing diagonals, offering a novel geometric perspective on this purely combinatorial concept. Additionally, we provide an enumeration of interval posets using this bijection and demonstrate its application to block-wise simple permutations.

1 Introduction

In [4], Tenner defined the concept of an interval poset of a permutation. This is an effective way of capturing all the intervals of a permutation and the set of inclusions between them in one glance. Tenner dealt with structural aspects of the interval poset and characterized the posets P that can be seen as interval posets of some permutations.

An interval poset might correspond to more than one permutation. For instance, all simple permutations of a given order n share the same interval poset. Tenner, in the aforementioned paper, enumerated binary interval posets and binary tree interval posets but left open the following question:

Question 1.1. *How many tree interval posets have n minimal elements?*

This question was answered by Bouvel, Cioni and Izart in [2]. They also noted that the number of tree interval posets is equal to the number of ways to place non-crossing diagonals in a convex $(n+1)$ -gon such that no quadrilaterals are created.

In this work we suggest a simple bijection between the set of tree interval posets and the set of $(n+2)$ -gons, satisfying the conditions listed above. We use this bijection also for enumerating the whole set of interval posets by using a broader set of polygons. In [2], the enumeration of the entire set of interval posets was done in an algebraic way, using generating functions, while our bijection grants a geometric view to the interval posets.

Another set of interval posets that can be enumerated by polygons is the one corresponding to block-wise simple permutations, a term that was introduced in a recent paper by the current authors [1].

2 Background

Definition 2.1. Let S_n the symmetric group on n elements. Let $\pi = a_1 \cdots a_n \in S_n$. An *interval* (or *block*) of π is a non-empty contiguous sequence of entries $a_i a_{i+1} \cdots a_{i+k}$ whose values also form a contiguous sequence of integers. For $a < b$, $[a, b]$ denotes the interval of values that range from a to b . Clearly, $[n] := [1, n]$ is an interval, as well as $\{i\}$ for each $i \in [n]$. These are called trivial intervals. The other intervals are called *proper*.

For example, the permutation $\pi = 314297856$ has $[5, 9] = 97856$ as a proper interval as well as the following proper intervals: $[1, 4], [5, 6], [7, 8], [7, 9], [5, 8]$.

A permutation $\pi \in \mathcal{S}_n$ is called *simple* if it does not have proper intervals. For example, the permutation 3517246 is simple.

Following Tenner [4], we define an *interval poset* for each permutation as follows:

Definition 2.2. The interval poset of a permutation $\pi \in \mathcal{S}_n$ is the poset $P(\pi)$ whose elements are the non-empty intervals of π ; the order is defined by set inclusion (see for example Figures 1 and 2). The minimal elements are the intervals of size 1.

In [4], the interval poset is embedded in the plane so that each node's direct descendants are increasingly ordered according to the minimum of each interval from left to right. We note that in [2] another embedding of the same poset was presented.

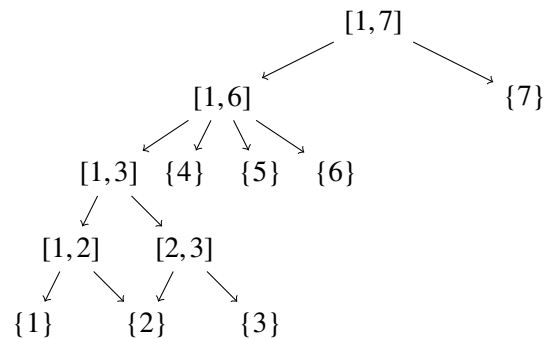


Figure 1: Interval poset of the permutations: 5123647, 5321647, 4612357, 4632157, 7463215, 7461235, 7532164, 7512364

If π is a simple permutation, the interval poset of π comprises the entire interval $[1, \dots, n]$ with minimal elements $\{1\}, \dots, \{n\}$ as its only descendants. Hence, all simple permutations of a given order n share the same interval poset (see for example Figure 2).

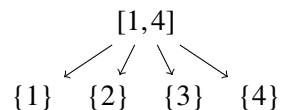


Figure 2: Interval poset of permutations 3142 and 2413.

3 Geometrical view of interval posets

3.1 General interval posets

Bouvel, Cioni and Izart[2], provided a formula for the number of interval posets with n minimal elements and added it to OEIS as sequence A348479 [3].

Here we provide a geometrical view to the interval posets by providing a bijection from the set of interval posets with n minimal elements to a distinguished set of dissections of the convex $(n+1)$ -gon, which we define below.

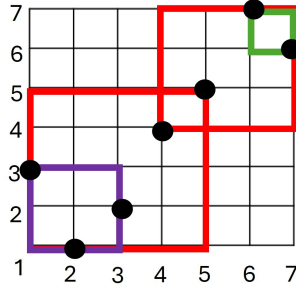


Figure 3: The permutation $\pi = 3124576$ and its blocks in a graphical way.

We identify a polygon with its set of vertices and denote a diagonal or an outer edge of the polygon from vertex i to vertex j by $\{i, j\}$.

Definition 3.1. A dissection of an $(n + 1)$ -gon will be called *diagonally framed* if for each two crossing diagonals, their vertices are connected to each other. Explicitly, if $\{a, b\}$ and $\{c, d\}$ are two crossing diagonals, then the diagonals or outer edges $\{a, d\}, \{b, d\}, \{c, b\}, \{c, a\}$ must also exist. See Figure 5 for an example.

Before we proceed, we have to present two observations which provide some details on the structure of interval posets and will be used in the sequel.

Observation 3.2. Let $\pi \in S_n$. If I and J are intervals of π such that $I \not\subseteq J$ and $J \not\subseteq I$ and $I \cap J \neq \emptyset$, then $I \cap J, I \cup J, I - J$ and $J - I$ are intervals of π .

For example, take $\pi = 3124576$, then $I = [1, 5]$ and $J = [4, 7]$ are intersecting intervals of π and thus $I \cup J = [1, 7], I \cap J = [4, 5], I - J = [1, 3], J - I = [6, 7]$ are also intervals of π , as can be seen in Figure 3 which depicts the permutation π in the common graphical way.

Observation 3.3. If $P(\pi)$ is the interval poset of $\pi \in S_n$, then no element of $P(\pi)$ has exactly 3 direct descendants, since every permutation of order 3 must contain a block of order 2.

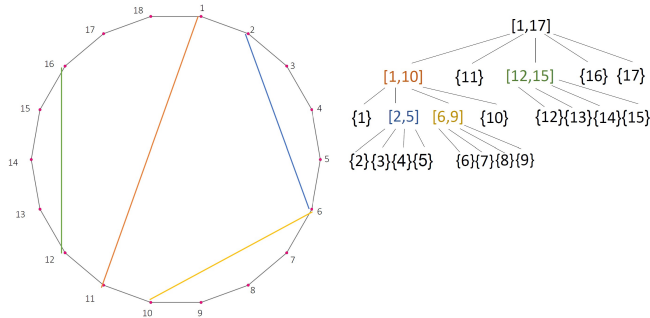


Figure 4: Right: the interval poset P . Left: the polygon $\Phi(P)$

We are ready now to present the main result of this subsection.

Theorem 3.4. The number of interval posets with n minimal elements is equal to the number of diagonally framed dissections of the convex $(n + 1)$ -gon such that no quadrilaterals are present (see Figure 6 in the appendix for some examples of the bijection in small values of n).

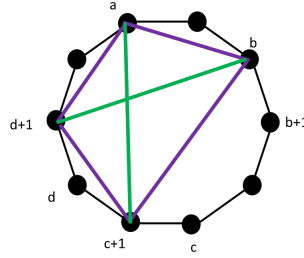


Figure 5:

Proof. We define a bijection between the set of interval posets with n minimal elements and the set of diagonally framed dissections of convex $(n+1)$ -gons without quadrilaterals as follows:

Let P be the interval poset of some $\pi \in S_n$. We set $\Phi(P)$ to be the convex $(n+1)$ -gon whose set of diagonals is

$$\{\{a, b+1\} \mid [a, b] \text{ is an internal node of } P\},$$

i.e. to each interval of the form $[a, b]$ corresponds a diagonal $\{a, b+1\}$ in $\Phi(P)$; note that singletons intervals correspond to outer edges in the polygon (see Figure 4 for an example).

We claim now that $\Phi(P)$ must be a diagonally framed $(n+1)$ -gon. Indeed, if $\{a, c+1\}$ and $\{b, d+1\}$ are two crossing diagonals in $\Phi(P)$, where $a \leq b \leq c \leq d$, then $I = [a, c]$ and $J = [b, d]$ are intersecting intervals in P and by Observation 3.2 we have that $I \cup J = [a, d]$, $I \cap J = [b, c]$, $I - J = [a, b-1]$ and $J - I = [c+1, d]$ are intervals in P corresponding respectively to the diagonals $\{a, d+1\}$, $\{b, c+1\}$, $\{a, b\}$ and $\{c+1, d+1\}$. (See Figure 5 for an illustration).

Moreover, $\Phi(P)$ must not contain any quadrilateral. Otherwise, if $a < b < c < d$ are such that $\{a, b, c, d\}$ is a quadrilateral (without any subdivision) then P must contain the intervals $[a, b-1]$, $[b, c-1]$, $[c, d-1]$ and $[a, d-1]$ so we must have that the first three intervals are direct descendants of the fourth one and they are the only ones. By Observation 3.3, this is impossible. \square

3.2 Tree interval posets

A *tree poset* is a poset whose Hasse diagram is a tree.

In [2], the authors calculated the generating function of the number of tree interval posets using generating functions and mentioned that this is equal to the number of ways to place non-crossing diagonals in a convex $(n+2)$ -gon such that no quadrilaterals are created (sequence A054515 from OEIS [3]).

Using the function Φ defined above, one can easily produce a combinatorial proof of the following result.

Theorem 3.5. *The number of tree interval posets with n minimal elements is equal to the number of non crossing dissections of the convex $(n+1)$ -gon such that no quadrilaterals are present (see Figure 7 in the appendix for some examples of the bijection).*

Proof. We use the same mapping Φ which was applied in the proof of Theorem 3.4. It is now sufficient to prove that no crossing diagonals are obtained. This is implied by the fact that intersecting diagonals stem from intersecting intervals which can not exist in a tree since they cause a circle. (See Figure 1). \square

3.3 Interval posets of block-wise simple permutations

In [1], the current authors introduced the notion of block-wise simple permutations. We cite here the definition:

Definition 3.6. A permutation $\pi \in S_n$ is called *block-wise simple* if it has no interval of the form $p_1 \oplus p_2$ or $p_1 \ominus p_2$, where \oplus and \ominus stand for direct and skew sums of permutations respectively.

There are no block-wise simple permutations of orders 2 and 3. For $n \in \{4, 5, 6\}$, a permutation is block-wise simple, if and only if it is simple. One of the first nontrivial examples of block-wise simple permutations is 4253716.

In [1], the current authors enumerated the interval posets of block-wise permutations.

The first few values of the sequence of these numbers are 1, 1, 1, 5, 10, 16, 45, 109, 222, 540. This is sequence A054514 from OEIS [3] which also counts the number of ways to place non-crossing diagonals in a convex $(n+4)$ -gon such that there are no triangles or quadrilaterals.

The geometrical interpretation of interval posets of block-wise permutations is as follows:

Theorem 3.7. *The number of interval posets that represent a block-wise simple permutation of order n is equal to the number of ways to place non-crossing diagonals in a convex $(n+1)$ -gon such that no triangles or quadrilaterals are present (see Figure 8 in the appendix for some examples of the bijection).*

Proof. We use again the mapping Φ , defined earlier. In [4] (Theorem 6.1), the author claimed that $P(\sigma)$ is a tree interval poset if and only if σ contains no interval of the form $p_1 \oplus p_2 \oplus p_3$ or $p_1 \ominus p_2 \ominus p_3$. From here, and by Definition 3.6, it is obvious that an interval poset of a block-wise simple permutation is a tree. Hence it is sufficient to prove that for an interval poset P of a block-wise permutation, $\Phi(P)$ has no triangles. This holds due to the fact that if $\Phi(P)$ contains a triangle with edges $\{a, b\}, \{b, c\}, \{a, c\}$ with $a < b < c$ then P must contain the intervals $[a, b-1], [b, c-1]$ and $[a, c-1]$ and thus $[a, c-1]$ is the direct parent of $[a, b-1]$ and $[b, c-1]$ which contradicts the definition of block-wise simple permutations. \square

References

- [1] E. Bagno, E. Eisenberg, S. Reches & M. Sigron (2023): *Blockwise simple permutations*. arXiv:2303.13115. Note.
- [2] M. Bouvel, L. Cionl & B. Izart (2021): *The interval posets of permutations seen from the decomposition tree perspective*. arXiv:2110.10000. Note.
- [3] OEIS Foundation Inc. (2022): *The On-Line Encyclopedia of Integer Sequences*. Available at <http://oeis.org>.
- [4] B.E. Tenner (2022): *Interval Posets of Permutations. Orders*, doi:10.1007/s11083-021-09576-1.

4 Appendix

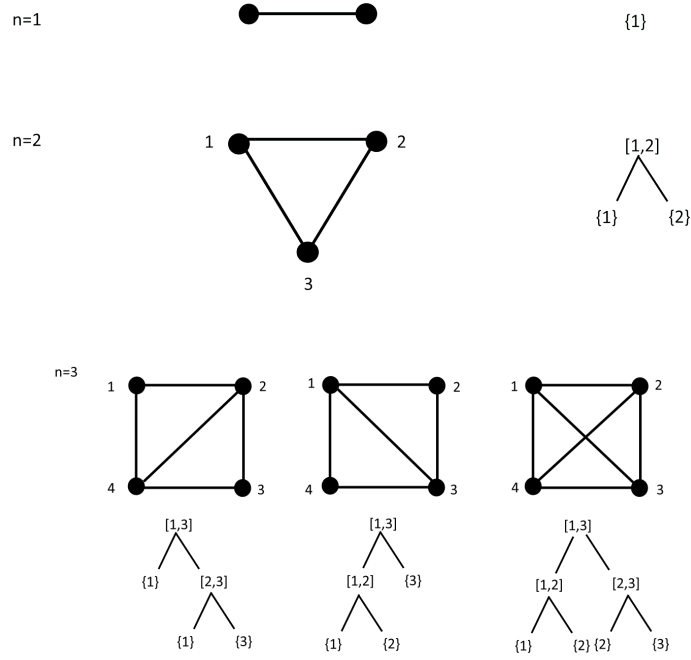


Figure 6: The bijection for small values of n

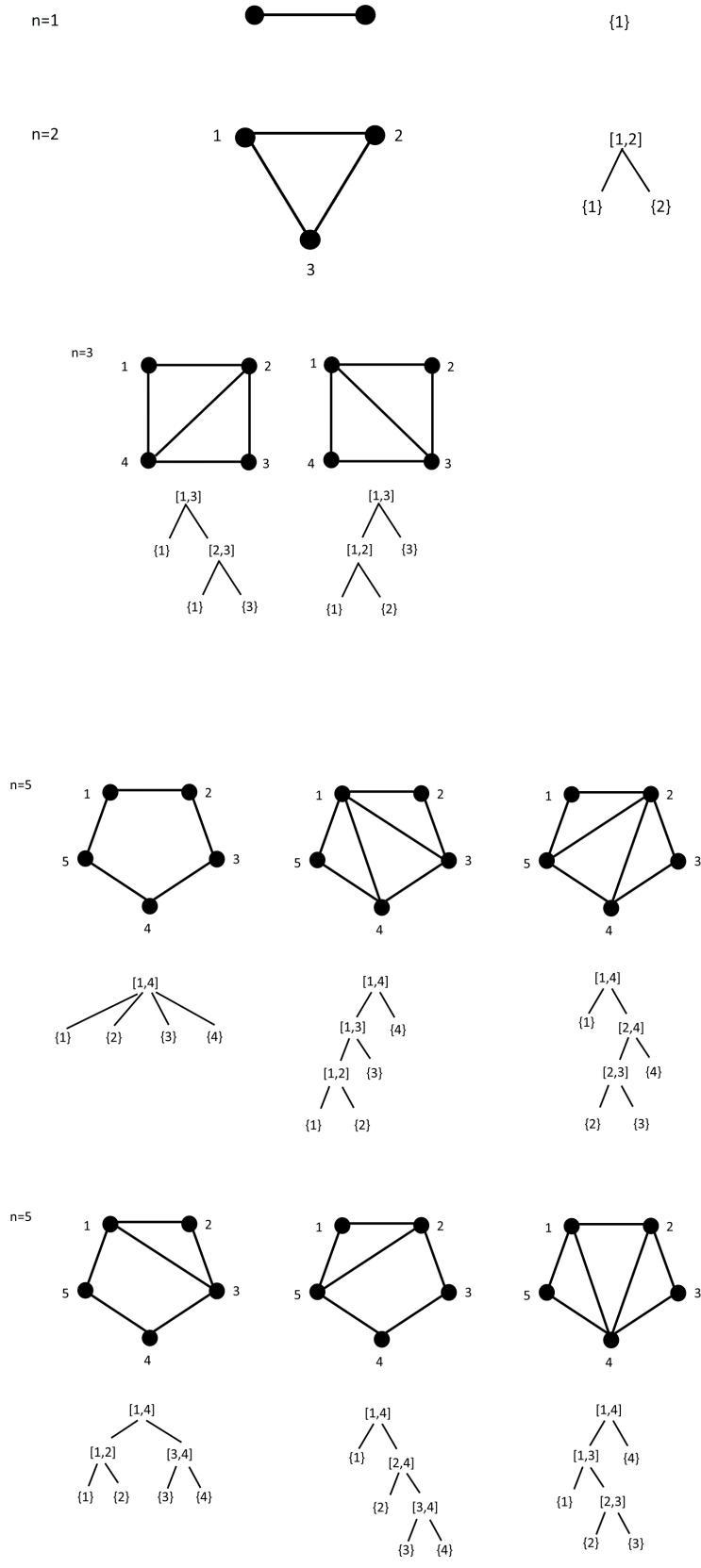


Figure 7: Examples for the bijection of tree intervals for small values of n

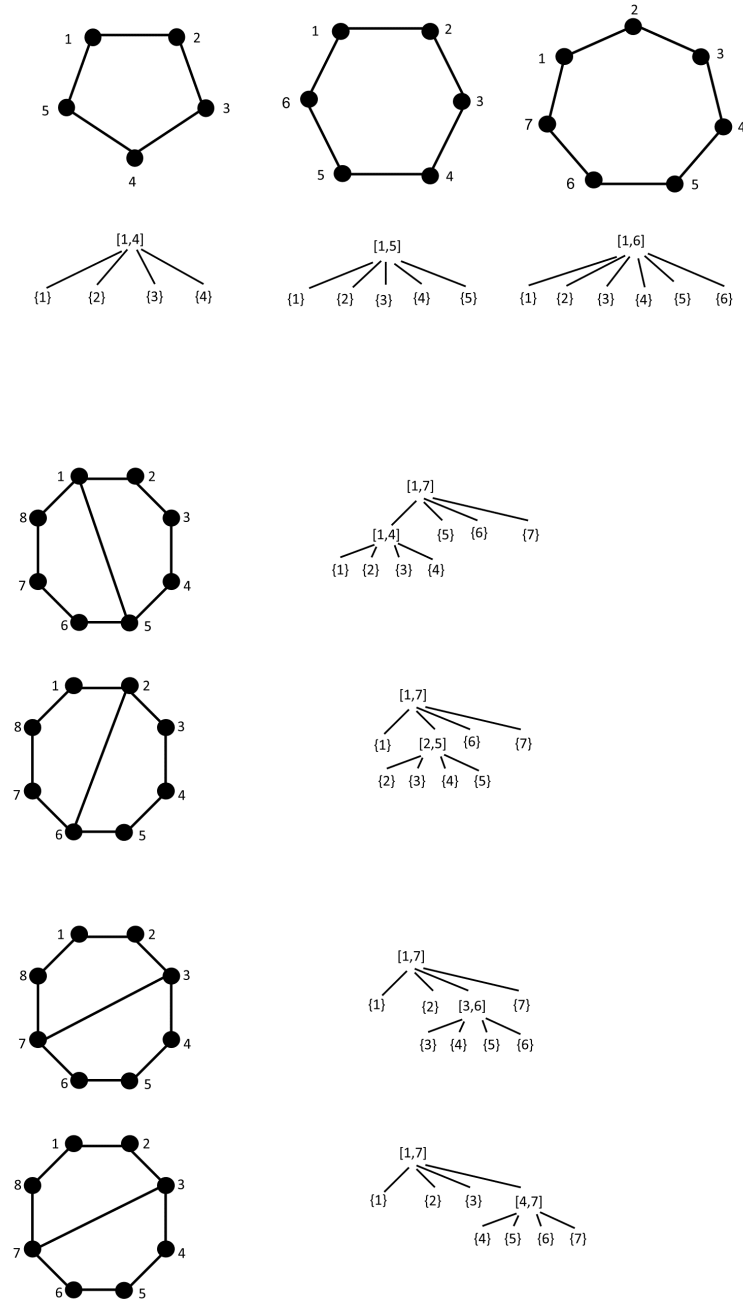


Figure 8: Examples for the bijection of block-wise simple intervals for small values of n

Type- B analogue of Bell numbers using Rota's Umbral calculus approach

Eli Bagno

David Garber

Jerusalem College of Technology,
21 HaVaad HaLeumi St.,
Jerusalem, Israel,
and Michlalah College Jerusalem,
36 Barukh Duvdevani
St., Jerusalem, Israel
bagnoe@g.jct.ac.il

Holon Institute of Technology,
52 Golomb St., P.O.Box 305,
5810201 Holon, Israel
garber@hit.ac.il

Rota used the functional L to recover old properties and obtain some new formulas for the Bell numbers. Tanny used Rota's functional L and the celebrated Worpitzky identity to obtain some expression for the ordered Bell numbers, which can be seen as an evident to the fact that the ordered Bell numbers are gamma-positive. In this paper, we extend some of Rota's and Tanny's results to the framework of the set partitions of Coxeter type B .

1 Introduction

Rota [12] declares:

“It is the author's conviction that formula (4), which we derive below, is the natural description of the exponential numbers. The basic idea is a general one, and can be applied to a variety of other combinatorial investigations. We shall see that it easily leads to quick derivations of the properties of the B_n .”

The ‘exponential numbers’ mentioned in the cited paragraph are the obsolete name for what we call today *Bell numbers*, which count set partitions of the set $[n] = \{1, \dots, n\}$. Rota's Formula (4) reads: $B_n = L(u^n)$, where L is a linear functional defined on the vector space of polynomials in the indeterminate u and B_n is the n -th Bell number.

Rota [12] used the functional L to recover old properties and obtain some new formulas for the Bell numbers. Explicitly, let $V = \mathbb{R}[u]$ be the vector space of all real polynomials in the single variable u . Then any sequence of polynomials of degree $0, 1, 2, \dots$ is a basis for this vector space, in particular the sequence of the *falling factorials* $(u)_0 = 1, (u)_1 = u, (u)_2 = u(u-1), \dots$ is a basis as well. Let $L : \mathbb{R}[u] \rightarrow \mathbb{R}$ be the linear functional that is uniquely defined by $L((u)_j) = 1$, for all $j \geq 0$. Rota [12] states the following theorem:

Theorem 1.1. *Let $n \geq 0$. Then:*

- (1) $B_n = L(u^n)$,
- (2) $B_{n+1} = \sum_{j=0}^n \binom{n}{j} B_j$,
- (3) $B_n = \frac{1}{e} \sum_{j \geq 0} \frac{j^n}{j!}$.

Tanny [14] used Rota's functional L and the celebrated Worpitzky identity $m^n = \sum_{k=0}^n \binom{n+m-k}{n} a_{n,k}$, to obtain the following expression for the ordered Bell numbers :

$$F_n(x) = \sum_{k=1}^n a_{n,k} x^{n-k+1} (1+x)^{k-1},$$

where $F_n(x)$ is the generating function of the number of ordered set partitions of the set $[n]$ and $a_{n,k}$ are Eulerian numbers, a.k.a. the number of permutations in the symmetric group S_n having k descents, see [9]. In modern terms, this expression can be seen as an evident to the fact that the ordered Bell numbers are gamma-positive.

In this paper, we extend some of Rota's and Tanny's results to the framework of the set partitions of Coxeter type B .

In Section 2, we recall the definition of set partitions (and ordered set partitions) of type B and introduce the definition of Bell numbers and Bell polynomials of type B . In Section 3, we extend the first two parts of Rota's Theorem 1.1 to Bell numbers of type B . Actually part (2) of that result will be proven both by Rota's method and by a counting argument. In Section 4, we use Brenti's generalization of Worpitzky's identity to obtain a gamma-positivity result for the ordered Bell polynomials of type B .

2 Set partitions and Bell numbers of type B

2.1 Set partitions of type B

We now recall the definition of set partitions of type B (see Dolgachev-Lunts [5, p. 755] and Reiner [10, Section 2]; mentioned implicitly in Dowling [6] and Zaslavsky [16] in the form of signed graphs):

Definition 2.1. *Denote: $[\pm n] := \{\pm 1, \dots, \pm n\}$. A set partition of $[n]$ of type B or a signed set partition is a set partition of the set $[\pm n]$ such that the following conditions are satisfied:*

- *If B appears as a block in the set partition, then $-B$ (which is obtained from B by negating all its elements) also appears in that partition.*
- *There exists at most one block satisfying $-B = B$. This block is called the zero block (if it exists, it is a subset of $[\pm n]$ of the form $\{\pm i \mid i \in C\}$ for some $C \subseteq [n]$).*

For example, the following is a set partition of $[6]$ of type B :

$$\{\{1, -1, 4, -4\}, \{2, 3, -5\}, \{-2, -3, 5\}, \{6\}, \{-6\}\}.$$

Note that every non-zero block B has a corresponding block $-B$ attached to it. For the sake of convenience, we write for the pair of blocks $B, -B$, only the representative block containing the minimal *positive* number appearing in $B \cup -B$. For example, the pair of blocks $B = \{-2, -3, 5\}, -B = \{2, 3, -5\}$ will be represented by the single block $\{2, 3, -5\}$.

Our convention will be to write first the zero block and denote it by B_0 if exists and then the non-zero blocks of a set partition of type B in such a way that the sequence of absolute values of the minimal elements of the blocks is increasing. We call this the *standard presentation*.

For example, the following is a set partition of $[6]$ of type B in its standard presentation:

$$\{B_0 = \{1, -1, 4, -4\}, B_1 = \{2, 3, -5\}, B_2 = \{6\}\}.$$

2.2 Stirling numbers and Bell numbers of type B

Definition 2.2. Let $S^B(n, k)$ be the number of set partitions of type B having k representative non-zero blocks. This is known as the Stirling number of type B of the second kind (see sequence A085483 in OEIS [7]).

It is easy to see that $S^B(n, n) = S^B(n, 0) = 1$ for each $n \geq 0$. The following recursion for $S^B(n, k)$ is well-known ([6, Theorem 7; see the Erratum], [3, Corollary 3], for $m = 2$, and [15, Equation (1)], for $m = 2, c = 1$):

Proposition 2.3. For each $1 \leq k < n$,

$$S^B(n, k) = S^B(n-1, k-1) + (2k+1)S^B(n-1, k). \quad (1)$$

The following result was proved combinatorially in [1] using a ‘balls into urns’ approach:

Theorem 2.4. Let $x \in \mathbb{R}$ and let $n \in \mathbb{N}$. Then we have:

$$x^n = \sum_{k=0}^n S^B(n, k)(x)_k^B, \quad (2)$$

where $(x)_k^B := (x-1)(x-3)\cdots(x-2k+1)$ and $(x)_0^B := 1$, called the falling factorial of type B .

Remark 2.5. There is a simple connection between the falling factorials of types A and B :

$$\left(\frac{x-1}{2}\right)_n = \left(\frac{x-1}{2}\right)\left(\frac{x-1}{2}-1\right)\cdots\left(\frac{x-1}{2}-n+1\right) = \frac{1}{2^n}(x-1)(x-3)\cdots(x-2n+1) = \frac{1}{2^n}(x)_n^B,$$

where the falling factorial of type A is defined as follows: $(x)_k := x(x-1)(x-2)\cdots(x-k+1)$.

We define the *Bell number of type B* as follows: $B_n^B = \sum_{k=0}^n S^B(n, k)$. Obviously, the Bell number counts all the set partitions of the set $[n]$ of type B . A similar definition appears in Mező and Ramírez [8].

We define also the *Bell polynomial of type B* as follows: $B_n^B(u) = \sum_{k=0}^n S^B(n, k)u^k$.

Sagan and Swanson [13] discussed *ordered set partitions of type B* , which are defined as follows:

Definition 2.6. An ordered set partition of $[n]$ of type B having k non-zero blocks is a sequence of sets $(S_0, S_1, \dots, S_{2k+1})$ which form a set partition of $[n]$ of type B , such that the following two order conditions are satisfied:

1. For each $1 \leq i \leq n$, $i \in S_0$ if and only if $-i \in S_0$ (i.e. the zero block S_0 is always the first block).
2. For each $1 \leq j \leq k$, we have $S_{2j} = -S_{2j-1}$.

Similar to the ordinary Bell number of type B , we define the *ordered Bell number of type B* and its associated *ordered Bell polynomial* as follows:

Definition 2.7. $B_n^{B,o} = \sum_{k=0}^n 2^k k! S^B(n, k)$ and $B_n^{B,o}(x) = \sum_{k=0}^n 2^k k! S^B(n, k) x^k$

3 Type-B analogue of Rota's result

Let V be the vector space of all polynomials in the variable x . It is easy to see that the set $\{(x)_k^B\}_{k \in \mathbb{Z}}$ is a basis of V . We use this basis to define a functional $L : V \rightarrow \mathbb{R}$ by $L((x)_k^B) = u^k$ for each $k \in \mathbb{Z}$, where u is a fixed real number.

We will use the following lemma in the proof of Theorem 3.2(2):

Lemma 3.1. *Let $p(x)$ be any polynomial. Then: $L((x-1)p(x-2)) = u \cdot L(p(x))$.*

Proof. Since the set of falling factorials of type B

$$\{1, (x-1), (x-1)(x-3), \dots, (x-1)(x-3) \cdots (x-2n+1)\}$$

is a basis for $\mathbb{R}_n[x]$, we can write any polynomial $p(x)$ as a linear combination of this set as follows:

$$p(x) = a_0 + a_1(x-1) + a_2(x-1)(x-3) + a_n(x-1)(x-3) \cdots (x-2n+1). \quad (3)$$

Substituting $x-2$ for x , we have now:

$$p(x-2) = a_0 + a_1(x-3) + a_2(x-3)(x-5) + \cdots + a_n(x-3)(x-5) \cdots (x-2n+3).$$

Multiplying the last equality by $(x-1)$ yields:

$$(x-1)p(x-2) = a_0(x-1) + a_1(x-1)(x-3) + \cdots + a_n(x-1) \cdots (x-2n+3). \quad (4)$$

Applying the operator L on Equations (3) and (4), we get the desired result. \square

We present here the analogue for type B of the celebrated results by Rota [12] (for part (2), see also Mező and Ramírez [8, p. 258]):

Theorem 3.2. *For each $n \in \mathbb{N}$, we have:*

(1) $B_n^B(u) = L(x^n)$.

(2) $B_{n+1}^B(u) = B_n^B(u) + u \cdot \sum_{j=0}^n 2^{n-j} \binom{n}{j} B_j^B(u)$.

Proof. (1) By Theorem 2.4, we have $x^n = \sum_{P \in \Pi_n^B} (x)_{N(P)}^B$, where Π_n^B is the set of set partitions of the set $[n]$ of type B and $N(P)$ is the number of non-zero representative blocks in the set partition P . Now apply the functional L on both sides of this equation and use the linearity of L to get

$$L(x^n) = \sum_{P \in \Pi_n^B} L\left((x)_{N(P)}^B\right) = \sum_{P \in \Pi_n^B} u^{N(P)} = \sum_{k=0}^n S^B(n, k) u^k = B_n^B(u).$$

(2) For this result, we supply two different proofs: an algebraic one and a combinatorial one for $u = 1$. We start with the algebraic proof: Applying Lemma 3.1 to $p(x) = (x+2)^n$, we get:

$$\begin{aligned} B_{n+1}^B(u) - B_n^B(u) &= L(x^{n+1}) - L(x^n) = L((x-1)x^n) \stackrel{\text{Lemma 3.1}}{=} u \cdot L((x+2)^n) = \\ &= u \cdot L\left(\sum_{j=0}^n \binom{n}{j} x^j 2^{n-j}\right) = u \cdot \sum_{j=0}^n \binom{n}{j} 2^{n-j} L(x^j) = u \cdot \sum_{j=0}^n \binom{n}{j} 2^{n-j} B_j^B(u). \end{aligned}$$

The combinatorial proof for $u = 1$ is as follows: The left hand side counts the total number of set partitions of the set $[n]$ of type B for any number of non-zero representative blocks. we show that the right hand side counts the same thing in a different way. We divide in two cases according to the location of $n+1$:

1. If $n + 1$ is located in the zero-block, then we have B_n^B possibilities to locate the other elements.
2. Otherwise, $n + 1$ is located in a non-zero block. Then, for each $0 \leq k \leq n$, assume that $n - k$ is the number of elements that share a block with $n + 1$. The rest k elements can be located in B_k^B ways. Finally, we have 2^{n-k} possibilities to sign the elements in the block containing $n + 1$.

□

4 Gamma-positivity of ordered Bell polynomial of type B

Let V be the vector space of all polynomials in the variable x . It is easy to see that the set $\{(x)_k^B\}_{k \in \mathbb{Z}}$ is a basis of V . We use this basis to define a new functional $L^o : V \rightarrow \mathbb{R}$ by

$$L^o((x)_k^B) = 2^k k! u^k \tag{5}$$

for each $k \in \mathbb{Z}$, where u is a fixed real number. By slightly modifying the proof of Theorem 3.2 we get that $L^o(x^n) = B_n^{B,o}(u)$.

Brenti [4] obtained a Worpitzky-like identity for type B as follows: For each $m, n \in \mathbb{N}$, one has

$$(1 + 2m)^n = \sum_{k=0}^n \binom{n+m-k}{n} E_{n,k}^B,$$

where $E_{n,k}^B$ are the Eulerian numbers of type B , which counts the number of signed permutations in the Coxeter group of type B having k descents; this set of numbers constitutes the sequence A060187 in OEIS [7]. For a combinatorial proof of this identity, see [2].

The following result shows the gamma-positivity of the ordered Bell polynomials of type B :

Proposition 4.1. $B_n^{B,o}(u) = \sum_{k=0}^n E_{n,k}^B u^k (1 + u)^{n-k}$.

Proof. If we write $x = 1 + 2m$, then we have:

$$\begin{aligned} B_n^{B,o}(u) &= L^o(x^n) = \sum_{k=0}^n L^o \left[\binom{n+m-k}{n} \right] E_{n,k}^B = \\ &= \sum_{k=0}^n L^o \left[\binom{n + \frac{x-1}{2} - k}{n} \right] E_{n,k}^B = \sum_{k=0}^n \frac{E_{n,k}^B}{n!} L^o \left[\binom{n + \frac{x-1}{2} - k}{n} \right] = \\ &\stackrel{(*)}{=} \sum_{k=0}^n \frac{E_{n,k}^B}{n!} L^o \left[\sum_{\ell=0}^n \binom{n}{\ell} (n-k)_{n-\ell} \left(\frac{x-1}{2} \right)_\ell \right] = \\ &\stackrel{\text{Rem. 2.5}}{=} \sum_{k=0}^n \frac{E_{n,k}^B}{n!} \sum_{\ell=0}^n \binom{n}{\ell} (n-k)_{n-\ell} L^o \left[\frac{(x)_\ell^B}{2^\ell} \right] = \\ &\stackrel{\text{Eqn. (5)}}{=} \sum_{k=0}^n E_{n,k}^B \sum_{\ell=0}^n \binom{n}{\ell} \frac{(n-k)_{n-\ell}}{n!} \ell! u^\ell = \sum_{k=0}^n E_{n,k}^B \sum_{\ell=0}^n \binom{n-k}{n-\ell} u^\ell = \\ &= \sum_{k=0}^n E_{n,k}^B u^k \sum_{\ell=0}^n \binom{n-k}{\ell-k} u^{\ell-k} = \sum_{k=0}^n E_{n,k}^B u^k (1 + u)^{n-k}, \end{aligned}$$

where Equality (*) is based on the binomial Umbral identity of the falling factorials of type A :

$$(a + b)_n = \sum_{k=0}^n \binom{n}{k} (a)_k (b)_{n-k}, \text{ see e.g. Roman [11, p. 29].}$$

□

References

- [1] E. Bagno & D. Garber (2022): *Signed partitions - A 'balls into urns' approach*. *Bull. Math. Soc. Sci. Math. Roumanie* 65 (113)(1), pp. 63–71.
- [2] E. Bagno, D. Garber & M. Novick (2022): *The Worpitzky identity for the groups of signed and even-signed permutation*. *J. Alg. Combin.* 55(2), pp. 413–428, doi:10.1007/s10801-021-01056-4.
- [3] M. Benoumhani (1996): *On Whitney numbers of Dowling lattices*. *Discrete Math.* 159(1-3), pp. 13–33, doi:10.1016/0012-365X(95)00095-E.
- [4] F. Brenti (1994): *q-Eulerian polynomials arising from Coxeter groups*. *Europ. J. Combin.* 15(5), pp. 417–441, doi:10.1006/eujc.1994.1046.
- [5] I. Dolgachev & V. Lunts (1994): *A character formula for the representation of a Weyl group in the cohomology of the associated toric variety*. *J. Alg.* 168(3), pp. 741–772, doi:10.1006/jabr.1994.1251.
- [6] T. A. Dowling (1973): *A class of geometric lattices based on finite groups*. *J. Combin. Theory, Ser. B* 14(1), pp. 61–86, doi:10.1016/S0095-8956(73)80007-3. Erratum: *J. Combin. Theory, Ser. B* 15 (1973), 211.
- [7] OEIS Foundation Inc. (2022): *The On-Line Encyclopedia of Integer Sequences*. Available at <http://oeis.org>.
- [8] I. Mező & J.L. Ramírez (2022): *Set partitions and partitions without singleton blocks of type B*. *Period. Math. Hungarica* 85(2), pp. 246–263, doi:10.1007/s10998-021-00439-1.
- [9] T. K. Petersen (2015): *Eulerian numbers*. Birkhauser, Basel, doi:10.1007/978-1-4939-3091-3.
- [10] V. Reiner (1997): *Non-crossing partitions for classical reflection groups*. *Discrete Math.* 177(1–3), pp. 195–222, doi:10.1016/S0012-365X(96)00365-2.
- [11] S. Roman (1984): *The Umbral Calculus*. Academic Press, doi:10.1007/978-1-4939-3091-3.
- [12] G.-C. Rota (1964): *The number of partitions of a set*. *Amer. Math. Monthly* 71(5), pp. 498–504, doi:10.1080/00029890.1964.11992270.
- [13] B. Sagan & J. Swanson (2024): *q-Stirling numbers in type B*. *Europ. J. Combin.* 118, p. 103899, doi:10.1016/j.ejc.2023.103899.
- [14] S.M. Tanny (1975): *On some numbers related to the Bell numbers*. *Canad. Math. Bull.* 17(5), pp. 733–738, doi:10.4153/CMB-1974-132-8.
- [15] D. G. L. Wang (2014): *On colored set partitions of type B_n* . *Cent. Eur. J. Math.* 12(9), pp. 1372–1381, doi:10.1007/s10998-021-00439-1. Available at <http://eudml.org/doc/269733>.
- [16] T. Zaslavsky (1981): *The geometry of root systems and signed graphs*. *Amer. Math. Monthly* 88(2), pp. 88–105, doi:10.1080/00029890.1981.11995201.

Dyck Paths Enumerated by the \mathbb{Q} -bonacci Numbers

Elena Barucci Antonio Bernini Stefano Bilotta Renzo Pinzani

Dipartimento di Matematica e Informatica “U. Dini”

Università degli Studi di Firenze
Viale G.B. Morgagni 65, 50134 Firenze, Italy.

elena.barucci@unifi.it
antonio.bernini@unifi.it
stefano.bilotta@unifi.it
renzo.pinzani@unifi.it

1 Introduction

The q -generalized Fibonacci numbers [5, 7] can be combinatorially interpreted in many different ways, for $q \in \mathbb{N}^+$. One of them involves the Dyck paths having height at most two. More precisely, the subsets of these bounded Dyck paths avoiding q consecutive valleys at height 1 are enumerated according to their semilength (i.e. the number of steps of the path divided by 2) by the mentioned sequence [1, 2]. The same numbers count the binary strings avoiding $q+1$ consecutive 1’s, with $q \geq 1$, according to their length [4, 8]. Note that to be exact we should write that the enumerating sequences are the $(q+1)$ -generalized Fibonacci numbers, for $q \geq 1$.

Not long ago Baril, Kirgizov, and Vajnovszki [3] introduced the set \mathcal{W}_n^q of the q -decreasing strings which are binary strings of length n where each maximal factor $0^a 1^b$, with $a > 0$, satisfies $q \cdot a > b$, for $q \geq 1$. Moreover, among other results, the authors give a bijection between \mathcal{W}_n^q and the set $\mathcal{B}_n(1^{q+1})$ of the binary strings of length n avoiding $q+1$ consecutive 1’s, for $q \geq 1$.

Recently [6], Kirgizov generalized the q -decreasing strings to the case where q is a positive rational number, $q \in \mathbb{Q}^+$, and they are enumerated by the numbers called \mathbb{Q} -bonacci by the author. In this paper we provide a class of restricted Dyck paths that result in having the same enumeration (according to their semilength). More precisely, we consider the Dyck paths having height at most two and introduce some constraints on the numbers of consecutive valleys at height one which must be followed by a suitable number of valleys at height zero, depending on the value of $q \in \mathbb{Q}^+$.

2 Preliminaries

In the paper we indicate a Dyck path in linear notation as a string over the alphabet $\{U, D\}$, where U and D replace the up and the down steps of the path, respectively. We figure a Dyck path in a Cartesian coordinate system, starting from the origin and ending in a point of the x -axis. A valley is a substring DU , while a peak is a substring UD . The height of a valley is the ordinate reached by the D step. We refer to a valley at height 1 or at height 0 with 1 -valley or 0 -valley, respectively. For the peaks, with 1 -peak (or 0 -peak) we mean a peak UD whose D step reaches the ordinate 1 (or 0).

Since we are going to deal only with Dyck paths having height at most 2, we do not record this restriction in our notation.

Let \mathcal{D}_n^q , with $q \geq 1$, denote the set of the Dyck paths having height at most 2, and avoiding $q+1$ consecutive 1 -peaks (which is the same as avoiding q consecutive 1 -valleys), and having semilength n ,

where $n \geq 0$.

Clearly, a Dyck path $P \in \mathcal{D}_n^q$ starts with one of the factors $UD, UUDD, UUDUDD, \dots, U(UD)^q D$ (where $(UD)^q$ is the string obtained by concatenating UD to itself q times). Then the path P is obtained by concatenating one of these factors to a path of suitable length, and the set \mathcal{D}_n^q can be generated by

$$\mathcal{D}_n^q = \begin{cases} \varepsilon, & \text{if } n = 0; \\ \bigcup_{j=0}^q U(UD)^j D \cdot \mathcal{D}_{n-1-j}^q, & \text{if } n \geq 1. \end{cases}$$

It is thus enumerated by the sequence of the $(q+1)$ -generalized Fibonacci numbers

$$f_n^{(k)} = \begin{cases} 1, & \text{if } n = 0; \\ \sum_{i=1}^k f_{n-i}^{(k)}, & \text{if } n \geq 1 \quad (f_\ell^{(k)} = 0 \text{ if } \ell < 0). \end{cases}$$

Note that for $q = 1$ we get the classical Fibonacci numbers.

3 Construction in the case $q \in \mathbb{Q}^+$

3.1 The particular case $q = 1/s$

If $P \in \mathcal{D}_n^q$ with $q \in \mathbb{N}^+$, then q consecutive 1-peaks in P are necessarily followed by at least one 0-valley. In the case where $q \in \mathbb{Q}^+$, we require that the number of 0-valleys have some constraints.

We start with the particular value $q = 1/s$. In this case we impose that each 1-peak of a Dyck path P must be followed by at least s consecutive 0-valleys if after the 1-peak there is enough space to contain s consecutive 0-valleys. If a 1-peak occurs near the end of P and there is no space to contain s consecutive 0-valleys, then no other 1-peaks can occur up to the end of P . Summarizing, we give the following definition.

Definition 3.1 Let $\mathcal{D}_n^{1/s}$ denote the set of Dyck paths P of semilength n having height at most 2, where either P has no 1-peaks ($P = (UD)^n$) or each 1-peak in P is followed by at least s consecutive 0-valleys, except the last 1-peak which can be followed by less than s consecutive 0-valleys.

The construction is straightforward: a path $P \in \mathcal{D}_n^{1/s}$ starts with one of the UD or $UUD(DU)^{s-1}D$ factors (or P is a suitable prefix of this last one factor, if $n \leq s$), then P is obtained by concatenating one of these factor to a Dyck path $Q \in \mathcal{D}_{n-1}^{1/s}$ or $Q \in \mathcal{D}_{n-s-1}^{1/s}$. In the case P begins with the longer factor, since Q starts, of course, with an up step U , then really the first 1-peak of P is followed by s consecutive 0-valleys (so that $P \in \mathcal{D}_n^{1/s}$): the first $s-1$ ones are the $s-1$ consecutive 0-valleys of the factor, and the last one is given by the last step D of the factor and the first U step of Q .

Not considering, for the moment, paths with a semilength less or equal to s , we can write

$$\mathcal{D}_n^{1/s} = UD \cdot \mathcal{D}_{n-1}^{1/s} \cup UUD(DU)^{s-1}D \cdot \mathcal{D}_{n-s-1}^{1/s} \quad (\text{for } n > s). \quad (1)$$

Thus, the set $\mathcal{D}_n^{1/s}$ is enumerated by

$$w_n = w_{n-1} + w_{n-s-1}$$

omitting at this stage the initial conditions.

We note that this recurrence relation matches the one enumerating the q -decreasing strings in the case $q = 1/s$ stated in [6].

As far as the initial conditions are concerned, we observe that prepending the factor UD generates Dyck paths having semilength starting from 1, so that the empty Dyck path ε must be considered as a legal path of $\mathcal{D}_0^{1/s}$ (actually, the only one!).

Prepending the factor $UUD(DU)^{s-1}D$ generates Dyck paths having semilength starting from $s+1$. For semilengths less or equal to s , we note that the construction described in equation (1) does not generate the paths $UUD, UUDUD, \dots, UUD(DU)^tD$ with $t = 0, 1, \dots, s-2$. These paths are suitable prefixes of $UUD(DU)^{s-1}D$ which however satisfy Definition 3.1, so that they must be considered among the initial conditions.

Therefore, the generation of the set $\mathcal{D}_n^{1/s}$ can be completely described as follows:

$$\mathcal{D}_n^{1/s} = \begin{cases} \varepsilon, & \text{if } n = 0; \\ UD, & \text{if } n = 1; \\ UD \cdot \mathcal{D}_{n-1}^{1/s} \cup U \cdot p_{n-1}(UD(DU)^{s-1}) \cdot D, & \text{if } 2 \leq n \leq s+1; \\ UD \cdot \mathcal{D}_{n-1}^{1/s} \cup UUD(DU)^{s-1}D \cdot \mathcal{D}_{n-s-1}^{1/s}, & \text{if } n > s+1. \end{cases}$$

In the above formula $p_{n-1}(UD(DU)^{s-1})$ is the prefix of semilength $n-1$ of $UD(DU)^{s-1}$.

It is not difficult to see that $\mathcal{D}_n^{1/s}$ is enumerated by

$$w_n = \begin{cases} 1, & \text{if } n = 0; \\ n, & \text{if } 1 \leq n \leq s+1; \\ w_{n-1} + w_{n-s-1}, & \text{if } n > s+1. \end{cases}$$

This sequence matches the one enumerating the q -decreasing strings for $q = 1/s$ that can be deduced from [6].

3.2 The general case $q = r/s$

Following the outline of the constructions in the cases where q is an integer, and where $q = 1/s$, in the general case $q = r/s$ (we suppose r and s to be coprime) we require that a path P avoid $r+1$ consecutive 1-peaks, and if r consecutive 1-peaks occur in P , then they must be followed by at least s consecutive 0-valleys. Clearly, we have to deal with the case where p consecutive 1-peaks, with $p = 1, 2, \dots, r-1$, occur in P . When this happens, we impose that the p consecutive 1-peaks must be followed by a number v of consecutive 0-valleys such that

$$\frac{p}{v} \leq \frac{r}{s} \quad \text{for } p = 1, 2, \dots, r-1. \quad (2)$$

Moreover, we have to deal with the case where the rightmost block of r consecutive 1-peaks occurs near the end of P (more precisely, when after this block there is no space to contain s consecutive 0-valleys). In this case, no other 1-peaks can occur up to the end of P . Finally, we allow the paths end with a consecutive block of 1-peaks (clearly, less than $r+1$). Summarizing, we give the following definition.

Definition 3.2 Let $\mathcal{D}_n^{r/s}$ denote the set of Dyck paths P of semilength n having height at most 2, where

- each block B of r consecutive 1-peaks in P is followed by at least s consecutive 0-valleys, except the rightmost block B which can be followed by less than s consecutive 0-valleys, and

- each block C of p consecutive 1-peaks in P , with $p = 1, 2, \dots, r-1$, is followed by at least ν consecutive 0-valleys such that $p/\nu \leq r/s$, and
- the path P can end with $(UD)^t D$, with $t = 1, 2, \dots, r$ (in other words, P ends with t consecutive 1-peaks ($t \leq r$) followed by a down step).

Clearly, the number ν of consecutive 0-valleys, since ν is an integer, satisfies our request (2) when $\nu \geq \lceil p \cdot \frac{s}{r} \rceil$. Moreover, we note that in the case where p consecutive 1-peaks, with $p = 1, 2, \dots, r-1$, occur near the end of P and there is not enough space to contain ν consecutive 0-valleys, then the path P does not belong to $\mathcal{D}_n^{r/s}$, according to the second bullet in Definition (3.2). For example, if $q = 4/5$, the path $P = UUDUDDUDUD$ is not allowed, since after two consecutive 1-peaks ($p = 2$) at least three consecutive 0-valleys ($\nu = 3$, according to request (2)) must occur. On the other hand, the path $P = UUDUDUDUDDUDUDUD$ is allowed.

Also in this (general) case, the construction is straightforward. A path $P \in \mathcal{D}_n^{r/s}$ starts with one of the factors UD or $U(UD)^p(DU)^{\lceil ps/r \rceil - 1}D$, with $p = 1, 2, \dots, r$ (or P is a suitable prefix of $U(UD)^r(DU)^{s-1}D$). With an argument similar to the one used in the case where $q = 1/s$, it is not difficult to get the following construction:

$$\mathcal{D}_n^{r/s} = \begin{cases} \varepsilon, & \text{if } n = 0; \\ UD, & \text{if } n = 1; \\ UD \cdot \mathcal{D}_{n-1}^{r/s} \cup U \cdot p_{n-1}((UD)^r(DU)^{s-1}) \cdot D \\ \cup U(UD)^p(DU)^{\lceil ps/r \rceil - 1}D \cdot \mathcal{D}_{n-p-\lceil ps/r \rceil}^{r/s}, & \text{if } 2 \leq n \leq r+s \\ & \text{and } n-p-\lceil ps/r \rceil \geq 1 \\ & \text{with } 1 \leq p \leq r-1; \\ UD \cdot \mathcal{D}_{n-1}^{r/s} \cup U(UD)^p(DU)^{\lceil ps/r \rceil - 1}D \cdot \mathcal{D}_{n-p-\lceil ps/r \rceil}^{r/s}, & \text{if } n > r+s \\ & \text{and } 1 \leq p \leq r; \end{cases}$$

Then we have (in the following $\chi(f) = 1$ if f is true, and $\chi(f) = 0$ otherwise)

$$w_n = \begin{cases} 1, & \text{if } n = 0; \\ 1, & \text{if } n = 1; \\ w_{n-1} + 1 + \sum_{p=1}^{r-1} \chi(n-p-\lceil ps/r \rceil \geq 1) w_{n-p-\lceil ps/r \rceil}, & \text{if } 2 \leq n \leq r+s; \\ w_{n-1} + \sum_{p=1}^r w_{n-p-\lceil ps/r \rceil}, & \text{if } n > r+s. \end{cases}$$

In order to respect the second bullet in Definition 3.2 we added the factor $\chi(n-p-\lceil ps/r \rceil \geq 1)$ in the case $2 \leq n \leq r+s$ of the definition of w_n , and the statement $n-p-\lceil ps/r \rceil \geq 1$ since at least $\lceil ps/r \rceil$ consecutive 0-valleys must occur after p consecutive 1-peaks, with $1 \leq p \leq r-1$ (in order to have $\mathcal{D}_{n-p-\lceil ps/r \rceil}^{r/s} \neq \emptyset$).

Also, in this case, the recurrence relations match the ones enumerating the q -decreasing strings in the case $q = r/s$ founded in [6].

4 Acknowledgements

This work is partially supported by the INdAM – GNCS Project 2023 “Aspetti combinatori ed enumerativi di strutture discrete: stringhe, ipergrafi e permutazioni”, code CUP_E53C22001930001.

References

- [1] E. Barcucci, A. Bernini & R. Pinzani (2022): *From the Fibonacci to Pell numbers and beyond via Dyck paths*. *Pure Math. Appl. (P.U.M.A.)* 30, pp. 17–22, doi:10.2478/puma-2022-0004.
- [2] E. Barcucci, A. Bernini & R. Pinzani (2024): *Sequences from Fibonacci to Catalan: A combinatorial interpretation via Dyck paths*. *RAIRO Theor. Inform. Appl.* 58:8, doi:10.1051/ita/2024007.
- [3] J.-L. Baril, S. Kirgizov & V. Vajnovszki (2022): *Gray codes for Fibonacci q -decreasing words*. *Theoret. Comput. Sci.* 927, pp. 120–132, doi:10.1016/j.tcs.2022.06.003.
- [4] A. Bernini (2017): *Restricted binary strings and generalized Fibonacci numbers*. In A. Dennunzio, E. Formenti, L. Manzoni & A. Porreca, editors: *Cellular Automata and Discrete Complex Systems. AUTOMATA 2017. Lecture Notes in Computer Science*, 10248, Springer International Publishing, Cham, pp. 32–43, doi:10.1007/978-3-319-58631-1_3.
- [5] M. Feinberg (1963): *Fibonacci-Tribonacci*. *Fibonacci Quart.* 1, pp. 71–74.
- [6] S. Kirgizov (2022): *\mathbb{Q} -bonacci numbers and words*. *Fibonacci Quart.* 60(5), pp. 187–195.
- [7] E. P. Miles Jr. (1960): *Generalized Fibonacci numbers and associated matrices*. *Amer. Math. Monthly* 67, pp. 745–752, doi:10.1080/00029890.1960.11989593.
- [8] V Vajnovszki (2001): *A loopless generation of bitstrings without p consecutive ones*. In C. S. Calude, M. J. Dinneen & S. Sburlan, editors: *Combinatorics, Computability and Logic. Discrete Mathematics and Theoretical Computer Science*, Springer, London, pp. 227–240, doi:10.1007/978-1-4471-0717-0_19.

Diagram Calculus for the Affine Temperley–Lieb Algebra of Type D

Riccardo Biagioli

Università di Bologna
Bologna, Italy

riccardo.biagioli2@unibo.it

Giuliana Fatabbi

Università degli Studi di Perugia
Perugia, Italy

giuliana.fatabbi@unipg.it

Elisa Sasso

Università di Bologna
Bologna, Italy

elisa.sasso2@unibo.it

Let (W, S) be a Coxeter system of affine type \tilde{D} , and let $\text{TL}(W)$ the corresponding generalized Temperley-Lieb algebra. In this extended abstract we define an infinite dimensional associative algebra made of decorated diagrams which is isomorphic to $\text{TL}(W)$. Moreover, we describe an explicit basis for such an algebra of diagrams which is in bijective correspondence with the classical monomial basis of $\text{TL}(W)$, indexed by the fully commutative elements of W .

1 Introduction

The Temperley-Lieb algebra is a very classical mathematical object studied in algebra, combinatorics, statistical mechanics and mathematical physics, introduced by Temperley and Lieb in 1971 [13]. Thanks to Kauffman [11] and Penrose [12], it was showed that the Temperley-Lieb algebra can be realized as a diagram algebra, that is an associative algebra with a basis given by certain diagrams on the plane. On the other hand, Jones presented the Temperley-Lieb algebra in terms of abstract generators and relations. In [10], he also showed that this algebra occurs naturally as a quotient of the Hecke algebra of type A . The realization of the Temperley-Lieb algebra as a Hecke algebra quotient was generalized by Graham in [5]. He defined the so-called generalized Temperley Lie algebra $\text{TL}(\Gamma)$ for any Coxeter system of type Γ and showed that $\text{TL}(\Gamma)$ has a monomial basis indexed by the fully commutative elements of the underlying Coxeter group.

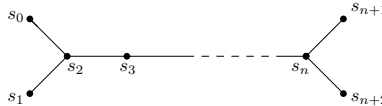


Figure 1: Coxeter graph of type \tilde{D}_{n+2} .

During the years, diagrammatic representations for $\text{TL}(\Gamma)$ have been found for each Coxeter system of finite type but only for two systems of affine type. More precisely, in [7], [9] and [6] Green defined a diagram calculus in finite Coxeter types B, D, E and H . For affine types, in [4] Fan and Green provided a realization of $\text{TL}(\tilde{A})$ as a diagram algebra on a cylinder and, more recently, in [2, 3] Ernst represented $\text{TL}(\tilde{C})$ as an algebra of decorated diagrams. In this extended abstract, we present a new diagrammatic representation for $\text{TL}(\tilde{D}_{n+2})$. Our method can be extended also to the affine case \tilde{B} . Here we recall the presentation of $\text{TL}(\tilde{D}_{n+2})$ given by Green in [8] that we consider as definition: $\text{TL}(\tilde{D}_{n+2})$ is the $\mathbb{Z}[\delta]$ -algebra generated by $\{b_0, b_1, \dots, b_{n+2}\}$ with defining relations:

(d1) $b_i^2 = \delta b_i$ for all $i \in \{0, \dots, n+2\}$,

(d2) $b_i b_j = b_j b_i$ if s_i and s_j are not adjacent nodes in the Coxeter graph of type \tilde{D}_{n+2} ;

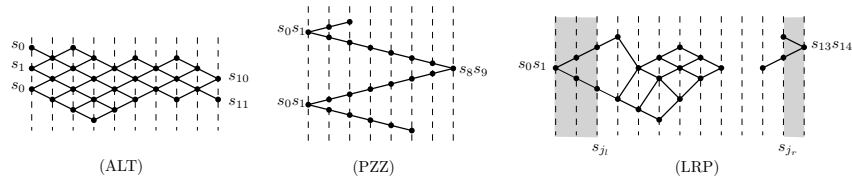


Figure 2: Some fully commutative heaps.

(d3) $b_i b_j b_i = b_i$ if s_i and s_j are adjacent nodes in the Coxeter graph of type \tilde{D}_{n+2} .

Similarly to Ernst in [2, 3], we define the diagrams for our representation starting from the classical ones of type A and adding decorations on the edges.

Then, as usual, we define a product of decorated diagrams by concatenation. This operation turns this set into an infinite dimensional $\mathbb{Z}[\delta]$ -algebra, of which we consider a quotient modulo some new relations. Within this quotient, we consider a specific $\mathbb{Z}[\delta]$ -subalgebra called *algebra of admissible diagrams* and denoted by $\mathbb{D}(\tilde{D}_{n+2})$. The main result of this extended abstract states that $\mathbb{D}(\tilde{D}_{n+2})$ and $\text{TL}(\tilde{D}_{n+2})$ are isomorphic $\mathbb{Z}[\delta]$ -algebras.

2 Fully commutative elements of Coxeter groups

Let M be a square symmetric matrix indexed by a finite set S , satisfying $m_{ss} = 1$ and, for $s \neq t$, $m_{st} = m_{ts} \in \{2, 3, \dots\} \cup \{\infty\}$. The *Coxeter group* W associated with the *Coxeter matrix* M is defined by generators S and relations $(st)^{m_{st}} = 1$ if $m_{st} < \infty$. These relations can be rewritten more explicitly as $s^2 = 1$ for all s , and

$$\underbrace{sts \cdots}_{m_{st}} = \underbrace{tst \cdots}_{m_{st}},$$

where $m_{st} < \infty$, the latter being called *braid relations*. When $m_{st} = 2$, they are simply *commutation relations* $st = ts$. For $w \in W$, the *length* of w , denoted by $\ell(w)$, is the minimum length l of an expression $s_1 \cdots s_l$ of w with $s_i \in S$. The expressions of length $\ell(w)$ are called *reduced*.

Definition 2.1 *An element $w \in W$ is fully commutative (FC) if any reduced expression of w can be obtained from any other reduced expression of w using only commutation relations.*

The concept of heap helps in studying problems related to full commutativity, for more details see for instance [1]. Briefly, given a reduced expression of $w = s_{i_1} \cdots s_{i_k} \in W$, its *heap* is a poset on the index set $\{1, \dots, k\}$ together with a labeling map. Heaps can be represented via Hasse diagrams; moreover, if $w \in \text{FC}$, its heap does not depend on its reduced expression. Fully commutative elements heaps of type \tilde{D}_{n+2} have been classified in [1, §3.2]: they can be split in five disjoint families, depending on the shapes of their associated heaps, whose elements are respectively called Alternating elements (ALT), Left-Peaks (LP), Right-Peaks (RP), Left-Right-Peaks (LRP) and Pseudo-Zigzags (PZZ), see Figure 2.

3 Decorated diagrams

A *concrete pseudo k -diagram* consists of a finite number of disjoint plane curves, called *edges*, embedded in a box having k nodes on the top (*north*) face and k nodes on the bottom (*south*) face. The nodes are endpoints of edges and all other embedded edges must be closed (isotopic to circles) and disjoint from

the box. We refer to a closed edge as a *loop*. It follows that there cannot exist isolated nodes and from each node a single edge starts. By $\{a, b\}$ we mean an edge that joins the node a to the node b .

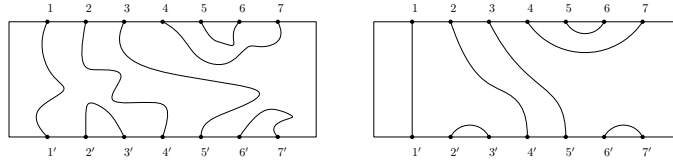


Figure 3: Two equivalent concrete pseudo 7-diagrams.

We say that two concrete pseudo k -diagrams are (*isotopically*) *equivalent* if one can be obtained from the other by isotopically deforming the edges such that any intermediate diagram is also a concrete pseudo k -diagram (Figure 3). We define a *pseudo k -diagram* as an equivalence class of concrete pseudo k -diagrams with respect to isotopic equivalence. Given two of these diagrams D, D' , we define the product $D'D$ as the pseudo k -diagram obtained by placing D' on top of D so that node i of D coincides with node i' of D' and then rescaling.

Now let D be a concrete pseudo k -diagram. Consider the set $\Omega = \{\bullet, \circ\}$ and the monoid Ω^* . Our goal is to adorn the edges of D with elements of Ω which we call *decorations*. In particular, \bullet is called a *L-decoration* and \circ is called a *R-decoration*. We call a *LR-decorated pseudo k -diagram* a pseudo k -diagram decorated with these decorations up to certain rules that we do not list here. We denote the set of LR-decorated pseudo k -diagrams by $T_k^{LR}(\Omega)$ and define $\mathcal{P}_k^{LR}(\Omega)$ to be the $\mathbb{Z}[\delta]$ -module having the elements of $T_k^{LR}(\Omega)$ as a basis.

As before, we define multiplication in $\mathcal{P}_k^{LR}(\Omega)$ by concatenating two basis elements and then extend it bilinearly, see the first equality in Figure 4. We can show that this product gives a structure of $\mathbb{Z}[\delta]$ -

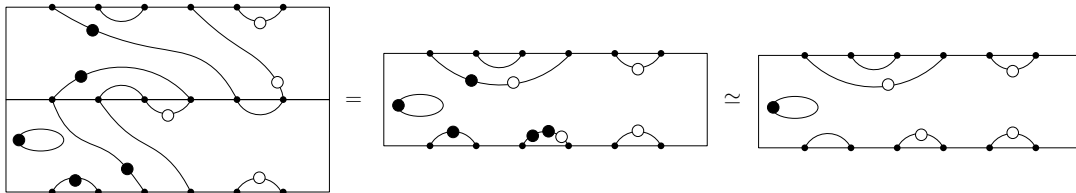


Figure 4: Product of two concrete decorated pseudo diagrams and its reduction.

algebra to $\mathcal{P}_k^{LR}(\Omega)$, which is an infinite dimensional algebra.

Let $\widehat{\mathcal{P}}_k^{LR}(\Omega)$ be the $\mathbb{Z}[\delta]$ -quotient algebra of $\mathcal{P}_k^{LR}(\Omega)$ by the relations in Figure 5. We say that a

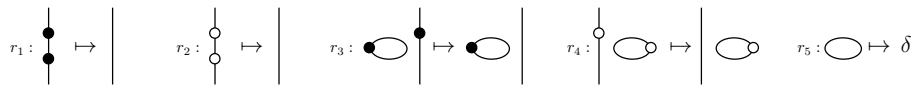


Figure 5: The defining relations of $\widehat{\mathcal{P}}_k^{LR}(\Omega)$.

LR-decorated diagram is *irreducible* if there are no relations to apply. Similarly to [2, Proposition 3.4.1], one can prove that the set of LR-decorated irreducible diagrams forms a basis for $\widehat{\mathcal{P}}_k^{LR}(\Omega)$. An example of irreducible diagram is in Figure 4, right.

We are particularly interested in a special subset of irreducible diagrams, called the *simple diagrams* D_0, \dots, D_{n+2} , defined as in Figure 6. It is easy to prove that the simple diagrams satisfy the relations (d1)-

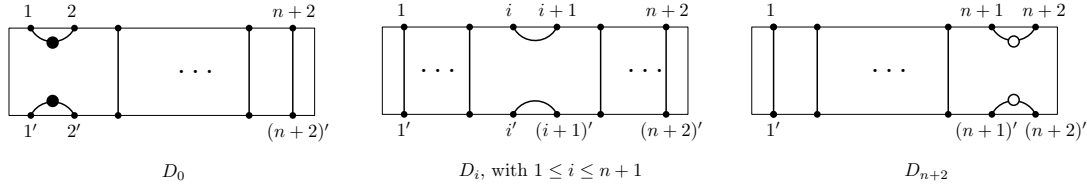


Figure 6: The simple diagrams.

(d3) of the $\text{TL}(\tilde{D}_{n+2})$ generators, defined in Section 1, simply replacing b_i by D_i . Denote by $\mathbb{D}(\tilde{D}_{n+2})$ the $\mathbb{Z}[\delta]$ -subalgebra of $\widehat{\mathcal{P}}_{n+2}^{LR}(\Omega)$ generated as a unital algebra by the simple diagrams with multiplication inherited by $\widehat{\mathcal{P}}_{n+2}^{LR}(\Omega)$.

3.1 Admissible diagrams

We consider a subset of irreducible diagrams called *admissible diagrams* and denoted by $\text{ad}(\tilde{D}_{n+2})$; here we state some fundamental properties the admissible diagrams must satisfy:

1. the only loop edges that can occur are depicted in Figure 7;

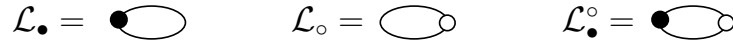


Figure 7: Allowable loops in \tilde{D} -admissible diagrams.

2. the total number of $\mathcal{L}_{\bullet}^{\circ}$ and of \bullet on non-loop edges must be even, as the total number of $\mathcal{L}_{\bullet}^{\circ}$ and of \circ on non-loop edges must be even too;
3. there cannot be a \circ -decoration to the left of a \bullet -decoration and vice versa.

We divide the admissible diagrams in five disjoint families, based on the displacement of the decorations on the edges (ALT, LP, RP, LRP and PZZ-diagrams). As one can guess, there will be a correspondence between this diagram classification and the heaps classification in Section 2. Moreover we give a definition of the *length of a diagram* $\ell(D)$ that depends on the shape of the edges and on the number of decorations on them. We use the length of a diagram for the inductive argument of our main result. The following picture summarizes the several structures introduced above: the main result is described by the last equality.

$$T_{n+2}^{LR}(\Omega) \hookrightarrow \mathcal{P}_{n+2}^{LR}(\Omega) \twoheadrightarrow \widehat{\mathcal{P}}_{n+2}^{LR}(\Omega) \supset \mathbb{D}(\tilde{D}_{n+2}) = \text{ad}(\tilde{D}_{n+2}).$$

4 Cut and paste operation

In this section we present the major combinatorial technique we used to prove the faithfulness of the diagrammatic representation we just defined.

An edge on the north face e of $D \in \text{ad}(\tilde{D}_{n+2})$ joining two consecutive edges is called a *simple edge* if either (a) e is undecorated, or (b) $e = \{1, 2\}$ and it is decorated by a single \bullet , or (c) $e = \{n+1, n+2\}$ and it is decorated by a single \circ .

We consider a subset of simple edges, called *suitable* edges, and to each of those, we assign a *neighbor* edge which often is a $\mathcal{L}_\bullet^\circ$ or leaves one of the nodes adjacent to the suitable edge. Now we can introduce the following procedure, called *cut and paste operation*, an example is in Figure 8, right.

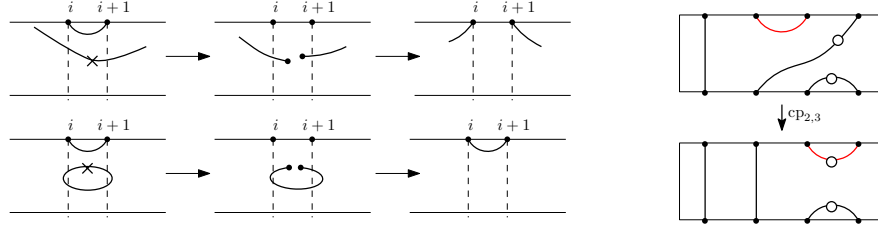


Figure 8: Cut and paste operation.

Definition 4.1 (Cut and paste operation) Let D be an admissible diagram with a suitable edge $e = \{i, i+1\}$.

- (cp1) Delete the simple edge e .
- (cp2) Cut the neighbor of e and join the two free endpoints of the cut edge to the nodes i and $i+1$ as to obtain two new non intersecting edges or the edge $\{i, i+1\}$, see Figure 8, left.
- (cp3) If the neighbor edge of e was decorated, then distribute its decorations on the new edges in an admissible way.

The importance of this procedure relies in the possibility of finding a unique diagram D' from another diagram D that factorizes as $D = D_i D'$ with the property that D' has length equal to $\ell(D) - 1$. Thanks to this inductive argument, we can prove the results stated in the next section. Moreover, this procedure provides an algorithm to factorize an admissible diagram into a product of simple diagrams.

5 Main results

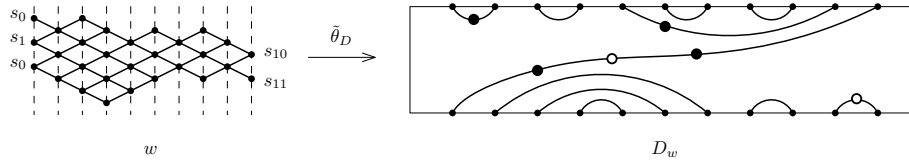
Let $s_{i_1} s_{i_2} \cdots s_{i_k}$ be a reduced expression of $w \in \text{FC}(\tilde{D}_{n+2})$ and define $D_w := D_{i_1} D_{i_2} \cdots D_{i_k}$. Note that D_w does not depend on the chosen reduced expression of w since $w \in \text{FC}(\tilde{D}_{n+2})$. Define the $\mathbb{Z}[\delta]$ -algebra homomorphism

$$\tilde{\theta}_D : \text{TL}(\tilde{D}_{n+2}) \rightarrow \mathbb{D}(\tilde{D}_{n+2}) \quad \text{such that} \quad \tilde{\theta}_D(b_i) = D_i$$

for all $i = 0, \dots, n+2$. Clearly $\tilde{\theta}_D$ is surjective and maps the monomial basis element $b_w := b_{i_1} b_{i_2} \cdots b_{i_k}$ into the diagram D_w . Our goal is to show that this map is actually an algebra isomorphism. The proof is based on five steps that are summarized in the next theorem. Point 4) is by induction on the length of an admissible diagram and it uses the cut and paste algorithm.

Theorem 5.1

1. Let w be a FC element of a certain type, then the image of the basis element b_w is a diagram of the analogous type and vice versa (for instance, w has a heap of type (ALT) if and only if $\tilde{\theta}_D(b_w) = D_w$ is an ALT-diagram, and so on).
2. The lengths of w and D_w are equal.



3. Every admissible diagram D is of the form D_w where $w \in \text{FC}(\tilde{D}_{n+2})$.
4. The admissible diagrams form a $\mathbb{Z}[\delta]$ -algebra that coincides with $\mathbb{D}(\tilde{D}_{n+2})$. Moreover, the set of admissible diagrams is a basis for $\mathbb{D}(\tilde{D}_{n+2})$.
5. The map $\tilde{\theta}_D$ is an algebra isomorphism.

References

- [1] R. Biagioli, F. Jouhet & P. Nadeau (2015): *Fully commutative elements in finite and affine Coxeter groups*. *Monatsh. Math.* 178(1), pp. 1–37, doi:10.1007/s00605-014-0674-7. arXiv:1402.2166.
- [2] D.C. Ernst (2012): *Diagram calculus for a type affine C Temperley–Lieb algebra, I*. *J. Pure Appl. Algebra* 216(11), pp. 2467–2488, doi:10.1016/j.jpaa.2012.03.013. arXiv:0910.0925.
- [3] D.C. Ernst (2018): *Diagram calculus for a type affine C Temperley–Lieb algebra, II*. *J. Pure Appl. Algebra* 222(12), pp. 3795–3830, doi:10.1016/j.jpaa.2018.02.008. arXiv:1101.4215.
- [4] C.K. Fan & R.M. Green (1999): *On the affine Temperley–Lieb algebras*. *J. London Math. Soc.* 60(2), pp. 366–380, doi:10.1112/S0024610799007796. arXiv:q-alg/9706003.
- [5] J.J. Graham (1995): *Modular Representations of Hecke Algebras and Related Algebras*. Ph.D. thesis, University of Sydney.
- [6] R.M. Green (1998): *Cellular algebras arising from Hecke algebras of type H_n* . *Math. Z.* 229(2), pp. 365–383, doi:10.1007/PL00004661. arXiv:q-alg/9712019.
- [7] R.M. Green (1998): *Generalized Temperley–Lieb algebras and decorated tangles*. *J. Knot Theory Ramifications* 7(2), pp. 155–171, doi:10.1142/S0218216598000103. arXiv:q-alg/9712018.
- [8] R.M. Green (2006): *Star reducible Coxeter groups*. *Glasg. Math. J.* 48(3), pp. 583–609, doi:10.1017/S0017089506003211. arXiv:math/0509363.
- [9] R.M. Green (2009): *On the Markov trace for Temperley–Lieb algebras of type E_n* . *J. Knot Theory Ramifications* 18(2), pp. 237–264, doi:10.1142/S0218216509006872. arXiv:0704.0283.
- [10] V.F.R. Jones (1987): *Hecke algebra representations of braid groups and link polynomials*. *Ann. of Math.* 126(2), pp. 335–388, doi:10.2307/1971403.
- [11] L.H. Kauffman (1987): *State models and the Jones polynomial*. *Topology* 26(3), pp. 395–407, doi:10.1016/0040-9383(87)90009-7.
- [12] R. Penrose (1971): *Angular momentum: an approach to combinatorial spacetime*. *Quantum theory and beyond* 151, pp. 395–407.
- [13] H.N.V. Temperley & E.H. Lieb (1971): *Relations between the “percolation” and “colouring” problem and other graph-theoretical problems associated with regular planar lattices: some exact results for the “percolation” problem*. *Proc. Roy. Soc. London Ser. A* 322(1549), pp. 251–280, doi:10.1098/rspa.1971.0067.

Optimal Generation of Strictly Increasing Binary Trees and Beyond

Olivier Bodini 

LIPN
Villetaneuse, France
Institut Galilé
Université Sorbonne Paris-Nord
olivier.bodini@lipn.univ-paris13.fr

Francis Durand 

LIPN
Villetaneuse, France
Institut Galilé
Université Sorbonne Paris-Nord
francis.durand@ens-paris-saclay.fr

Philippe Marchal 

LAGA
Villetaneuse, France
Institut Galilé
Université Sorbonne Paris-Nord
marchal@math.univ-paris13.fr

1 Introduction

Tree-like data structures are fundamental in computer science, serving as critical tools for modeling phenomena, testing, and creating diverse and representative datasets that enable effective training of machine learning frameworks by exposing models to a wide range of possible input scenarios. Recognizing the necessity for random samplers of these structures, a comprehensive body of work has emerged offering algorithms for generating simple tree families. These algorithms, ranging from general approaches like the Recursive method and Boltzmann sampler [2] to specific ones like the BBJ algorithm for binary trees [1], excel in efficiency and minimal randomness usage. Despite the abundance of techniques for simple trees, the random generation of increasing trees—a vital component in priority queue management—remains underexplored. This gap is attributed to their non-uniform internal structure, challenging the creation of homogeneous algorithms. Our paper introduces a groundbreaking algorithm for the optimal generation of strictly increasing binary trees, leveraging a novel approach pioneered by Ph. Marchal [3] that ensures both entropy and time efficiency and we prove here random bit complexity. Additionally, we present an enhanced algorithm that adopts an innovative approximation schema for the recursive method. This method is tailored for all weighted unary-binary increasing trees, guaranteeing minimal randomness consumption. This dual approach not only advances the field of random increasing tree generation but also sets a new standard for algorithmic efficiency and randomness optimization. The next section is devoted to presenting the first algorithm and evaluating its cost. Section 3 addresses the second algorithm, which represents an improvement of the recursive method through the use of a Monte Carlo process. The final section demonstrates how the second method can be adapted to operate within a more general framework.

2 An Ad Hoc Approach for Sampling Strictly Increasing Trees

Before introducing the first algorithm, let us make a first easy but important observation. Strictly increasing binary trees are intimately related to permutations (this has been known for a long time). Given a

sequence of distinct real numbers in the interval $[0,1]$, we can derive a permutation based on the relative order of the sequence, and thus an increasing tree in the following way. Starting with the position of the smallest number in the sequence, we place a 1 at this position in the permutation (and at the same time, we create a root for the tree), and continue this process until all the numbers have been placed (in the permutation and in the binary tree). If the sequence alternates between increasing and decreasing values, the resulting permutation will also alternate and the tree will be a strictly increasing binary tree. Furthermore, suppose the sequence is chosen randomly and uniformly. In that case, the permutation reflects a uniform distribution of all possible permutations and the binary tree constructed is uniform among binary trees of size n . This leads us to look for an algorithm that generates a random, uniform, and alternating sequence, which in turn allows us to construct a uniform alternating permutation and hence a uniform strictly increasing binary tree.

Algorithm 1 Generate an Alternating Permutation of Size n

Require: $n > 1$

```

1:  $D, Y \leftarrow$  arrays of size  $n$ 
2:  $p \leftarrow \lfloor \log_2(n) \rfloor + 1$ 
3: while precision on  $Y$  is not sufficient do
4:    $p \leftarrow p + 1$ 
5:   while  $Y$  is empty do
6:      $r \leftarrow U_0$  with  $p$  digits.
7:      $D[0] \leftarrow 1 - \sin^2\left(\frac{\pi}{2}r\right)$ 
8:     for  $i \leftarrow 0$  to  $n - 2$  do
9:        $r \leftarrow U_{i+1}$  with  $p$  digits.
10:       $D[i + 1] = r^2(1 - D[i])$ 
11:       $\alpha \leftarrow \sqrt{\frac{1 - D[n-1]}{1 - D[0]}}$ 
12:       $threshold \leftarrow \frac{1}{\alpha + \alpha^{-1}}$ 
13:       $proba \leftarrow \text{rand}()$ 
14:      if  $proba \leq threshold$  then
15:         $Y \leftarrow (1 - D[0], D[1], 1 - D[2], \dots)$ 
16:      else if  $threshold < proba \leq 2 \cdot threshold$  then
17:         $Y \leftarrow (1 - D[n - 1], D[n - 2], 1 - D[n - 3], \dots)$ 
18:      else
19:        restart
20:   Sort  $Y$  to determine if its elements are strictly ordered
21: return the alternating sequence associated with  $Y$ 

```

This algorithm is essentially a rewritten version of Marchal's algorithm, modified to bypass inefficient trigonometric calculations. More precisely, by making the following change of variable $C_n = \sin^2\left(\frac{\pi}{2}X_n\right)$ where X_n is the sequence described in [3], we do not alter the properties of the sequence and we significantly improve the computation. We refer the reader to the initial note by Marchal [3] for the proof of his algorithm's validity. A key point is that the algorithm uses a rejection procedure but it is proved in [3] that the rejection probability is bounded above by $1 - \frac{2}{3\pi}$ (which is independent of n).

The remaining issue concerns the random-bit complexity. Naturally, for each uniform i.i.d. random variable U_i , we aim to generate only enough digits to ensure sufficient precision, thus avoiding any order

ambiguity in the final sequence.

As a preliminary, the number of strictly increasing binary trees of size n is asymptotically equivalent to $2 \left(\frac{2}{\pi}\right)^{n+1} n!$. Thus, the entropy is $n \log_2(n) + \frac{2n}{\pi} + o(n)$. As we draw n uniform variables, we need at least $\log_2(n)$ bits of precision to be coherent with the entropy.

Now, note that $D[n]$ converges when n goes to the infinity to the distribution

$$Q := \sum_{n=0}^{\infty} (-1)^n \prod_{i=0}^n U_i^2$$

Indeed, we can derive from the algorithm the following induction: $Q = U^2(1 - V^2Q)$ where U and V are i.i.d. uniform distributions on $[0, 1]$. Consequently, the density function $d(z)$ of Q follows the functional equation:

$$d(z) = \int_z^1 \frac{1}{2\sqrt{t}} \cdot \frac{d(1-z/t)}{t} dt.$$

To our knowledge, solving this equation is not straightforward. For that, let us start from the density of the random variable $Q_0 = U^2(1 - V^2)$, which is:

$$d(z) := \frac{\arctan\left(\frac{\sqrt{1-z}}{\sqrt{z}}\right)}{2\sqrt{z}},$$

from which we can observe that the iteration $Q_{n+1} = U^2(1 - V^2Q_n)$ generates random variables whose density is alternatively a polynomial in $A := \arctan\left(\frac{\sqrt{1-z}}{\sqrt{z}}\right)$ and in $\bar{A} := \arctan\left(\frac{\sqrt{z}}{\sqrt{1-z}}\right)$, divided by \sqrt{z} . Moreover, the polynomials follow a quite simple recurrence which alternates

$$P_{n+1}(\bar{A}) = \int_{\bar{A}}^{\frac{\pi}{2}} P_n(A) dA$$

and

$$P_{n+1}(A) = \int_0^A P_n(\bar{A}) d\bar{A}.$$

So, the limiting polynomial follows the functional equation

$$\int_0^x \left(\int_x^{\frac{\pi}{2}} P(t) dt \right) dx = P(x),$$

from which we derive that the limiting distribution admits two solutions depending on the parity on n : $\frac{2\sqrt{1-z}}{\pi\sqrt{z}}$ and $\frac{2\sqrt{z}}{\pi\sqrt{1-z}}$. Therefore, we can conclude that the density of the limiting random variable is

$$ds(z) = \frac{\left(\frac{\sqrt{1-z}}{\sqrt{z}}\right) + \left(\frac{\sqrt{z}}{\sqrt{1-z}}\right)}{\pi}.$$

In order to evaluate the required precision, let us assume that all the $D[i]$ are independent and identically distributed and follow the symmetrized distribution Q of density $ds(z)$. This statement is not strictly true; however, it is asymptotically acceptable. This acceptability arises because the dependency between $D[i]$ and $D[j]$ decreases exponentially fast as the distance $|j - i|$ increases. This property follows directly from

$$Q := \sum_{n=0}^{\infty} (-1)^n \prod_{i=0}^n U_i^2$$

and is really important because it implies that the errors do not accumulate.

Let δ denote the number of digits of precision (in base 2). Then, the probability $P_{n,\delta}$ that there is no ambiguity in a calculation of the alternating sequence of size n with δ digits of precision is given by:

$$P_{n,\delta} := n! [z^n] \prod_{i=1}^{2^\delta-1} \left(1 + z \frac{ds\left(\frac{i}{2^\delta}\right)}{\sum_{i=1}^{2^\delta-1} ds\left(\frac{i}{2^\delta}\right)} \right),$$

Now, putting $N := 2^\delta$ and $p_{i,n} := \frac{ds\left(\frac{i}{N}\right)}{\sum_{i=1}^{N-1} ds\left(\frac{i}{N}\right)}$, when n is large, $\sum_{i=1}^{N-1} ds\left(\frac{i}{N}\right) / N$ tends to 1, and $\sum_{i=1}^{N-1} p_{i,n}^2$ is asymptotically equivalent to $\frac{1}{n} \int_{1/n}^{1-1/n} ds(x) dx$, which simplifies to $\sum_{i=1}^{N-1} p_{i,n}^2 \sim \frac{2 \ln(n)}{\pi^2 n}$. Consequently, $P_{n,\delta}$ is asymptotically equivalent to $1 - \frac{N(N-1)}{2} \frac{2 \ln(n)}{\pi^2 n}$. Indeed, the extraction $[z^n]$ is nothing more than the $PSet_n$ of the $B(p) = \sum p_{i,n}$ where the $p_{i,n}$ are seen as $n-1$ different atoms and it is well known that $PSet_n$ can be expressed as a multivariate polynomial in the $B(p^i)$. Moreover the two dominant contributions are $\frac{B(p)^n}{n!} - \frac{B(p^2)B(p)^{n-2}}{2(n-2)!}$. Finally, to achieve a probability of rejection ε , we require $\delta = 2 \log_2(n) + \log_2(\ln(n)) + \kappa + \log_2 \varepsilon$. Thus, the required precision is proportional (factor 2) to the entropy. It is worth noting that improving the precision by one unit halves the risk of ambiguity.

This algorithm is highly efficient, but it cannot be extended to other types of increasing trees. The following section introduces a more flexible approach.

3 Toward an Efficient Recursive Methods Without Preliminary Calculations

We aim at generating a random strictly increasing binary tree with an exact size of n . The generating function for our combinatorial family satisfies the non-linear differential equation $T'(z) = T^2(z) + 1$ and let us denote by t_n the number of trees of size n . Utilizing the symbolic method for combinatorial structures, we derive a recursive method algorithm for our purpose. Our approach is now to avoid the exact and costly computation of t_n by a well-controlled approximation process, which paves the way for a systematic and efficient algorithm to generate these structures, as shown in Algorithm 2.

Algorithm 2 Generating T of Size n with the recursive method

- 1: **If** $n = 1$: Return the tree with one leaf labeled 1
 - 2: *Intelligently* generate $M \in \{0, 1, \dots, n-1\}$ the size of the left son of T . Note that $P(M = m)$ has to be proportional to $\binom{n-1}{m} t_m t_{n-1-m}$.
 - 3: Generate recursively T' and T'' of size m and $n-1-m$ and return T with T' as a left son and T'' as a right son with the indexes "shuffled" among the $n-1$ remaining atoms.
-

There are a few problems with this generation scheme: How do we compute the t_n 's? The classical way to do so is to pre-compute all of them but doing so requires at least $\Omega(n^2)$ in space and even more in time. Generating M can also be an issue, the classical way to generate M would require computing all the t_n 's and inverting the probability function which can be ineffective.

We describe a way to compute only the necessary bits of precision of t_n and a way to generate M without the exact knowledge of t_n , solving the two issues.

4 Generating M , the size of the left son

4.1 Having a direct expression for t_n

The differential equation $T'(z) = T^2(z) + 1$ can be integrated in $T(z) = \tan(z)$ because $T(0) = 0$ (there is no strictly increasing binary tree of size 0).

We can hence use the tangent development: $\tan(z) = \sum_{k \geq 1} \frac{4^k(4^k-1)|B_{2k}|}{(2k)!} z^{2k-1}$ with $|B_{2k}| = \frac{2(2k)!\zeta(2k)}{(2\pi)^{2k}}$ being the absolute value of the Bernoulli numbers, and ζ being the Riemann's zeta function. So we can extract the number of strictly increasing binary trees of size n . If n is even, $t_n = 0$ because \tan is odd, and for $n = 2k - 1$, $t_{2k-1} = (2k - 1)! \frac{2(4^k-1)\zeta(2k)}{\pi^{2k}}$.

In the following, we assume that $n = 2l - 1$ is odd since there is nothing to generate if n is even.

Since $P(M = m)$ is proportional to $\binom{n-1}{m} t_m t_{n-m-1}$, so $P(M = m) = 0$ when m is even. So we can write, for $m = 2k - 1$, that $P(M = m)$ is proportional to $\binom{2l}{2k-1} \frac{(2k-1)! 2(4^k-1) 4^k \zeta(2k)}{(2\pi)^{2k}} \frac{(2(l-k)-1)! (4^{l-k}-1) 4^{l-k} \zeta(2(l-k))}{(2\pi)^{2(l-k)}}$. We can simplify this expression and get rid of the constant terms, we get that $P(M = 2k - 1)$ is proportional to $f_k := (4^k - 1)\zeta(2k)(4^{l-k} - 1)\zeta(2(l - k))$, with $k \in \{1, 2, \dots, l - 1\}$.

Since $\zeta(2k) = \sum_{i \geq 1} \frac{1}{i^{2k}}$, by truncating the sum, $\zeta(2k) \geq 1 + \frac{1}{4^k}$. This leads to the following bound on f_k : $f_k \geq (4^k - 1)(1 + \frac{1}{4^k})(4^{n-k} - 1)(1 + \frac{1}{4^{n-k}})$, i.e. $f_k \geq (4^k - \frac{1}{4^k})(4^{n-k} - \frac{1}{4^{n-k}})$. With this lower bound and the fact that (f_k) are symmetric decreasing from 1 to $\lfloor \frac{l}{2} \rfloor$, we can bound f_k .

Lemma 1. For $k \in \{1, 2, \dots, l - 1\}$, $f_1 \geq f_k \geq f_{\lfloor \frac{l}{2} \rfloor}$ and $\frac{f_1}{f_{\lfloor \frac{l}{2} \rfloor}} \leq \frac{4\pi^4}{45} \leq 9$

This lemma implies a Monte-Carlo method for sampling M .

Algorithm 3 Generate M such that $P(M = 2k - 1)$ is proportional to f_k

- 1: Generate $X \in \{1, 2, \dots, l - 1\}$.
 - 2: Test $U f_1 \leq f_X$ where U is a uniform random number in $[0, 1]$, if true return $M = X$, else generate another X and start again.
-

Lemma 1 implies that Algorithm 3 has a constant reject.

Lemma 2. Algorithm 3 has a running time of $O(\log(n))$

This lemma is not only due to the fact that the reject is constant but also to the fact that it is possible to compute only the first bits of f_1 , f_k , and U and check whether we can conclude that $U f_1 \leq f_X$ or $U f_1 > f_X$. If we cannot conclude, we can add the next bits and recheck until we can accept or reject X as M .

It is possible to compute only the first bits of $f_k = (4^k - 1)\zeta(2k)(4^{l-k} - 1)\zeta(2(l - k))$ because we can truncate $\zeta(2k) = \sum_{i \geq 1} \frac{1}{i^{2k}}$ to a finite order.

With this method of generating M , we can state the complexity of Algorithm 2.

Theorem 1. Algorithm 2 has a running time of $O(n \log(n))$, which is optimal.

The algorithm is optimal because of the Shannon entropic principle: It is not possible to generate a strictly increasing binary tree of size n (n odd) with less than $\log_2(t_n)$ random bits, so the best sampling algorithm will run in at least $\log_2(t_n) = \Omega(n \log(n))$

5 Adapting the Algorithm for Different Families of Binary-Increasing Trees

5.1 Binary Trees

Consider various families of increasing binary trees, each characterized by distinct generating functions; for instance, non-plane binary trees have $T'(z) = \frac{T^2(z)}{2} + 1$, leading to $T(z) = \frac{\tan(z/\sqrt{2})}{\sqrt{2}}$. For all families of binary-increasing trees with generating functions of the form $T' = aT^2 + c$ as described in Algorithm 2, the algorithm remains applicable. This is due to the consistent structure shared by these trees, characterized by having one more leaf than internal nodes.

5.2 Binary-Unary Trees

For a family with generating function $T'(z) = a(T(z) + \alpha)^2 + a\gamma^2$, where $a, c > 0$ and $b \geq 0$, we focus on those written as $T'(z) = a(T(z) + \alpha)^2 + a\gamma^2$, the case $T'(z) = a(T(z) + \alpha)^2 - a\gamma^2$ being similar. The algorithm generates trees of size n with $t_n \sim \frac{n!a^n\gamma^{n+1}}{(\frac{\pi}{2} - \tan^{-1}(\frac{\alpha}{\gamma}))^{n+1}}$, allowing effective recursive generation. And

$$t_n = n!a^n\gamma^{n+1} \left(\sum_{k \geq 0} \frac{(-1)^{n+1}}{(\tan^{-1}(\frac{\alpha}{\gamma}) - \pi(k + \frac{1}{2}))^{n+1}} + \frac{(-1)^{n+1}}{(\tan^{-1}(\frac{\alpha}{\gamma}) + \pi(k + \frac{1}{2}))^{n+1}} \right)$$

is still computable by truncating its expression.

Algorithm 4 Generating T of Size n for Binary-Unary Trees

- 1: **If** $n = 1$: Return the tree with one leaf colored in one of the c colors.
 - 2: **For** $n > 1$: The tree can have a unary root with probability $\frac{bt_{n-1}}{t_n}$ or a binary root.
 - 3: Let X be a Bernoulli variable with parameter $\frac{bt_{n-1}}{t_n}$; if $X = 1$, call the algorithm recursively with parameter $n - 1$ and return the tree consisting of a unary root and labeled 1 with T' as its only son.
 - 4: Generate $M \in \{0, 1, \dots, n - 1\}$ with $P(M = m)$ proportional to $\binom{n-1}{m} t_m t_{n-1-m}$.
 - 5: Generate recursively T' and T'' of size m and $n - 1 - m$ and return T with T' as a left son and T'' as a right son with the indexes "shuffled" among the $n - 1$ remaining axioms.
-

References

- [1] Axel Bacher, Olivier Bodini & Alice Jacquot (2017): *Efficient random sampling of binary and unary-binary trees via holonomic equations*. *Theor. Comput. Sci.* 695, pp. 42–53, doi:10.1016/J.TCS.2017.07.009.
- [2] Philippe Duchon, Philippe Flajolet, Guy Louchard & Gilles Schaeffer (2004): *Boltzmann Samplers for the Random Generation of Combinatorial Structures*. *Combinatorics, Probability and Computing* 13(4–5), p. 577–625, doi:10.1017/S0963548304006315.
- [3] Ph. Marchal (2012): *Generating random alternating permutations in time $n \log n$* . Available at <https://www.math.univ-paris13.fr/~marchal/altperm1.pdf>. Unpublished note.

Uniform Sampling and Visualization of 3D Reluctant Walks

Benjamin Buckley

Marni Mishna*

Simon Fraser University
Burnaby, BC, Canada

bbuckley@sfu.ca

mmishna@sfu.ca

A family of walks confined to the first orthant whose defining stepset has drift outside of the region can be challenging to sample uniformly at random for large lengths. We address this by generalizing the 2D walk sampler of Lumbroso et al. to handle 3D walks restricted to the first orthant. The sampler includes a visualizer and means to animate the walks.

1 Introduction

A combinatorial class of lattice walks is a set of walks, where each step is taken from a finite set of possibilities, and that the walks remain within some region (typically a cone centered at the origin). The simplicity of lattice walks contributes to their ubiquity. Indeed, many families of discrete objects have natural bijections to lattice walks models. Visualizations of the large scale behaviour of uniformly generated random walks can reveal underlying structure of the walks, and consequently, of related objects.

Here, we will focus on the uniform random generation of 3-D walks confined to the first orthant ($\mathbb{Z}_{\geq 0}^3$), in particular models that are said to be *reluctant* because the drift of the stepset (the vector sum) is outside of the cone. The random sampling algorithm naturally extends the 2D sampling algorithm for reluctant walks of Lumbroso et al. [13]. Fig. 1 illustrates a visualization of a 16,778 step reluctant walk restricted to the first orthant, (with drift $(-1, -1, -1)$) taken from the Unity interface we have developed. We have represented the walk steps using blocks, and have coloured the walk in a way to make its evolution clear.

1.1 Motivation

When the stepset has zero drift, naive generation (where steps are drawn at random, and a walk is rejected once it leaves the region) can be effective. However, when the drift is outside of the region, particularly when it fits the criterion (defined below) of being reluctant such a strategy is defeated by the very small proportion of unrestricted walks that remain in the region.

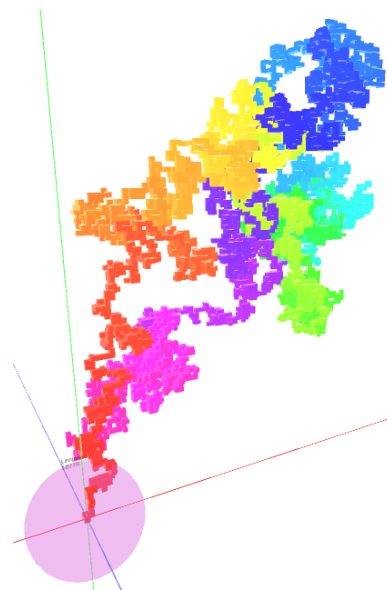


Figure 1: A walk of length 16,778 generated uniformly at random from a stepset with drift $(-1, -1, -1)$. Progression of the walk is given by hue, with the initial position at $(0,0,0)$ in red and the final position in magenta.

*This work is generously funded by NSERC Discovery Grant (Canada) RGPIN-04157

Some recent asymptotic studies of weighted walk models have provided explicit examples of complete asymptotic formulas in terms of the weights and have pointed to some potential sources of interesting phenomena. For example in both [6], and [14] the authors fix a stepset, and then determine asymptotic counting formulas as a function of weightings of each allowable step. In both cases, the exponential growth factor in the dominant term is smooth as a function of the weights, but the sub-exponential growth is defined piecewise in terms of the drift of the model. That is to say, as the drift changes the asymptotic growth formula undergoes phase transitions as the drift crosses into different regions. One of our motivations is to understand how these changes influence a uniformly chosen model near these critical drift regimes. A robust, easy to use visualizer can be useful to identify phenomena that may subsequently be established analytically or algebraically.

1.2 Definitions

To describe the algorithm, we first make some notation explicit. We focus on finite stepsets $\mathcal{S} \subset \mathbb{Z}^3$ with the property that the defining set of vectors are not contained in any half-space. A walk starts at the origin, takes steps in \mathcal{S} such that there is no step outside of the first orthant, \mathbb{N}^3 . Although our random sampler is generic, in this abstract we present various integer weightings of $\{\pm e_1, \pm e_2, \pm e_3\}$. In this context a model is reluctant if the drift has all negative components, that is $\sum_{s \in \mathcal{S}} s \in \mathbb{Z}_{<0}^3$. We also consider weightings that result in a drift outside the first orthant.

2 Algorithm

The sampling algorithm we use, like the 2D sampling algorithm of Lumbroso *et al.* [13] has three key elements. It is a rejection algorithm, but rather than generating an unrestricted walk, our algorithm generates walks confined to a judiciously chosen half-space containing the first orthant, and rejects generated walks if they exit the first orthant. The set of walks confined to a half space is in bijection with a set of generalized Dyck words, for which a grammar can be made explicit [8]. Given the grammar, uniform sampling can be effectively accomplished using a Boltzmann sampler. We visualize the walks in Unity to give access to an interface, and open the possibility to animation. Images and videos demonstrating this are available to view at <https://benbuckleyanimator.wixsite.com/portfolio/randwalkviz>.

2.1 A well chosen half-space

From our restriction on \mathcal{S} we can deduce that there exists a half space (and possibly many) so that, asymptotically, the number of walks confined to that half space has the same exponential growth factor as walks further confined to the first orthant. We say more about this in the example below, but generically this follows from a result of Garbit and Raschel [11]. This means that the proportion of the walks the half-space that are *not* in the first orthant grows sub-exponentially with length.

Since walks confined to a half space only interact with a single boundary, these walks are in bijection with a one dimensional model. To define this one dimensional model, the stepset \mathcal{S} is projected onto a vector $\mathbf{v} = (a, b, 1) \in \mathbb{R}^3$ orthogonal to the hyperplane defining the halfspace to create a 1D stepset. We can make this set explicit: $\mathcal{A}_{\mathbf{v}} = \{ai + bj + k \mid (i, j, k) \in \mathcal{S}\}$, with steps in \mathbb{R} . The exponential growth of the asymptotic number of walks in the first orthant model is the equal to the minimum value of the Laurent polynomial $S(x, y, z) = \sum_{(i, j, k) \in \mathcal{S}} x^i y^j z^k$ attained in a suitable domain of $\mathbb{R}_{>0}^3$. Suppose the minimum value is attained at (x^*, y^*, z^*) . Provided $z^* \neq 1$, we can show that the 1D model associated to $\mathbf{v} = (a, b, 1)$ with $a = \log(x^*)/\log(z^*)$, $b = \log(y^*)/\log(z^*)$ is such that the exponential growth of the

asymptotic number of walks in the model is the same as the orthant model. Remark, that because of our conditions on \mathcal{S} we have $x^*y^*z^* \neq 0$. In the case where $z^* = 1$, we can project on to $(a, b, 0)$ with $a = \log(x^*)$ and $b = \log(y^*)$. If $x^* = y^* = z^* = 1$, then any halfspace containing the orthant can be used.

2.2 Generalized Dyck words

The previous process yields a one-dimensional stepset with real valued steps. In order to create a grammar for a Boltzmann sampler, we need a stepset with integer valued steps. We find a rational approximation – that is, a stepset with steps in \mathbb{Q} sharing a small denominator, typically < 10 for ease of computation – and then multiply through by the denominators to obtain a stepset with steps in \mathbb{Z} . The closer the approximation, the less the rejection although we will always generate walks in the orthant uniformly. In our example, the previous step yields integer steps directly.

Given a finite set of integers, the grammar for 1D walks that do not go below 0 is described in [8]. We have applied to this to our running example in Fig. 2 below.

2.3 Boltzmann generation

Given the grammar, the sampling is done using a Boltzmann sampling strategy, following [9], although any random generating strategy for a grammar could be used in its place. Here we describe only how we compute some of the required elements.

Given the grammar, we used Maple’s Comstruct library to solve the algebraic equations to obtain explicit generating function for these 1-dimensional walks. It turns out that this is the most difficult step for computational reasons: when stepset has large integers, the number of rules in the grammar can grow quickly, potentially inhibiting the computation of the generating function.

When an explicit generating function is not determined a high order series approximation is used in the computation.

Given the generating function, we determine its dominant singularity, and in the case of a series approximation we determine an approximation. Boltzmann sampling uses generating function evaluations at, or near, this singularity.

Boltzmann sampling can generate objects of any size, but, for a given size the sampling is uniform. To generate large objects, one uses evaluation points close to or at the dominant singularity, provided it is not a pole. In these problems, the singularities are branch points, not poles, and so we can evaluate directly at the dominant singularities.

2.4 Running the algorithm

Thus, in summary, for a given model \mathcal{S} , we first have a set up phase to first determine the best half-space, then generate the corresponding grammar, and do the computational preamble (generating function evaluations) for the Boltzmann sampler. To generate a walk, we run the sampler, biject back to a 3D walk, and eliminate those walks that leave the first orthant. The preamble is done in Maple, but we run the sampler in Unity.

$$\begin{aligned}
 \mathcal{D} &= \mathcal{D} \times \mathcal{P}_{aux} \\
 \mathcal{P}_{aux} &= \varepsilon + \mathcal{L}_1 \times \mathcal{P}_{aux} \\
 \mathcal{L}_1 &= a_1 \times \mathcal{D} + a_2 \times \mathcal{D} + a_3 \times \mathcal{D} \\
 \mathcal{R}_1 &= b_1 \times \mathcal{D} + b_2 \times \mathcal{D} + b_3 \times \mathcal{D} + b_4 \times \mathcal{D} + b_5 \times \mathcal{D} + b_6 \times \mathcal{D} \\
 \mathcal{D} &= \mathcal{L}_1 \times \mathcal{R}_1 + \varepsilon \\
 a_1, \dots, a_3 &= \text{Atoms representing } +1 \text{ steps} \\
 b_1, \dots, b_6 &= \text{Atoms representing } -1 \text{ steps}
 \end{aligned}$$

Figure 2: Grammar for 1-dimensional walks with inventory $A(u) = 3u + 6/u$

3 Drawn walks

3.1 The reluctant model in Fig.1

The walk in Fig. 1 is sampled from the set of walks with the stepset $\{e_1, e_2, e_3, -e_1, -e_1, -e_2, -e_2, -e_3, -e_3\}$. Note that this is a multiset with each of the negative steps appearing twice, increasing their probability. We calculate the drift by taking the vector sum of these steps, and obtaining $-e_1 - e_2 - e_3 = (-1, -1, -1)$. We note that this model fits in the framework of Theorem 2 in [14], from which we deduce that the number of walks of length n grows, up to a constant, like $(6\sqrt{2})^n n^{-3}$. Thus, the exponential growth factor is $6\sqrt{2}$.

Find the halfspace The inventory of the stepset is $S(x, y, z) = \sum_{(i,j,k) \in \mathcal{S}} x^i y^j z^k = x + 2/x + y + 2/y + z + 2/z$. A gradient computation shows this function is minimized at $\sqrt{2}(1, 1, 1)$. The optimal vector is $\mathbf{v} = (a, b, 1)$ with $a = \frac{\log(x^*)}{\log(z^*)}$ and $b = \frac{\log(y^*)}{\log(z^*)}$, that is, $\mathbf{v} = (1, 1, 1)$. The stepset \mathcal{S} is projected onto this vector to obtain a 1-dimensional stepset with steps in \mathbb{R} :

$$\mathcal{A}_{\mathbf{v}} = \{1, 1, 1, -1, -1, -1, -1, -1, -1\}.$$

Remark that the inventory of this walk is $A(u) = 3u + 6/u$. Because the drift is negative, by Banderier and Flajolet [1, Theorem 4], the exponential growth factor for the walks that are never negative is $A(\tau)$, where τ is the critical point of $A(u)$. In this case $A'(\sqrt{2}) = 0$ and $A(\tau) = 2\sqrt{6}$. Furthermore, we know that the number of walks of length n , grows like $(2\sqrt{6})^n n^{-3/2}$ (up to a multiplicative constant) We are assured: This walk has an identical exponential growth factor to the 3D walks. The proportion of these walks that are images of walks remaining in the first orthant is approximately $\frac{n^{-3}}{n^{-3/2}} = n^{-3/2}$, for length n .

Find the grammar As the steps are all integer valued, we can proceed directly to the grammar phase. We use the formulation of grammars for generalized Dyck words, as described in [8], to obtain a grammar for half-hyperplane walks in our 1-dimensional stepset. Following this specification, we obtain the grammar in Figure 2.

Boltzmann sampling To generate a walk, we next set up the Boltzmann sampler. This requires generating functions for each non-terminal in the grammar. The generator requires an evaluation point, which we take as the dominant singularity $\frac{1}{6\sqrt{2}}$. Walks that are deemed too short or outside the target length are

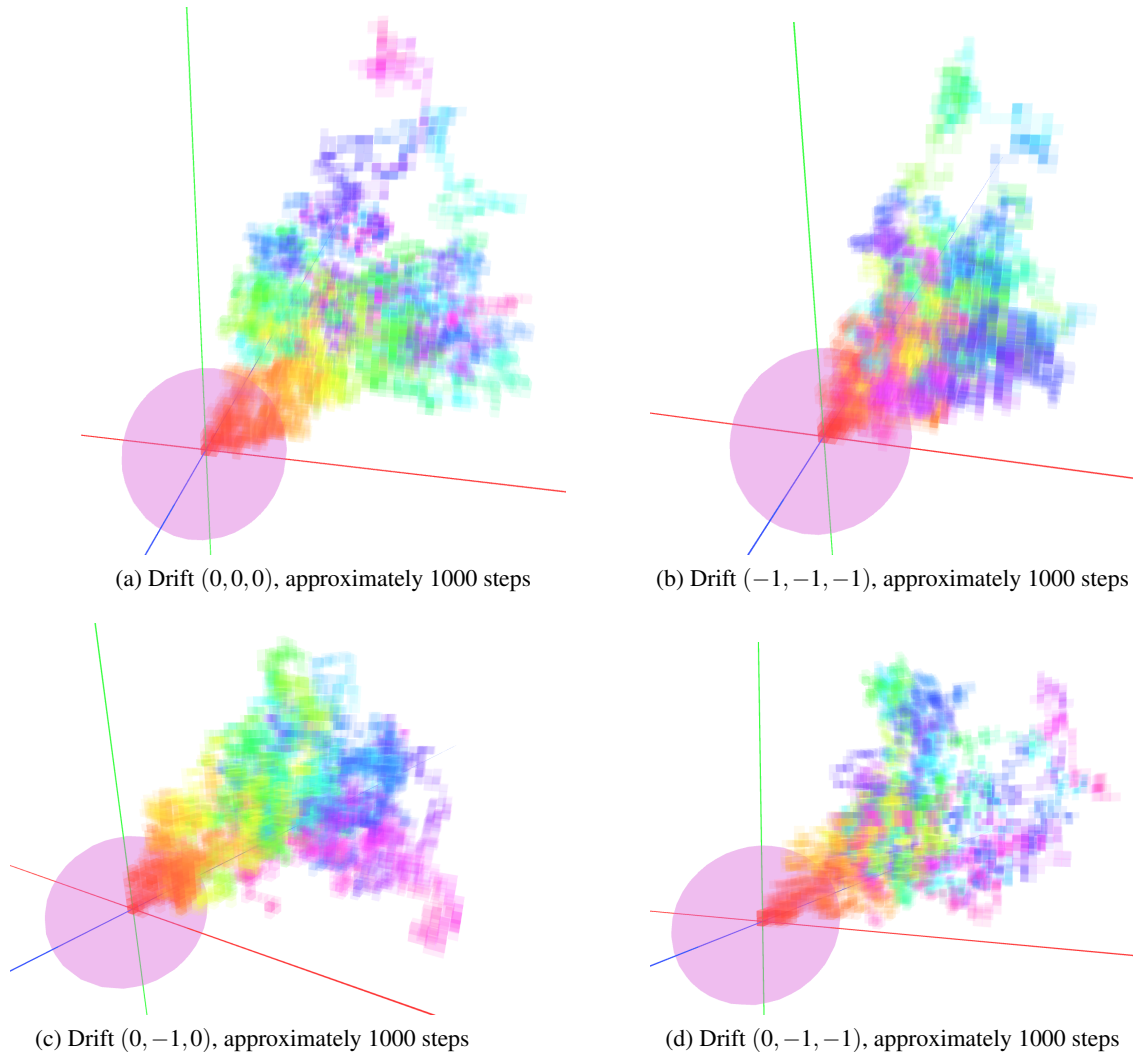


Figure 3: Randomly generated lattice walks with various stepsets. Images each show 10 walks in the same space. A translucent sphere with radius 20 is centered at the origin to give a sense of scale.

rejected. The inverse bijection is applied to determine the corresponding 3-dimensional step from which it was projected. The walk is accepted if it stays in the orthant, and rejected if it exits.

Verification Since there is some computational approximation in the Boltzmann sampler, we did some tests to convince ourselves that we were achieving uniform generation. We generated 10,000 walks of length 10, and tallied the number of walks ending at each point. The proportions were compared with the exact proportion of walks expected to end at each point, calculated using recurrences. As the number of walks increases, we calculate the root-mean-square error between the discrete distribution generated by our results, and the distribution implied by exact counting. The root-mean-square error decreases from ≈ 0.00340361673 when generating 10 walks, $\approx 2.815461678 \times 10^{-6}$ when generating 1,000,000 walks. While this isn't a proof, it does lend confidence that the algorithm generates walks uniformly.

3.2 Comparative study of models

Some examples of walks in \mathbb{Z}^3 , restricted to the positive octant, can be seen in Fig. 3. Multiple walks are shown in each image.

The development of a typical walk changes significantly depending on the drift.

For example, note that in Fig. 3(b), when the drift tends towards the XZ plane, the walks tend to form an arc from the origin towards a point further out on the XZ plane. In contrast, in Fig. 3(d), then the drift tends towards the origin, a cluster of magenta near the origin and green on the outer regions suggests that walks tend to venture far away from the origin, not returning until the end of the walk.

Our algorithm makes it possible to generate long reluctant walks with high probability in cases where a generating a walk of comparable length under the naive scheme is virtually statistically impossible. Using the approximate number of orthant walks of length n to be $(6\sqrt{2})^n n^{-3}$, we can see that in the naive scheme a walk of length n has probability on the order of $\left(\frac{6\sqrt{2}}{9}\right)^n n^{-3}$ of being generated in a given run. This is less than .0002% when $n = 100$, where as the Boltzmann sampler has probability $n^{-3/2}$, which is 0.1% for $n = 100$. Both take strategies take linear time to generate a walk of length n .

To be even more concrete, we generated 10 walks with drift $(-1, -1, -1)$ of length around 100 using both strategies. With the naive rejection 133,065,405 walks were generated in total in order to find 10 that remained in the positive octant, the rest being rejected. This took approximately 8 minutes and 8.7 seconds. In contrast, when using Boltzmann sampling targeting lengths between 95 and 105 steps, it was only necessary to generate 202,669 walks to find 10 that were the appropriate length and remained in the positive octant. This was done in approximately 8.4 seconds. Both computations were done on similar conditions on a 2020 Macbook Pro with an Apple M1 chip.

4 Future work

One of the appeals of this project – and one of the reasons we’ve created it in the game engine Unity – is that the application provides the user with multiple ways to visualize and understand walks. The reader is encouraged to visit the site for some animations associated with reluctant walks.

We have discussed the colour coding already. It is also possible to animate sets of walks, with each frame showing a simultaneous position of every walk in the set. Recently, we’ve made it possible to view the convex hull of the simultaneous positions of the walks at a given step, with the convex hull changing as the walk progresses. (See Figure 4).

Other future plans include tools for generating data to give a sense of how quickly the walks are generated with different methods, and how many rejections occur. We also plan to create animations visualizing walks in terms of their bijection with other objects – for example, queues and tableaux. We also hope to make the Unity interface available in the future.

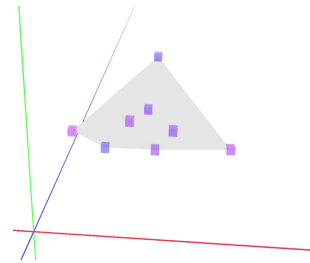


Figure 4: *The convex hull at step 220 in the progression of 10 walks.*

References

- [1] Cyril Banderier & Philippe Flajolet (2002): *Basic analytic combinatorics of directed lattice paths*. *Theoretical Computer Science* 281, pp. 37–80, doi:10.1016/S0304-3975(02)00007-5.
- [2] Alin Bostan, Mireille Bousquet-Mélou, Manuel Kauers & Stephen Melczer (2016): *On 3-dimensional lattice walks confined to the positive octant*. *Annals of Combinatorics* 20(4), pp. 661–704, doi:10.1007/s00026-016-0328-7.
- [3] Alin Bostan & Manuel Kauers (2009): *Automatic classification of restricted lattice walks*. In: *21st International Conference on Formal Power Series and Algebraic Combinatorics (FPSAC 2009)*, *Discrete Math. Theor. Comput. Sci. Proc. AK*, Assoc. Discrete Math. Theor. Comput. Sci., Nancy, pp. 201–215, doi:10.46298/dmtcs.2724.
- [4] Alin Bostan, Kilian Raschel & Bruno Salvy (2014): *Non-D-finite excursions in the quarter plane*. *Journal of Combinatorial Theory. Series A* 121, pp. 45–63, doi:10.1016/j.jcta.2013.09.005.
- [5] C. Banderier & Ph. Flajolet (2002): *Basic analytic combinatorics of directed lattice paths*. *Theoretical Computer Science* 281(1-2), pp. 37–80, doi:10.1016/S0304-3975(02)00007-5.
- [6] J. Courtiel, S. Melczer, M. Mishna & K. Raschel (2017): *Weighted lattice walks and universality classes*. *Journal of Combinatorial Theory. Series A* 152, pp. 255–302, doi:10.1016/j.jcta.2017.06.008.
- [7] Denis Denisov & Vitali Wachtel (2015): *Random walks in cones*. *The Annals of Probability* 43(3), pp. 992–1044, doi:10.1214/13-AOP867.
- [8] Philippe Duchon (2000): *On the enumeration and generation of generalized Dyck words*. In: *Discrete Mathematics*, 225, pp. 121–135, doi:10.1016/S0012-365X(00)00150-3.
- [9] Philippe Duchon, Philippe Flajolet, Guy Louchard & Gilles Schaeffer (2004): *Boltzmann samplers for the random generation of combinatorial structures*. *Combinatorics, Probability and Computing* 13(4-5), pp. 577–625, doi:10.1017/S0963548304006315.
- [10] Jetlir Duraj (2014): *Random walks in cones: the case of nonzero drift*. *Stochastic Processes and their Applications* 124(4), pp. 1503–1518, doi:10.1016/j.spa.2013.12.003.
- [11] Rodolphe Garbit & Kilian Raschel (2016): *On the exit time from a cone for random walks with drift*. *Revista Matemática Iberoamericana* 32(2), pp. 511–532, doi:10.4171/RMI/893.
- [12] Samuel Johnson, Marni Mishna & Karen Yeats (2018): *A combinatorial understanding of lattice path asymptotics*. *Advances in Applied Mathematics* 92, pp. 144–163, doi:10.1016/j.aam.2017.08.001.
- [13] Jérémie Lumbroso, Marni Mishna & Yann Ponty (2017): *Taming reluctant random walks in the positive quadrant*. In: *Random generation of combinatorial structures—GASCom 2016*, *Electron. Notes Discrete Math.* 59, Elsevier Sci. B. V., Amsterdam, pp. 99–114, doi:10.1016/j.endm.2017.05.008.
- [14] Marni Mishna & Samuel Simon (2020): *The asymptotics of reflectable weighted walks in arbitrary dimension*. *Advances in Applied Mathematics* 118, pp. 102043, 19, doi:10.1016/j.aam.2020.102043.

Pop Stacks with a Bypass

Lapo Cioni*

Dipartimento di Informatica
University of Pisa, Pisa, Italy
lapo.cioni@di.unipi.it

Luca Ferrari†

Dipartimento di Matematica e Informatica “U. Dini”
University of Firenze, Firenze, Italy
luca.ferrari@unifi.it

Rebecca Smith

Department of Mathematics
SUNY Brockport, Brockport, New York
rsmith@brockport.edu

We consider sorting procedures for permutations making use of pop stacks with a bypass operation, and explore the combinatorial properties of the associated algorithms.

1 Introduction

Sorting techniques for permutations constitute a flourishing research area in contemporary combinatorics. Starting from the stack-sorting procedure introduced by Knuth [7], over 150 articles (many of which are quite recent) have explored the subject. For example, machines making use of various types of containers have been studied, as well as networks of the corresponding devices. Typical containers that are considered in this context include stacks, queues, and dequeues, as well as their corresponding “pop” versions. In particular, a *pop stack* is a stack whose push and pop operations are similar to the usual ones for stacks, except that a pop operation extracts *all* the elements from the stack, rather than just the element on the top. The sorting power of a pop stack was originally studied by Avis and Newborn [3], who also considered pop stacks in parallel. Concerning pop stacks in series, we mention that the right-greedy version of pop stacks in series were introduced by Pudwell and Smith [8] and successively studied by Claesson and Guðmundsson [6].

In this work, we consider a new variant of a pop stack, in which we also allow a bypass operation. This gives the possibility of sending an element of the input permutation directly into the output, without necessarily pushing it into the pop stack first. This increases the sorting power of a pop stack, and inspires interesting combinatorial questions on its properties. We provide a detailed analysis of this sorting device, specifically our results are the following:

- We characterize the set of sortable permutations in terms of two forbidden patterns; the enumeration of the resulting class of pattern avoiding permutations was already known (odd-indexed Fibonacci numbers, sequence A001519 in [9]). However, we give an independent proof of this enumerative result by describing a bijective link with a restricted class of Motzkin paths;
- We describe an algorithm to compute the preimage of a given permutation, and use it to characterize and enumerate permutations having exactly 0, 1, and 2 preimages;
- We provide a complete description of the preimages of principal classes of permutations, determine in which cases the preimages are classes, and in such cases we compute the basis of the resulting classes;

*Partially supported by INdAM – GNCS group.

†Partially supported by INdAM – GNCS project CUP E53C23001670001.

- We characterize the set of sortable permutations for the compositions of our sorting algorithm with other classical sorting algorithms (the characterization being given in terms of forbidden patterns);
- We consider the device consisting of two pop stacks in parallel with a bypass, and we determine the basis of the associated class of sortable permutations.

2 Sorting using a pop stack with a bypass: characterization and enumeration of sortable permutations

A *pop stack* is a container in which elements can be stacked on top of each other, on which two operations are defined: the PUSH operation, which inserts an element in the stack, and the POP operation, which extract *all the elements* from the stack. The difference between a pop stack and a (classical) stack lies therefore in the way the elements are removed from the stack.

When using a stack or a pop stack to sort a permutation, the elements of the permutation are processed from left to right, and either the current element of the permutation is pushed into the stack at the top, or the topmost element (or all the content from top to bottom in the case of a pop stack) of the stack is popped into the output, where the topmost element is placed to the right of any other elements already in the output. We now introduce a new kind of pop stack, by allowing an additional BYPASS operation, which takes the current element of the permutation and places it directly into the output, in the next available position. More formally, given a permutation $\pi = \pi_1 \pi_2 \dots \pi_n$, a pop stack with bypass has the following allowed operations:

PUSH: Insert the current element of the input into the pop stack, on top of all the other elements (if there are any);

POP: Remove all the elements in the pop stack, from top to bottom, sending them into the output;

BYPASS: Output the current element of the input.

Our goal is to use a pop stack with a bypass to sort permutations. However, not every permutation can be sorted. Algorithm 1 (see below), called PSB (an acronym for “PopStacksort with Bypass”), is an optimal sorting algorithm, by which we can sort all sortable permutations.

Specifically, the algorithm PSB maintains elements in the pop stack only when they are consecutive in value (and increasing from top to bottom), which ensures that a POP operation is not directly responsible of the possible failing of the sorting procedure. Using the algorithm PSB, we can characterize the class of sortable permutations. Denote by **PSB** the map associated with the algorithm PSB. Moreover, given a set of permutations T , we indicate with $\text{Av}_n(T)$ the set of permutations of size n avoiding all patterns of the set T , and we write id_n for the identity permutation of size n . As usual, the set of all permutations of size n is denoted S_n .

Proposition 2.1. *Given $\pi \in S_n$, we have that $\text{PSB}(\pi) = id_n$ if and only if $\pi \in \text{Av}_n(231, 4213)$.*

The class of permutations sortable using a popstack with a bypass is a superclass of those sortable using a classical popstack, which is $\text{Av}_n(231, 312)$, as shown in [3]. The sequence $(|\text{Av}_n(231, 4213)|)_n$ is the sequence of odd-indexed Fibonacci numbers, i.e. sequence A001519 in [9], as shown in [1]. It is possible to give a bijective proof of this enumerative result by providing a link with a class of restricted Motzkin paths, whose enumeration can be easily carried out by standard techniques. The next proposition describes which kinds of restrictions on Motzkin paths need to be considered.

Proposition 2.2. *The set of permutations of size n sortable using a pop stack with a bypass is in bijection with the set of Motzkin paths whose total number of up and horizontal steps is n , which have no peaks, and are such that every maximal sequence of down steps reaches the bottom level.*

```

1  $S := \emptyset$ ;
2  $i := 1$ ;
3 while  $i \leq n$  do
4   if  $S = \emptyset$  or  $\pi_i = \text{TOP}(S) - 1$  then
5      $\text{PUSH}$ ;
6   else if  $\pi_i < \text{TOP}(S) - 1$  then
7      $\text{BYPASS}$ ;
8   else
9      $\text{POP}$ ;
10     $\text{PUSH}$ ;
11   $i := i + 1$ ;
12  $\text{POP}$ ;

```

Algorithm 1: PSB (S is the pop stack; $\text{TOP}(S)$ is the current top element of the pop stack; operations PUSH , POP , BYPASS are “Insert into the pop stack”, “Pour the whole content of the pop stack into the output”, “Bypass the pop stack”, respectively; $\pi = \pi_1 \cdots \pi_n$ is the input permutation.)

3 Preimages of a permutation

In this section, we investigate the set of permutations $\mathbf{PSB}^{-1}(\sigma)$ of all permutations whose output under PSB is σ . We remark that the investigation of preimages of permutations under various sorting operators was initiated for Stacksort in [5], and is now a fertile area of research.

We introduce a recursive algorithm that generates all preimages of a given permutation σ . Notice that our algorithm is actually defined for any sequence of distinct integers (not only permutations). This fact allows us to recursively execute it on subsequences of elements of a permutation. Recall that a *left-to-right maximum* of a permutation is an element which is larger than all elements to its left. Moreover, we say that two entries of π are *consecutive* when their *values* are consecutive integers, whereas we say that they are *adjacent* when their *positions* are consecutive integers.

Suppose that $\sigma = \sigma_1 \sigma_2 \cdots \sigma_n = \alpha \mu_k$, where μ_k is the maximum suffix of consecutive left-to-right maxima of σ (and α is the remaining prefix). For each entry m in μ_k , construct permutations as follows. First, remove the suffix of left-to-right maxima starting with m . Then reinsert the removed elements into the (remaining prefix of the) permutation in all possible ways, according to the following rules:

- The removed elements are reinserted in decreasing order;
- The maximum (i.e. n) is inserted to the immediate right of one of the remaining left-to-right maxima of σ (i.e. a left-to-right maximum to the left of m in σ) or at the beginning of σ ;
- The minimum (i.e. m) is inserted somewhere to the right of $m - 1$.

At this point, consider the prefix of all elements strictly before n and (recursively) compute all its possible preimages.

Using this algorithm we are able to characterize and enumerate permutations having a small number of preimages.

Given $n \geq 0$, let $C_n^{(k)} = \{\sigma \in S_n \mid |\mathbf{PSB}^{-1}(\sigma)| = k\}$, and let $c_n^{(k)} = |C_n^{(k)}|$.

Proposition 3.1.

$$C_n^{(0)} = \{\sigma = \sigma_1 \cdots \sigma_n \mid \sigma_n \neq n\}$$

and

$$c_n^{(0)} = (n-1)(n-1)!.$$

In the subsequent cases, for a given permutation π , we let $LTR(\pi)$ denote the set of the left-to-right maxima of π .

Proposition 3.2. *For $n \geq 3$, the set $C_n^{(1)}$ consists of all permutations of size n ending with n whose left-to-right maxima are consecutive and nonadjacent. More formally,*

$$C_n^{(1)} = \{\pi = \pi_1 \cdots \pi_n \mid \pi_n = n, \text{ there exists } k \geq 0 \text{ such that } LTR(\pi) = \{n-k, \dots, n\} \\ \text{and for every } \pi_i, \pi_j \in LTR(\pi), \pi_i \neq \pi_j, \text{ we have } |j-i| > 1\}.$$

Moreover

$$c_n^{(1)} = \sum_{k=2}^{\lceil \frac{n}{2} \rceil} (n-k)! \binom{n-k-1}{k-2}.$$

The first terms of the sequence $c_n^{(1)}$ (starting from $n = 1$) are 1, 0, 1, 2, 8, 36, 198, ..., and do not appear in the OEIS [9].

Proposition 3.3. *For $n \geq 4$, the set $C_n^{(2)}$ consists of all permutations of size n ending with n whose left-to-right maxima are consecutive and nonadjacent except for the first one, which is required to be nonconsecutive with the second one, and can possibly be adjacent to the second one. More formally,*

$$C_n^{(2)} = \{\pi = \pi_1 \cdots \pi_n \mid \pi_n = n, \text{ there exists } k \geq 0 \text{ such that } LTR(\pi) = \{\pi_1, n-k, \dots, n\}, \\ \text{with } \pi_1 \neq n-k-1, \text{ and for every } \pi_i, \pi_j \in LTR(\pi), \pi_i \neq \pi_j, i, j \neq 1, \text{ we have } |j-i| > 1\}.$$

Moreover

$$c_n^{(2)} = \sum_{k=3}^n \sum_{j=1}^{n-k} \frac{n-k-j+1}{j} (n-k)! \binom{n-j-k}{k-3}. \quad (1)$$

4 Preimages of classes

Next we consider the preimages of principal classes under **PSB**, and we completely determine the permutations ρ for which $\mathbf{PSB}^{-1}(\text{Av}(\rho))$ is a class. Moreover, in all cases we explicitly describe the basis of the resulting class. Our results are also useful in the next section, where we will compose PSB with other sorting algorithms.

We recall that a *permutation class* (or simply *class*) is a set C of permutations for which every pattern contained in a permutation in C is also in C . Every permutation class can be defined by the minimal permutations which do not lie inside it, its *basis*. A *principal permutation class* is a class whose basis consists of a single permutation.

In order to properly describe our results, we will use the notion of shuffle. Let ρ, σ be two sequences of distinct integers. Then we say that a sequence τ is a *shuffle* of ρ and σ if τ contains both ρ and σ as subsequences, and contains no elements other than those of ρ and σ . We denote the set of all possible shuffles of ρ and σ with $\rho \sqcup \sigma$.

Proposition 4.1. *Let ρ be a permutation.*

- If $\rho = \emptyset, 1$ or 12 , then we have that $\mathbf{PSB}^{-1}(\text{Av}(\rho)) = \text{Av}(\emptyset), \text{Av}(1)$, or $\text{Av}(12, 21)$, respectively.
- If $\rho = n\alpha$, for some $\alpha \in S_{n-1}$, $\alpha \neq \emptyset$, then $\mathbf{PSB}^{-1}(\text{Av}(\rho)) = \text{Av}(B)$, where $B = \{n(n+1)\alpha\} \cup \{(n+2)n\tau \mid \tau \in (n+1) \sqcup \alpha, \tau \neq (n+1)\alpha\}$.
- If $\rho = (n-1)\alpha n$, for some $\alpha \in S_{n-2}$, $\alpha \neq \emptyset$, then $\mathbf{PSB}^{-1}(\text{Av}(\rho)) = \text{Av}(B)$, where $B = \{(n-1)n\alpha\} \cup \{(n+1)(n-1)\tau \mid \tau \in n \sqcup \alpha, \tau \neq n\alpha\}$.
- In all the remaining cases, $\mathbf{PSB}^{-1}(\text{Av}(\rho))$ is not a permutation class.

5 Composition with other sorting algorithms

If we execute a sorting algorithm X on a permutation and then execute another sorting algorithm Y on the output permutation, we get a new sorting algorithm, whose associated map is the *composition* of the maps associated with X and Y .

In this section, we investigate permutations which are sortable by a composition of PSB with another sorting algorithm. Specifically, we will consider the algorithms Queuesort, Stacksort, and Bubblesort, whose associated maps will be denoted \mathbf{Q} , \mathbf{S} , and \mathbf{B} , respectively (recall that the algorithm Queuesort uses a queue with a bypass). Our results are summarized in the next proposition.

Proposition 5.1. *Let $\pi \in S_n$.*

- $\mathbf{S} \circ \mathbf{PSB}(\pi) = id_n$ if and only if $\pi \in \text{Av}(2341, 25314, 52314, 45231, 42531, 3\bar{5}241)$.
- $\mathbf{Q} \circ \mathbf{PSB}(\pi) = id_n$ if and only if $\pi \in \text{Av}(3421, 53241, 53214)$.
- $\mathbf{B} \circ \mathbf{PSB}(\pi) = id_n$ if and only if $\pi \in \text{Av}(2341, 3421, 3241, 25314, 52314, 53214)$.
- $\mathbf{PSB} \circ \mathbf{Q}(\pi) = id_n$ if and only if $\pi \in \text{Av}(4231, 2431, 54213)$.
- $\mathbf{PSB} \circ \mathbf{B}(\pi) = id_n$ if and only if $\pi \in \text{Av}(2341, 2431, 3241, 4231, 45213, 54213)$.

6 Pop stacks in parallel with a bypass

We now consider a sorting machine that consists of two pop stacks in parallel, where an entry can bypass the pop stacks and instead go directly to the output. Pop stacks in parallel without a bypass were introduced by Avis and Newborn [3], as already recalled in the Introduction, and later studies include that of Atkinson and Sack [2] and Smith and Vatter [10].

We now informally describe an algorithm to sort a permutation using the above described device. Since we are not interested in composing it with other sorting algorithm, our procedure will provide an output only when the sorting succeeds (otherwise the algorithm just fails).

Let $\pi = \pi_i \pi_2 \cdots \pi_n$ be a permutation in the input at the start. Let S_1, S_2 be the two pop stacks. Suppose that π_i is the next entry in the input.

- If π_i is the next entry needed in the output to obtain the identity, push π_i to the output (i.e. π_i bypasses the pop stacks).
- Else, if the top entry of S_j (for either $j = 1$ or $j = 2$) is the next entry needed in the output to obtain the identity, pop the contents of S_j .
- Else, if π_k is the top entry in the pop stack S_j (for either $j = 1$ or $j = 2$) where $\pi_i = \pi_k - 1$, push π_i into S_j .
- Else, if one of the pop stacks is empty, push π_i into an empty pop stack.

- Else, the sorting algorithm fails.

It is possible to prove that the above algorithm is optimal, as usual in the sense that it sorts all permutations sortable by the machine consisting of two pop stacks in parallel with a bypass option. Using the above procedure, we are able to characterize sortable permutations in terms of forbidden patterns.

Proposition 6.1. *The class of permutations sortable by two pop stacks in parallel with a bypass is*

$$\text{Av}(2341, 25314, 42513, 42531, 45213, 45231, 52314, 642135, 642153).$$

As correctly pointed out by one referee (whom we warmly thank), the inverse of the above class (i.e. the class whose basis is the set of inverses of the above permutations) has a regular insertion encoding [11], thus it is possible to automatically deduce its rational generating function [4], which is

$$\frac{(1-x)(1-2x)(1-4x)}{1-8x+20x^2-18x^3+3x^4}.$$

The sequence corresponding to the sortable permutations of size n has first terms 1, 2, 6, 23, 97, 418, 1800, 7717, 32969, 140558, ... and does not appear in [9] at the time of this writing. There is a potential match for the simple sortable permutations that we boldly put here as a conjecture:

Conjecture. *Let a_n be the number of simple permutations of size n which are sortable by a machine consisting of two pop stacks in parallel where entries are allowed to bypass the pop stacks. Then $a_0 = a_1 = 1$, $a_2 = 2$, $a_n = F_{2n-5} - 1$ if $n \geq 3$ is odd, and $a_n = F_{2n-5}$ if $n > 3$ is even (where F_n is the n -th Fibonacci number).*

References

- [1] M. D. Atkinson (1999): *Restricted permutations*. *Discrete Mathematics* 195, pp. 27–38, doi:10.1016/S0012-365X(98)00162-9.
- [2] M. D. Atkinson & J.-R. Sack (1999): *Pop-stacks in parallel*. *Information Processing Letters* 70, pp. 63–67, doi:10.1016/S0020-0190(99)00049-6.
- [3] D. Avis & M. Newborn (1981): *On pop-stacks in series*. *Utilitas Mathematica* 19, pp. 129–140.
- [4] C. Bean, J. S. Eliasson, T. K. Magnusson, E. Nadeau, J. Pantone & H. Ulfarsson: *Tilings: combinatorial exploration for permutation classes*. Available at <https://github.com/PermutaTriangle/Tilings/>.
- [5] M. Bousquet-Mélou (2000): *Sorted and/or sortable permutations*. *Discrete Mathematics* 225, pp. 25–50, doi:10.1016/S0012-365X(00)00146-1.
- [6] A. Claesson & B. Guðmundsson (2019): *Enumerating permutations sortable by k passes through a pop-stack*. *Advances in Applied Mathematics* 108, pp. 79–96, doi:10.1016/j.aam.2019.04.002.
- [7] D. E. Knuth (1968): *The Art of Computer Programming, Vol. 1*. Addison-Wesley, Boston.
- [8] L. K. Pudwell & R. Smith (2019): *Two-stack-sorting with pop stacks*. *Australasian Journal of Combinatorics* 74, pp. 179–195.
- [9] N. J. A. Sloane: *The Online Encyclopedia of Integer Sequences*. Available at oeis.org.
- [10] R. Smith & V. Vatter (2009): *The enumeration of permutations sortable by pop stacks in parallel*. *Information Processing Letters* 109, pp. 626–629, doi:10.1016/j.ipl.2009.02.014.
- [11] V. Vatter (2012): *Finding regular insertion encodings for permutation classes*. *Journal of Symbolic Computation* 47, pp. 259–265, doi:10.1016/j.jsc.2011.11.002.

Random Generation of Git Graphs

Julien Courtiel

Normandie Univ, UNICAEN, ENSICAEN, CNRS, GREYC, 14000 Caen, France

julien.courtiet@unicaen.fr

Martin Pépin

Normandie Univ, UNICAEN, ENSICAEN, CNRS, GREYC, 14000 Caen, France

martin.pepin@unicaen.fr

Version Control Systems, such as Git and Mercurial, manage the history of a project as a Directed Acyclic Graph encoding the various divergences and synchronizations happening in its life cycle. A popular workflow in the industry, called the *feature branch workflow*, constrains these graphs to be of a particular shape: a unique main branch, and non-interfering feature branches.

Here we focus on the uniform random generation of those graphs with n vertices, including k on the main branch, for which we provide three algorithms, for three different use-cases. The first, based on rejection, is efficient when aiming for small values of k (more precisely whenever $k = O(\sqrt{n})$). The second takes as input any number k of commits in the main branch, but requires costly precalculation. The last one is a Boltzmann generator and enables us to generate very large graphs while targeting a constant k/n ratio. All these algorithms are linear in the size of their outputs.

1 Motivation

In software development, Version Control Systems (VCS in short) such as Git or Mercurial are crucial. They facilitate collaborative work by allowing multiple developers to concurrently contribute to a shared file system. VCS automatically save all project versions over time, along with the associated changes.

Most VCS offer *branching* support, allowing developers to diverge from the main line of development and continue their work independently without affecting the main project line. These branches can be subsequently *merged*, in order to integrate changes from one branch into another, like new features or bug fixes.

In the abstract, the history of a VCS repository can be seen as a Directed Acyclic Graph (DAG), where vertices are the different versions of the project (also named *commits*) and arcs symbolize the changes between two versions. There are no restrictions on the shape of the graphs you can generate with a VCS, but many projects follow a *workflow*, that is a process and a set of conventions that define how branches are created, and how changes are integrated into the main codebase.

The purpose of this paper is to develop an efficient random sampler for DAGs that respect a particular workflow.

One benefit of such a sampler would be to integrate *property-based tests* into VCS development. In these tests, instead of specifying explicit input values and expected outcomes, we define properties that should be satisfied for a wide range of repositories, which are generated randomly during the test. By generating diverse graph structures that adhere to the workflow's specifications, we ensure a comprehensive examination of the VCS's behavior according to plausible scenarios. To give a concrete example based on work by one of the authors [2], a random DAG sampler could experimentally check the effectiveness of `git bisect`, an algorithm that finds the commit where a bug has been introduced.

In this paper, we will take a look at one of the simplest workflows, but one that is widely used in the corporate world: the *feature branch workflow*. In this workflow, the non-main branches do not interfere with each other, and are simply attached to the (unique) main branch. Here is a more formal definition of graphs induced by this workflow. (This definition originally comes from [6].)

Definition 1 (Git graph). A Git feature branch graph (or just Git graph) is a DAG that consists of:

- a main branch, that is a directed path of black vertices.
- potentially several feature branches, that are directed paths that start and end on vertices of the main branch. The set of intermediary vertices is not empty and consists of white vertices. A black vertex cannot be the end point of several feature branches, just one at most (but it can be the starting point of several branches).

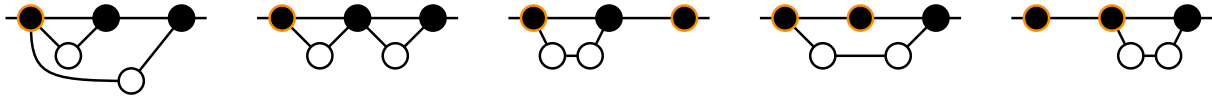


Figure 1: All Git graphs with 5 vertices including 3 black vertices. Edges are oriented from left to right. Free vertices are outlined in orange.

The size of a Git graph γ is its number of vertices. By convention, we assume that there exists a unique Git graph of size 0. Another important parameter is its number of black vertices, and will be denoted by $k(\gamma)$. A black vertex is said to be *free* if there is no feature branch ending on it, *i.e.* its indegree is at most 1. All Git graphs γ of size 5 with $k(\gamma) = 3$ are listed in Figure 1.

The fact that we forbid merges of multiple feature branches into the main one is not a restriction of the VCS, but is advisable to maintain a clearer and more understandable project history, reduce the risk of conflicts, and enhance traceability and maintainability. This restriction is also discussed in [2].

2 The uniform model

2.1 A recursive decomposition

We first describe a recursive decomposition of Git graphs, based on the number of black vertices. Consider the last black vertex v_k of a Git graph of size n and with $k > 1$ black vertices. There are only two possibilities: either v_k is free, or v_k is a merge between the main branch and a feature branch (which is unique, by definition). In the latter case, the feature branch starts with a black vertex, which can be any vertex of the main branch, but v_k . Removing v_k and the potential feature branch attached to it leads to a smaller Git graph.

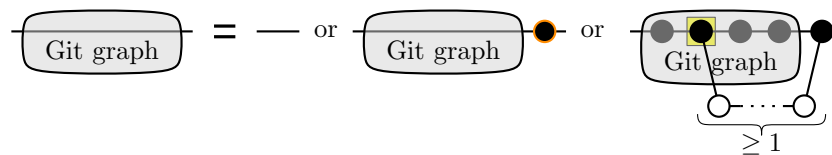


Figure 2: How to decompose a Git graph.

By this reasoning, illustrated by Figure 2, we obtain the induction formula

$$g_{n,k} = g_{n-1,k-1} + \sum_{\ell \geq 1} (k-1)g_{n-1-\ell,k-1}, \quad \text{with } g_{0,k} = \begin{cases} 1 & \text{if } k = 0 \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where $g_{n,k}$ is the number of Git graphs with n vertices, k of them being black.

This induction is sufficient to write a recursive generator (see [7] for the general theory of recursive samplers and [5] for a more modern point of view in the context of the symbolic method). We do not extend on this generator as we will present with Algorithm 2 a more efficient sampler¹, also based on the recursive method.

It is straightforward (especially if you are familiar with the symbolic method [4]) to translate Formula (1) into a differential equation whose solution is the generating function $G(z, u)$ of Git graphs:

$$G(z, u) = 1 + zuG(z, u) + \frac{z^2 u^2}{1-z} \frac{\partial G}{\partial u}(z, u), \quad \text{where } G(z, u) := \sum_{n \geq 0} \sum_{k \geq 0} g_{n,k} z^n u^k. \quad (2)$$

Note that $G(z, u)$ is not analytic at $z = 0$ since the number of Git graphs grows as a factorial (we have $g_{2k-1,k} \geq (k-1)!$ by considering a Git graph with only merge commits and feature branches with 1 white node). For this reason, the previous equation does not seem to be usable for Boltzmann sampling.

2.2 Most Git graphs look alike under the uniform distribution

A large random Git graph is with high probability of the same shape: about half of the commits are on the main branch, and most commits on the main branch are merges of size-1 branches.

Proposition 1. *Let u be any real positive number. Consider γ_n a random Git graph of size n taken with probability $\frac{u^{k(\gamma_n)}}{\sum_{\gamma \text{ Git graph of size } n} u^{k(\gamma)}}$. Then the random variable $\frac{k(\gamma_n)}{n}$ converges in probability to $\frac{1}{2}$ when n goes to $+\infty$. (Note that $u = 1$ corresponds to the uniform distribution).*

The intuition behind this result is that a large number of branches greatly increases the number of ways of connecting them to the main branch, hence favoring graphs with many short branches over ones with fewer but longer branches.

In particular, for any value of u , the average number of commits in the main branch is asymptotically equivalent to $\frac{n}{2}$. This motivates the introduction of a variant of this model which we detail in the next half of this paper, and which allows more control over the number of commits on the main branch.

2.3 A rejection algorithm

Before delving into the next model, it is worth noting that there is an efficient rejection-based sampling algorithm for the case where k is small based on the following inclusion. Consider a variation \mathcal{H} of the model where every black vertex but the first one is the endpoint of a feature branch but it is allowed to have zero commit on a feature branch. Denote by $h_{n,k}$ the number of such graphs with n vertices including k on the main branch. Then Git graphs can be seen as a subset of these graphs by identifying

¹The straightforward recursive algorithm obtained from this recurrence runs in $O(nk)$ time, and requires the precomputation of $O(nk)$ numbers of size $O(n \log n)$, involving $O(n^2 k)$ operations on big integers in total.

empty feature branches pointing at the first commit in \mathcal{H} with free commits in Git graphs. Moreover, whenever $k \leq t\sqrt{n}$, for some constant $t > 0$, we have that

$$c_t \leq \frac{g_{n,k}}{h_{n,k}} \leq 1 \quad \text{for some } c_t > 0 \text{ that depends only on } t.$$

The constant c_t is obtained by considering the class of Git graphs with only one free commit as a subset of Git graphs. Explicit formulas exist for the cardinality of this class and for $h_{n,k}$ and their ratio is $\Theta(1)$ in the regime $k \leq t\sqrt{n}$. This yields Algorithm 1 for sampling uniform Git graphs with a small main branch. This algorithm can be implemented so as to perform $O(k)$ array accesses and $O(k)$ RNG calls² in average.

Algorithm 1 Rejection algorithm for Git graphs with n vertices, k of them being black

- 1: start with a chain of k black vertices
 - 2: arrange uniformly at random $(n - k)$ white vertices into $(k - 1)$ possibly empty chains
 - 3: attach the ends of these chains to the $(k - 1)$ last black vertices
 - 4: attach the start of every chain to a previous black vertex, chosen uniformly at random
 - 5: if any of the empty chains is not attached to the root, start over, otherwise return
-

3 The labeled-main distribution

3.1 Description of the model

Given the disadvantages of the uniform distribution, we propose a new model for random Git graphs that is easier to sample, gives more varied shapes, and with fine control over the number of black vertices. The principle is that a Git graph γ will have a probability to be generated proportional to $u^{k(\gamma)}/k(\gamma)!$ where u is a real positive parameter.

More precisely, we set $\widetilde{G}_n(u) := \sum_{k=1}^n g_{n,k} \frac{u^k}{k!}$ and $\widetilde{G}(z, u) := \sum_{n \geq 0} \widetilde{G}_n(u) z^n$. Thus \widetilde{G} resembles an exponential generating function, but with a scaling of $k!$ instead of a scaling of $n!$. Unlike G defined in the previous section, the function \widetilde{G} is analytic at $z = 0$ (a direct consequence of Theorem 1 below).

Definition 2. *The probability under the labeled-main distribution of a Git graph of size n and with k black vertices is defined as $\frac{u^k z^n}{k! \widetilde{G}(z, u)}$, where z and u are positive parameters inside the domain of convergence of \widetilde{G} .*

This is a multivariate Boltzmann model (exponential in u and ordinary in z). A sampler based on this distribution falls into the category of *Boltzmann generators*, for which a large number of results have been established, facilitating the generation of large objects [3].

By using the Borel transform [1] on Equation (2) with respect to the variable u , that is to say the mapping $\sum_{n,k \geq 0} a_{n,k} z^n u^k \mapsto \sum_{n,k \geq 0} \frac{a_{n,k} z^n u^k}{k!}$, we can obtain a differential equation for $\widetilde{G}(z, u)$:

$$\frac{\partial \widetilde{G}}{\partial u}(z, u) = z \widetilde{G}(z, u) + \frac{z^2 u}{1 - z} \frac{\partial \widetilde{G}}{\partial u}(z, u) \quad \text{and} \quad \widetilde{G}(z, 0) = 1. \quad (3)$$

Solving this differential equation gives a nice formula for \widetilde{G} .

²In practice, considering an RNG call to be $O(1)$ faithfully reflects the runtime performance of such an algorithm. It is thus a realistic complexity model, that we use in the rest of this document. It is however important to note that every RNG call needs to produce about $\log_2(n)$ random bits here.

Theorem 1. The function $\tilde{G}(z, u) = \sum_{n \geq 0} \sum_{k=1}^n g_{n,k} \frac{z^n u^k}{k!}$ is equal to

$$\tilde{G}(z, u) = \left(1 - \frac{z^2 u}{1-z}\right)^{-\frac{1-z}{z}}.$$

By a tedious but straightforward application of the transfer theorem [4], we can compute the average number of black vertices under the labeled-main distribution.

Proposition 2. Let $k(\gamma_n)$ be the number of commits in the main branch of a graph γ_n taken at random with probability $\mathbb{P}(\gamma_n) = \frac{u^{k(\gamma_n)}}{k(\gamma_n)!} \frac{1}{\tilde{G}_n(u)}$. The mean and variance of $k(\gamma_n)$ are asymptotically equivalent to

$$\mathbb{E}(k(\gamma_n)) \sim \frac{1 - \rho_u}{2 - \rho_u} n \quad \text{and} \quad \mathbb{V}(k(\gamma_n)) \sim \frac{\rho_u(1 - \rho_u)}{(2 - \rho_u)^3} n, \quad \text{where } \rho_u = \frac{\sqrt{1 + 4u} - 1}{2u}.$$

Remark that the expected value of the $k(\gamma)/n$ ratio can be any number between 0 and 1/2, depending on the value of u . This is one of the main benefits of the labeled-main distribution: given any $\alpha \in (0, \frac{1}{2})$, we can *tune* u in order to target Git graphs to have αn black vertices (and the variance is quite tight).

3.2 A bijection with cyclariums

The closed formula for \tilde{G} featured in Theorem 1 calls for a combinatorial interpretation. That is why we define a new family of combinatorial objects: the cyclariums.

A *cyclarium* is defined as a set of cycles of k black vertices labeled by $\{1, \dots, k\}$ where each vertex that has not the largest label inside its own cycle carries a non-empty chain of white vertices. See Figure 3 top left to see an illustration of a cyclarium. The set \mathcal{Y} of cyclariums has the natural combinatorial specification

$$\mathcal{Y} = \text{SET}(\mathcal{C}), \quad \text{SEQ}_{\neq 0}(\mathcal{L}) \times \mathcal{C} = \text{CYC}(\mathcal{U} \mathcal{L} \text{SEQ}_{\neq 0}(\mathcal{L})) \tag{4}$$

where $\text{SET}(\cdot)$, $\text{SEQ}_{\neq 0}(\cdot)$ and $\text{CYC}(\cdot)$ are respectively the operators for sets, non-empty sequences and cycles. Consequently the generating function of cyclariums (scaled by $k!$) is also given by the formula of Theorem 1 (for more details on the symbolic method, see [4]).

Proposition 3. The Git graphs with n vertices, k black vertices and f free vertices are in bijection with the cyclariums with n vertices, k black vertices and f cycles.

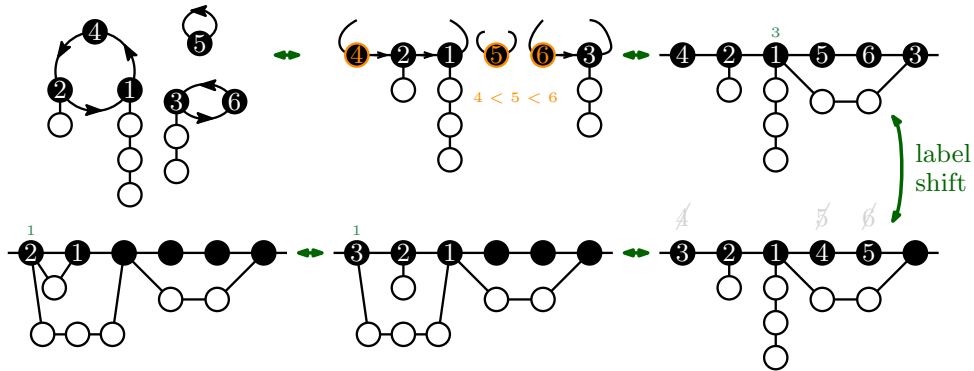


Figure 3: Outline of the bijection between Git graphs and cyclariums

The bijection is depicted in Figure 3. We give a quick overview of the transformation from cyclariums to Git graphs. First, we break each cycle just before the vertex with the largest label, so that they are directed paths. Then we sort these paths according to their largest label, and concatenate them. Now we process the black vertices from right to left. If a chain of white vertices is attached to the current black vertex v , then we connect this chain to the black vertex whose position is given by the label of v . If no chain is attached, we do nothing. Once the vertex has been processed, its label ℓ is deleted and we change all labels x such that $x > \ell$ by $x - 1$. We can check that we eventually obtain a Git graph.

Exploiting the fact that the permutations with f cycles are counted by the Stirling numbers of the first kind, we obtain a closed formula for $g_{n,k}$.

Corollary 1. *The number of Git graphs $g_{n,k}$ of size n and with k black vertices is 1 if $k = n$ and*

$$g_{n,k} = \sum_{f=1}^{k-1} \begin{bmatrix} k \\ f \end{bmatrix} \binom{n-k-1}{k-f-1}$$

for $k < n$, where $\begin{bmatrix} \cdot \\ \cdot \end{bmatrix}$ denotes the (unsigned) Stirling number of the first kind.

Algorithm 2 Exact sampler of Git graphs with n vertices and k black vertices

Additional optional input: f , the number of free vertices

- 1: If f is not given, sample it with probability $\begin{bmatrix} k \\ f \end{bmatrix} \binom{n-k-1}{k-f-1} / g_{n,k}$
 - 2: Generate a random permutation of size k with f cycles
 - 3: Generate a composition of $n - k$ into $k - f$ positive terms
 - 4: Form $k - f$ chains of white vertices whose lengths are given by the previous composition
 - 5: Attach them to the permutation to form a cyclarium
 - 6: Use the bijection from cyclariums to Git graphs
-

The bijection also suggests a sampling algorithm for Git graphs of size n if we fix the number k of black vertices and optionally the number f of free vertices: see Algorithm 2. It runs in $O(n)$ (with some optimization) but requires an expensive precomputation of the Stirling numbers of the first kind. This precomputation is in particular used to generate a uniform permutation of size k with f cycles³. If f must be sampled, we need to precompute $O(k^2)$ numbers of size $O(k \log k)$. If f is given, only $O(f(k - f))$ of them can be precalculated. We could also sample k to get a random generator of fixed size n . In this case, the complexity of precomputing the numbers becomes $O(n^2)$ and those numbers have size $O(n \log n)$.

3.3 A Boltzmann generator

Specification (4) induces a natural Boltzmann generator [3] for cyclariums, and hence by Proposition 3 a Boltzmann generator for Git graphs under the labeled-main distribution. Rather than simply generating a cyclarium of size n and applying the bijection, which would result in $O(n^2)$ complexity, we can mix the two approaches and achieve $O(n + f^2)$ complexity, where f is the number of free vertices (which is logarithmic in n in average). The details are given in Algorithm 3 and illustrated by Figure 4.

Our implementation of this algorithm in Python easily generates graphs larger than 10 million. We also recall that we can carefully choose the parameters z and u to target a size n and a ratio $\alpha \in (0, \frac{1}{2})$, where αn is the number of black vertices.

³The uniform sampler for permutations with a fixed number of cycles is recursive and comes from [9, page 33] but it might be improved by sampling a Poisson-Dirichlet distribution [8, Chapter 3] with a well-chosen θ parameter. We leave this as an open question.

Algorithm 3 Boltzmann sampler under the labeled-main distribution of parameters z and u

```

1:  $f \leftarrow \text{POISSON}(\ln \tilde{G}(z, u))$  ▷ Poisson distribution
2:  $\text{cycle\_lengths} \leftarrow$  array of  $f$  independent  $\text{LOGA}(\frac{uz^2}{1-z})$  ▷ Logarithmic series distribution
3:  $k \leftarrow$  total sum of  $\text{cycle\_lengths}$ 
4:  $g \leftarrow$  directed path of  $k$  black vertices denoted  $v[0], \dots, v[k-1]$  ▷ skeleton of our Git graph
5: while  $k > 0$  do
6:   extract a number  $x$  from  $\text{cycle\_lengths}$  with probability  $x/k$ 
7:   mark  $v[k-x]$ 
8:    $k \leftarrow k-x$ 
9: for  $j$  from 1 to number of black vertices  $-1$  do
10:  if  $v[j]$  is not marked then
11:     $i \leftarrow$  random number between 0 and  $j-1$ 
12:    link  $v[i]$  to a directed path of  $(1 + \text{GEOM}(z))$  white vertices ▷ Geometric distribution
13:    link the last vertex of this path to  $v[j]$ 
14: return  $g$ 

```

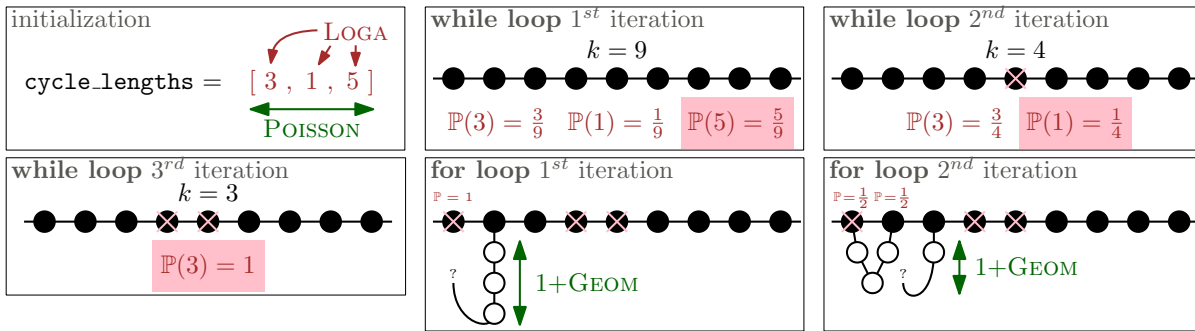


Figure 4: Illustration of the first steps of Algorithm 3.

4 Conclusion

In this work, we have developed three random generators for Git graphs.

A few questions remain unanswered. Firstly, our algorithms are unable to generate graphs for certain values of k efficiently (more precisely when k is in the window $\sqrt{n} \ll k \ll n$, and when $k \geq \frac{n}{2}$). In addition, it would be interesting to obtain an asymptotic estimate of the numbers g_n of Git graphs. The formula in Corollary 1 seems to be a good start to do so. Moreover, we could study potential phase transitions as k evolves as a function of n . In particular, we could investigate how the number of free vertices grows, as well as the gaps between each of them.

Finally, we could study more involved workflows, and enumerate DAGs that adhere to them.

References

[1] Émile Borel (1899): *Mémoire sur les séries divergentes*. *Annales scientifiques de l'École Normale Supérieure* 3e série, 16, pp. 9–131, doi:10.24033/asens.463.

- [2] Julien Courtiel, Paul Dorbec & Romain Lecoq (2023): *Theoretical Analysis of Git Bisect*. *Algorithmica*, doi:10.1007/s00453-023-01194-0.
- [3] Philippe Duchon, Philippe Flajolet, Guy Louchard & Gilles Schaeffer (2004): *Boltzmann samplers for the random generation of combinatorial structures*. *Combinatorics, Probability & Computing* 13(4-5), pp. 577–625, doi:10.1017/S0963548304006315.
- [4] Philippe Flajolet & Robert Sedgewick (2009): *Analytic Combinatorics*. Cambridge University Press, doi:10.1017/CBO9780511801655. Available at <https://algo.inria.fr/flajolet/Publications/book.pdf>.
- [5] Philippe Flajolet, Paul Zimmermann & Bernard Van Cutsem (1994): *A calculus for the random generation of labelled combinatorial structures*. *Theoretical Computer Science* 132(1-2), pp. 1–35, doi:10.1016/0304-3975(94)90226-7.
- [6] Romain Lecoq (2024): *Le feu ça brûle et l'informatique ça bugue : Combustion et régression dans les graphes*. Ph.D. thesis. In French, work in progress.
- [7] Albert Nijenhuis & Herbert Wilf (1978): *Combinatorial Algorithms: For Computers and Hand Calculators*, 2nd edition. Academic Press, Inc., USA, doi:10.1016/C2013-0-11243-3.
- [8] Jim Pitman (2006): *Combinatorial stochastic processes*. *Lecture Notes in Mathematics* 1875, Springer-Verlag, Berlin, doi:10.1007/b11601500. Lectures from the 32nd Summer School on Probability Theory held in Saint-Flour, July 7–24, 2002, With a foreword by Jean Picard.
- [9] Herbert S. Wilf (1999): *East Side, West Side . . . - an introduction to combinatorial families-with Maple programming*. Available at <https://www2.math.upenn.edu/~wilf/eastwest.pdf>.

Bijjective Enumeration and Sign-Imbalance for Permutation Depth and Excedances

Sen-Peng Eu

Department of Mathematics
National Taiwan Normal University
Taiwan, ROC

Chinese Air Force Academy
Taiwan, ROC

speu@math.ntnu.edu.tw

Tung-Shan Fu

Department of Applied Mathematics
National Pingtung University
Taiwan, ROC

tsfu@mail.nptu.edu.tw

Yuan-Hsun Lo

yhlo@mail.nptu.edu.tw

We present a simplified variant of Biane's bijection between permutations and 3-colored Motzkin paths with weight that keeps track of the inversion number, excedance number and a statistic so-called depth of a permutation. This generalizes a result by Guay-Paquet and Petersen about a continued fraction of the generating function for depth on the symmetric group \mathfrak{S}_n of permutations. In terms of weighted Motzkin path, we establish an involution on \mathfrak{S}_n that reverses the parities of depth and excedance numbers simultaneously, which proves that the numbers of permutations with even and odd depth (excedance numbers, respectively) are equal if n is even and differ by the tangent number if n is odd. Moreover, we present some interesting sign-imbalance results on permutations and derangements, refined with respect to depth and excedance numbers.

1 Introduction

1.1 Preliminaries and background

Let \mathfrak{S}_n be the symmetric group of permutations on $[n] := \{1, 2, \dots, n\}$. Given a permutation $\sigma = \sigma(1) \cdots \sigma(n) \in \mathfrak{S}_n$, an *inversion* of σ is a pair (i, j) of indices such that $1 \leq i < j \leq n$ and $\sigma(i) > \sigma(j)$. The *inversion number* of σ , denoted by $\text{inv}(\sigma)$, is defined to be the number of inversions of σ . Petersen and Tenner [11] presented a statistic called *depth* for a Coxeter group, which is defined in terms of factorizations of the elements into reflections. In the case of symmetric group, the reflections of \mathfrak{S}_n are the transpositions (ij) , $1 \leq i < j \leq n$, and the depth of a permutation $\sigma \in \mathfrak{S}_n$, denoted by $\text{depth}(\sigma)$, is defined as

$$\text{depth}(\sigma) = \min \left\{ \sum_{r=1}^k (j_r - i_r) : \sigma = (i_1 j_1)(i_2 j_2) \cdots (i_k j_k) \right\}. \quad (1.1)$$

They obtained a simple formula for the calculation of depth

$$\text{depth}(\sigma) = \sum_{\sigma(i) > i} (\sigma(i) - i), \quad (1.2)$$

which turns out to be one half of the *total displacement* of σ [9, Problem 5.1.1.28] (also called *Spearman's disarray* for \mathfrak{S}_n in [3]).

For any two sequences $\{\gamma_h\}_{h \geq 0}$ and $\{\lambda_h\}_{h \geq 1}$, let $\mathfrak{F}(\gamma_h, \lambda_h)$ denote the series in z defined by

$$\mathfrak{F}(\gamma_h, \lambda_h) = \frac{1}{1 - \gamma_0 z - \frac{\lambda_1 z^2}{1 - \gamma_1 z - \frac{\lambda_2 z^2}{1 - \gamma_2 z - \frac{\lambda_3 z^2}{\ddots}}}} \quad (1.3)$$

These are Jacobi-type continued fractions (also known as *J-fractions*). Flajolet [6] gave a combinatorial interpretation of (1.3) as the generating function for weighted Motzkin paths.

A *Motzkin path* of length n is a lattice path from the origin to the point $(n, 0)$ staying weakly above the x -axis, using the *up step* $(1, 1)$, *down step* $(1, -1)$ and *horizontal step* $(1, 0)$. Let U , D and H denote an up step, a down step and a horizontal step, respectively. Let \mathcal{M}_n denote the set of Motzkin paths of length n .

For a Motzkin path $\mu = x_1 \cdots x_n \in \mathcal{M}_n$, the *height* of each step x_j is the maximum y -coordinate achieved by x_j . For $x \in \{U, D, H\}$ and a nonnegative integer h , let $x^{(h)}$ denote a step x at height h . The *weight* $\omega(\mu)$ of μ is defined to be the product of the weight $\omega(x_j)$ of each step x_j for all $j \in [n]$. By the *area* under μ , denoted by $\text{area}(\mu)$, we mean the area of the region enclosed by μ and the x -axis. Guay-Paquet and Petersen [8] defined a weight of μ by setting

$$\omega(x_i) = \begin{cases} ht^{(2h-1)/2} & \text{if } x_i = U^{(h)} \text{ or } D^{(h)}; \\ (2h+1)t^h & \text{if } x_i = H^{(h)} \end{cases}$$

and established a surjective map $\Phi: \sigma \mapsto \mu$ of \mathfrak{S}_n onto \mathcal{M}_n with $\text{depth}(\sigma) = \text{area}(\mu)$, which yields the following generating function for permutations with respect to depth:

$$\sum_{n \geq 0} \sum_{\sigma \in \mathfrak{S}_n} t^{\text{depth}(\sigma)} z^n = \mathfrak{F}(\gamma_h, \lambda_h) \quad (1.4)$$

with $\lambda_h = h^2 t^{2h-1}$ and $\gamma_h = (2h+1)t^h$, i.e.,

$$\sum_{n \geq 0} \sum_{\sigma \in \mathfrak{S}_n} t^{\text{depth}(\sigma)} z^n = \frac{1}{1 - z - \frac{tz^2}{1 - 3tz - \frac{4t^3 z^2}{1 - 5t^2 z - \frac{9t^5 z^2}{1 - 7t^3 z - \frac{16t^7 z^2}{1 - \dots}}}}}$$

Guay-Paquet and Petersen [8] commented that map $\Phi: \mathfrak{S}_n \rightarrow \mathcal{M}_n$ is due to Foata and Zeilberger [7] and that the statistic depth coincides with a statistic, Edif, studied by Clarke, Steingrímsson and Zeng [2].

Following [13], a *3-colored Motzkin path* is a Motzkin path with three kinds of horizontal steps, denoted by H_1 , H_2 and H_3 . Using a double labeling scheme, Biane [1] established a bijection between permutations and 3-colored Motzkin paths with weight, which keeps track of the number of inversions. We refer to an alternative version of this bijection given by Sokal and Zeng in [13, Section 6.2]. Elizalde [5] used this bijection to study the joint distribution of multiple statistics on \mathfrak{S}_n . Using Biane's method, we propose a multivariate weight function on the steps of a 3-colored Motzkin path and turn the surjective map $\Phi: \mathfrak{S}_n \rightarrow \mathcal{M}_n$ of Guay-Paquet and Petersen into a bijection that gives a generalization of (1.4). (This resolves a conjecture raised by Petersen in an unpublished note [10].)

1.2 Main results

Let \mathscr{W}_n denote the set of 3-colored Motzkin paths of length n with a weight function ω defined by setting $\omega(H_3^{(0)}) = p$ and

$$\begin{cases} \omega(U^{(h)}) \in \{st^{2h-1}, st^{2h-1}q, \dots, st^{2h-1}q^{h-1}\}; \\ \omega(D^{(h)}) \in \{q^{2h-1}, q^{2h}, \dots, q^{3h-2}\}; \\ \omega(H_1^{(h)}) \in \{st^h q^h, st^h q^{h+1}, \dots, st^h q^{2h-1}\}; \\ \omega(H_2^{(h)}) \in \{t^h q^h, t^h q^{h+1}, \dots, t^h q^{2h-1}\}; \\ \omega(H_3^{(h)}) = pt^h q^{2h} \end{cases} \quad (1.5)$$

for all $h \geq 1$.

Let $\sigma = \sigma(1) \cdots \sigma(n) \in \mathfrak{S}_n$. We say that an index $i \in [n]$ is an *excedance* if $\sigma(i) > i$ and a *fixed point* if $\sigma(i) = i$. Let $\text{exc}(\sigma)$ and $\text{fix}(\sigma)$ denote the number of excedances and fixed points of σ , respectively. For all positive integers k , we use the notations for q -integer $[k]_q = 1 + q + \cdots + q^{k-1}$ and $[0]_q = 0$. One of our main results is the following bijection.

Theorem 1.1. *There is a bijection $\Psi : \mathfrak{S}_n \rightarrow \mathscr{W}_n$ such that a permutation $\sigma \in \mathfrak{S}_n$ with i inversions, j fixed points, k excedances and a depth of ℓ is carried to a 3-colored Motzkin path $\mu \in \mathscr{W}_n$ with weight $\omega(\mu) = q^i p^j s^k t^\ell$. Therefore, we have*

$$\sum_{n \geq 0} \left(\sum_{\sigma \in \mathfrak{S}_n} q^{\text{inv}(\sigma)} p^{\text{fix}(\sigma)} s^{\text{exc}(\sigma)} t^{\text{depth}(\sigma)} \right) z^n = \mathfrak{F}(\gamma_h, \lambda_h) \quad (1.6)$$

with $\lambda_h = s[h]_q^2 (qt)^{2h-1}$ and $\gamma_h = ((1+s)[h]_q + pq^h)(qt)^h$.

Remarks. By the weight given in (1.5), notice that a horizontal step $x \in \{H_2^{(h)}, H_3^{(h)}\}$ can be distinguished by the q -factor of $\omega(x)$ for each $h \geq 1$. That is, if $\omega(x)$ contains q^{h+d} then $x = H_2^{(h)}$ ($H_3^{(h)}$, respectively) if $0 \leq d \leq h-1$ ($d = h$, respectively). Thus, the bijection $\Psi : \mathfrak{S}_n \rightarrow \mathscr{W}_n$ can still be established with $H_2^{(h)}, H_3^{(h)}$ combined and the parameter p omitted.

The classical *Euler numbers* E_n , defined by

$$\begin{aligned} \sum_{n \geq 0} E_n \frac{z^n}{n!} &= \tan z + \sec z \\ &= 1 + z + \frac{z^2}{2!} + 2\frac{z^3}{3!} + 5\frac{z^4}{4!} + 16\frac{z^5}{5!} + 61\frac{z^6}{6!} + 272\frac{z^7}{7!} + 1385\frac{z^8}{8!} + \cdots, \end{aligned}$$

count the the number of *alternating permutations* in \mathfrak{S}_n , i.e., $\sigma \in \mathfrak{S}_n$ such that $\sigma_1 > \sigma_2 < \sigma_3 > \cdots \sigma_n$. The numbers E_{2n} are called the *secant numbers* and the numbers E_{2n+1} are called the *tangent numbers*.

Based on the bijection $\Psi : \mathfrak{S}_n \rightarrow \mathscr{W}_n$, we shall establish an involution on \mathfrak{S}_n in terms of weighted Motzkin paths, which reverses the parities of depth and the number of excedances simultaneously. This proves that the numbers of permutations with even and odd depth (excedance numbers, respectively) are equal if n is even and differ by E_n (E_n up to sign, respectively) if n is odd.

Theorem 1.2. *There is an involution $\sigma \mapsto \sigma'$ on \mathfrak{S}_n satisfying*

$$\text{depth}(\sigma) - \text{depth}(\sigma') = \text{exc}(\sigma) - \text{exc}(\sigma') = \text{inv}(\sigma) - \text{inv}(\sigma') \in \{1, 0, -1\}$$

and resulting in the following identities

$$(i) \sum_{\sigma \in \mathfrak{S}_n} (-1)^{\text{depth}(\sigma)} = \begin{cases} E_n & \text{for } n \text{ odd} \\ 0 & \text{for } n \text{ even.} \end{cases}$$

$$(ii) \sum_{\sigma \in \mathfrak{S}_n} (-1)^{\text{exc}(\sigma)} = \begin{cases} (-1)^{\frac{n-1}{2}} E_n & \text{for } n \text{ odd} \\ 0 & \text{for } n \text{ even.} \end{cases}$$

Note that the identity in Theorem 1.2(ii) is a classical result of Euler. When the fixed points of σ are ignored, we present some interesting results on the sign imbalances of permutations and derangements, refined with respect to depth and excedance numbers.

Theorem 1.3. For $n \geq 1$, we have

$$\sum_{\sigma \in \mathfrak{S}_n} (-1)^{\text{inv}(\sigma)} s^{\text{exc}(\sigma)} t^{\text{depth}(\sigma)} = (1-st)^{n-1}. \quad (1.7)$$

A *derangement* of size n is a permutation in \mathfrak{S}_n that contains no fixed point. Let $\mathcal{D}_n \subset \mathfrak{S}_n$ be the set of derangements of size n . Define

$$F_n = F_n(s, t) = \sum_{\sigma \in \mathcal{D}_n} (-1)^{\text{inv}(\sigma)} s^{\text{exc}(\sigma)} t^{\text{depth}(\sigma)}. \quad (1.8)$$

Several of the initial polynomials $F_n(s, t)$ are listed below:

$$\begin{aligned} F_1(s, t) &= 0, \\ F_2(s, t) &= -st, \\ F_3(s, t) &= s(1+s)t^2, \\ F_4(s, t) &= s^2t^2 - s(1+s)^2t^3, \\ F_5(s, t) &= -2s^2(1+s)t^3 + s(1+s)^3t^4. \end{aligned}$$

Collected in powers of t , the coefficient s -polynomials of $F_n(s, t)$ for $2 \leq n \leq 9$ are listed in Table 1. We obtain a neat expression for the sign imbalance of the joint distribution of depth and excedance numbers over \mathcal{D}_n .

Table 1: The s -polynomial coefficients of $F_n(s, t)$ in powers of t for $2 \leq n \leq 9$.

	t	t^2	t^3	t^4	t^5	t^6	t^7	t^8
F_2	$-s$							
F_3		$s(1+s)$						
F_4		s^2	$-s(1+s)^2$					
F_5			$-2s^2(1+s)$	$s(1+s)^3$				
F_6			$-s^3$	$3s^2(1+s)^2$	$-s(1+s)^4$			
F_7				$3s^3(1+s)$	$-4s^2(1+s)^3$	$s(1+s)^5$		
F_8				s^4	$-6s^3(1+s)^2$	$5s^2(1+s)^4$	$-s(1+s)^6$	
F_9					$-4s^4(1+s)$	$10s^3(1+s)^3$	$-6s^2(1+s)^5$	$s(1+s)^7$

Theorem 1.4. We have

$$\sum_{n \geq 1} \left(\sum_{\sigma \in \mathcal{D}_n} (-1)^{\text{inv}(\sigma)} s^{\text{exc}(\sigma)} t^{\text{depth}(\sigma)} \right) z^n = \sum_{k \geq 1} (-1)^k \left(\sum_{i=0}^{k-1} \binom{k-1}{i} s^{1+i} (1+s)^{k-1-i} z^{k+1+i} \right) t^k.$$

References

- [1] P. Biane (1993): *Permutations suivant le type d'excédance et le nombre d'inversions et interprétation combinatoire d'une fraction continue de Heine*. *European J. Combin.* 14(4), pp. 277–284, doi:10.1006/eujc.1993.1031.
- [2] R.J. Clarke, E. Steingrímsson & J. Zeng (1997): *New Euler–Mahonian statistics on permutations and words*. *Adv. Appl. Math.* 18(3), pp. 237–270, doi:10.1006/aama.1996.0506.
- [3] P. Diaconis & R.L. Graham (1977): *Spearman's footrule as a measure of disarray*. *J. Roy. Statist. Soc. Ser. B* 39(2), pp. 262–268, doi:10.1111/j.2517-6161.1977.tb01624.x.
- [4] D. Dumont (1995): *Further triangles of Seidel–Arnold type and continued fractions related to Euler and Springer numbers*. *Adv. Appl. Math.* 16(3), pp. 275–296, doi:10.1006/aama.1995.1014.
- [5] S. Elizalde (2018): *Continued fractions for permutation statistics*. *Discrete Math. Theor. Comput. Sci.* 19(2), doi:10.23638/DMTCS-19-2-11. arXiv:1703.08742.
- [6] P. Flajolet (1980): *Combinatorial aspects of continued fractions*. *Discrete Math.* 32(2), pp. 125–161, doi:10.1016/0012-365X(80)90050-3.
- [7] D. Foata & D. Zeilberger (1990): *Denert's permutation statistic is indeed Euler–Mahonian*. *Stud. Appl. Math.* 83(1), pp. 31–59, doi:10.1002/sapm199083131.
- [8] M. Guay-Paquet & T.K. Petersen (2014): *The generating function for total displacement*. *Electron. J. Combin.* 21(3):P3.37, doi:10.37236/4329.
- [9] D.E. Knuth (1998): *The Art of Computer Programming, vol. 3*, second edition. Addison Wesley Longman Publishing Co., Inc., USA.
- [10] T.K. Petersen: *A note for Doron: the generation function for total displacement (Spearman's footrule) and inversion*. Available at <https://sites.math.rutgers.edu/~zeilberg/mamarim/mamarimhtml/noga12yleFeedback.pdf>.
- [11] T.K. Petersen & B.E. Tenner (2015): *The depth of a permutation*. *J. Combin.* 6(1–2), pp. 145–178, doi:10.4310/JOC.2015.v6.n1.a9.
- [12] N.J.A. Sloane: *The On-Line Encyclopedia of Integer Sequences*. Available at <http://oeis.org>.
- [13] A.D. Sokal & J. Zeng (2022): *Some multivariate master polynomials for permutations, set partitions, and perfect matchings, and their continued fractions*. *Adv. Appl. Math.* 138:102341, doi:10.1016/j.aam.2022.102341.

Square-Triangle Tilings: Lift & Flip to Sample?

Thomas Fernique

HSE
Moscow, Russia
fernique@hse.ru

Olga Mikhailovna Sizova

Semenov Institute of Chemical Physics
Moscow, Russia
olstet@mail.ru

We introduce an elementary transformation called flips on tilings by squares and triangles and conjecture that it connects any two tilings of the same region of the Euclidean plane.

1 Introduction

We consider the tilings of a simply-connected bounded region of the Euclidean plane by two tiles: a unit square and a regular unit triangle (Fig. 1).

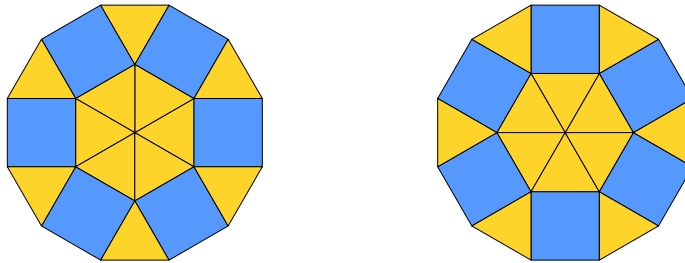


Figure 1: Two square-triangle tilings of the same region.

The goal is to introduce an elementary local transformation that allows to travel the space of all the possible tilings, in order to sample them using mixing times techniques as in, e.g., [2]. To do this, we shall embed these tilings in a larger set obtained by adding a rhombus tile, and show how these tilings can be naturally seen as discrete surfaces in 4-dimensional Euclidean space. This is a work-in-progress.

2 Lift

Let $j = \exp \frac{2i\pi}{3}$ and $a = \exp \frac{2i\pi}{12}$ be the third and twelfth roots of the unity, respectively. For $k = 0, 1, 2$, define the complex numbers $u_k := j^k$ and $v_k = a \cdot j^k$, seen as vectors in the Euclidean plane. Without loss of generality, the edges of the square and triangles tiles are directed by the u_i 's and v_i 's.

We want to associate with every edge $w_k \in \{u_k, v_k\}$ a vector \hat{w}_k in some higher dimensional space \mathbb{R}^n so that, when travelling around a tile, the sum of the vectors associated with the traversed edges (with negative sign if travelled in backward direction) is zero. This does not yield any restriction for the squares since each pair of parallel edges are travelled in opposite directions, hence their sum is always zero. The

same holds for any parallelogram (this will be used later). For the triangles, this yields two conditions (Fig. 2):

$$\hat{u}_0 + \hat{u}_1 + \hat{u}_2 = 0 \quad \text{and} \quad \hat{v}_0 + \hat{v}_1 + \hat{v}_2 = 0.$$

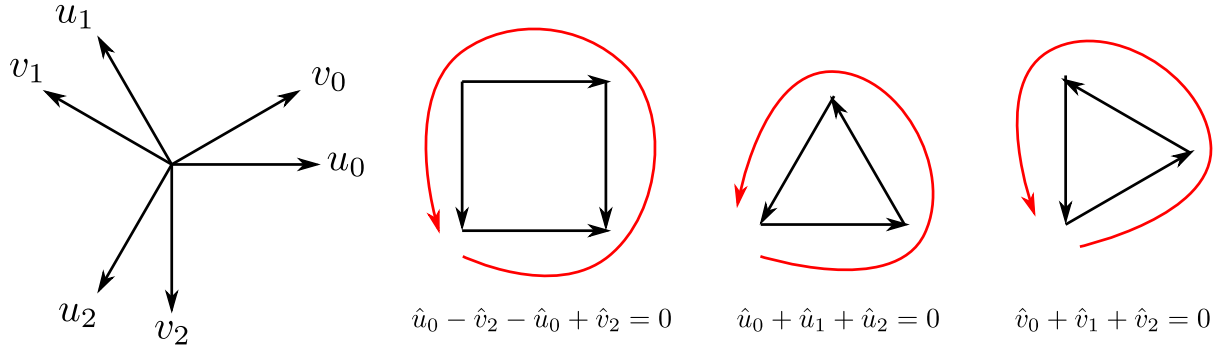


Figure 2: The vectors which define tiles (left) and the conditions on their lifts.

Since we have 2 constraints on 6 vectors, that is, a linear system of rank 4, it is natural to take vectors in \mathbb{R}^4 (higher dimension is useless, lower dimension will loose generality). We will define them as vectors in \mathbb{C}^2 . For example, take

$$\forall i \in \{0, 1, 2\}, \quad \hat{u}_k := (j^k, 0) \quad \text{and} \quad \hat{v}_k := (0, j^k).$$

The condition on the sum of vectors around each tile ensures (by induction) that the sum of the vectors associated with the traversed edges along every cycle is zero. In particular, if we map an arbitrary vertex s_0 of the tiling to $0 \in \mathbb{C}^2$, then the sum $h(s) \in \mathbb{C}^2$ of vectors associated with the edges along a path from v_0 to a vertex s of the tiling does not depend on the path. The vector $h(s)$ is called the *height* of s . This allows to see any square-triangle tiling as a sort of “stepped” two-dimensional surface embedded in \mathbb{C}^2 . Figure 3 illustrates this.

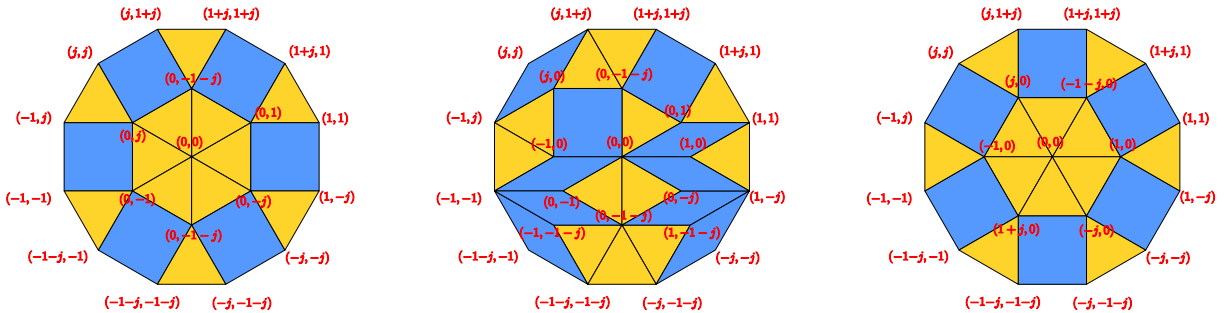


Figure 3: Square-triangle tilings with the height of every vertex indicated.

Every triangle has edges directed either by three u_k 's or by three v_k 's. We call it a *u-triangle* in the former case, a *v-triangle* in the latter case. The way the \hat{u}_k 's have been chosen ensure that, in a lifted tiling, the vertices of every *u-triangle* have all the same second coordinate (in \mathbb{C}); this common coordinate defines the *elevation* of the triangle. These three vertices differ by their first coordinate (in \mathbb{C}). The barycenter of these coordinates is called the *position* of the triangle. The situation is similar for *v-triangles*, with first and second coordinates exchanged.

3 Flip

In the context of tilings, it is common to introduce elementary transformations on tilings generally called *flips*. For example, in the case of tilings by rhombi, a flip denotes the half-turn rotation of a hexagon formed by three rhombi. Square-triangle tilings lack of such a “natural” flip. However, such a local transformation has been defined in [3], which features a third tile, namely a rhombus. This flip is depicted in Figure 4.

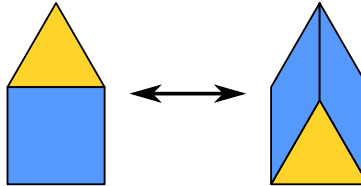


Figure 4: A flip.

It may seem strange to involve a new tile that takes us out of the space of square-triangle tilings. But if we look on this flip in the lifted tiling¹, it can be seen as an exchange between the faces of a triangular prism: the upper triangle and one “square” lateral face are replaced by the lower triangle and the other two “square” lateral faces (Figure 5). We are arguing here that squares and rhombi play the same role, differing only in the way they are represented in the Euclidean plane. To emphasize this, we use the same color for both tiles. We will also still use the term square-triangle tilings even when rhombi are present, with the special cases of lack of rhombi being called *pure* square-triangle tilings. The position and elevation of a triangle are affected as follows by a flip:

- the position does not change;
- the elevation changes by $\pm j^k$ for some $k \in \{0, 1, 2\}$.

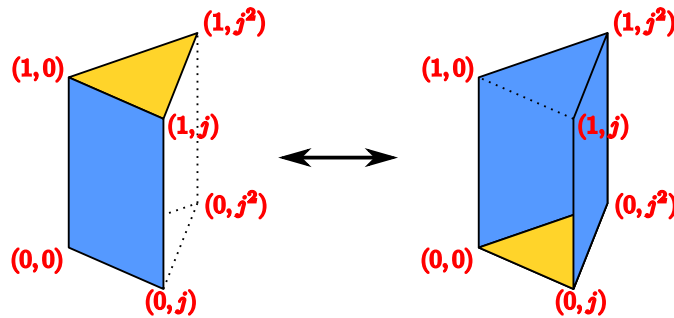


Figure 5: A flip seen in the lift.

4 Flip-connectivity

In [3] is defined the *zipper move*, which is a sequence of flips between two pure square-triangle tilings (while intermediary tilings may not be pure).

¹As for squares, pairs of parallel edges in rhombi do not yield any restriction for the lift.

Here, we would like to prove that, any two square-triangle tilings of the same region (not necessarily pure, that is, with possibly rhombus tiles) can be connected by a sequence of flips. This would yield the above claim as a corollary.

Recall that the position of a triangle is unchanged by a flip, only its elevation. Hence, if flip-connectivity holds, then for any pair T and T' of tilings of the same region, to each triangle of T corresponds a triangle of T' with the same position. Transforming T into T' by flips thus amounts to equal the elevations of pairs of triangles with identical position. In other words, we can “spot” a triangle in every possible tilings. Figure 6 illustrates this.

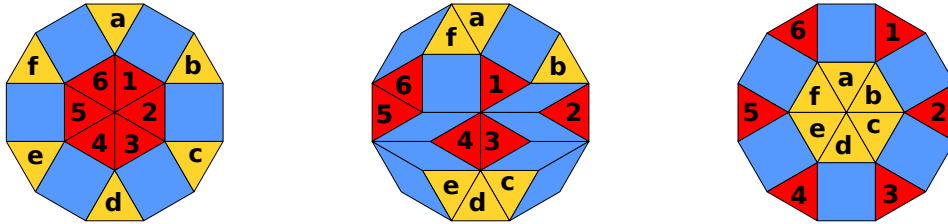


Figure 6: Every pair of triangles with the same position have the same label.

Hence, the wanted result will follow if we can prove the following result (a similar result has been proven for rhombus tilings in [1]):

Conjecture 1 *If $T \neq T'$ are two tilings of the same region, then there is a triangle in T that can be moved by a flip towards its elevation in T' , so that no triangle which has already at the same elevation in T and T' is moved.*

References

- [1] O. Bodini, Th. Fernique & É. Rémy (2008): *A Characterization of flip-accessibility for rhombus tilings of the whole plane*. *Information and Computation* 206, pp. 1065–1073, doi:10.1016/j.ic.2008.03.008.
- [2] M. Luby, D. Randall & A. Sinclair (2001): *Markov Chain Algorithms for Planar Lattice Structures*. *SIAM Journal on Computing* 31, pp. 167–192, doi:10.1137/S0097539799360355.
- [3] M. Oxborrow & Ch. L. Henley (1993): *Random square-triangle tilings: A model for twelvefold-symmetric quasicrystals*. *Physical review B* 48, pp. 6966–6998, doi:10.1103/PhysRevB.48.6966.

Restricted Permutations Enumerated by Inversions

Atli Fannar Franklín

Department of Mathematics
University of Iceland
Reykjavík, Iceland
aff6@hi.is

Anders Claesson

Department of Mathematics
University of Iceland
Reykjavík, Iceland
akc@hi.is

Christian Bean

School of Computer Science and Mathematics
Keele University
Keele, United Kingdom
c.n.bean@keele.ac.uk

Henning Úlfarsson

Department of Computer Science
Reykjavík University
Reykjavík, Iceland
henningu@ru.is

Jay Pantone

Department of Mathematical and Statistical Sciences
Marquette University
Milwaukee, WI, USA
jay.pantone@marquette.edu

Permutations are usually enumerated by size, but new results can be found by enumerating them by inversions instead, in which case one must restrict one's attention to indecomposable permutations. In the style of the seminal paper by Simion and Schmidt [6], we investigate all combinations of permutation patterns of length at most 3.

1 Introduction

To enumerate permutations by their number of inversions we need the set of permutations with a given number of inversions to be finite. This is generally not the case, since in many cases one can add a new maximal element to the end of the permutation to get a new one with equally many inversions. To get around this we introduce the notion of decomposability. We say that a permutation $\pi = \pi_1 \dots \pi_n$ is decomposable if there exists an index $i < n$ such that $\pi_1 \dots \pi_i$ is a permutation of the elements $1, \dots, i$. If we let $\rho = \pi_1 \dots \pi_i$ and $\tau = (\pi_{i+1} - i) \dots (\pi_n - i)$ we denote this $\pi = \rho \oplus \tau$. In the same vein, permutations that are not decomposable are called indecomposable. With this definition every permutation can be factored into a set of indecomposable factors, and we call these factors its components.

An inversion in a permutation $\pi = \pi_1 \dots \pi_n$ is a pair of indices (i, j) such that $i < j$ and $\pi_i > \pi_j$. The number of inversions in a permutation π will be denoted $\text{inv}(\pi)$. The following lemma highlights a relation between these two concepts.

Lemma 1 *Let π be a permutation on n elements and c components. Then $\text{inv}(\pi) \geq n - c$.*

Crucially this means that if $c = 1$ then $\text{inv}(\pi) \geq n - 1$. Thus the set of indecomposable permutations with k inversions is finite since it is contained in the set of all permutations on $k + 1$ elements or fewer. With this in mind, we define I_k to be the set of all indecomposable permutations with exactly k inversions.

Next we recall some notions related to patterns of permutations. We say two sequences a_1, \dots, a_m and b_1, \dots, b_m are order-isomorphic if $a_i < a_j$ holds if and only if $b_i < b_j$. For permutations $\pi = \pi_1 \dots \pi_n$, $\tau = \tau_1 \dots \tau_m$ we say that π contains the pattern τ if there exists a set of indices i_1, \dots, i_m such that $\pi_{i_1} \dots \pi_{i_m}$ is order-isomorphic to $\tau_1 \dots \tau_m$. We call such a set of indices an *occurrence* of the pattern τ . If π has no occurrence of τ we will say that π *avoids* the pattern τ . We will denote the subset of I_k containing the permutations avoiding τ by $I_k(\tau)$. Similarly we will denote the subset of I_k containing permutations avoiding several patterns $\tau_1, \tau_2, \dots, \tau_r$ by $I_k(\tau_1, \tau_2, \dots, \tau_r)$.

2 Single patterns

As examples we have that $I_k(1)$ and $I_k(21)$ have no non-empty elements and $|I_k(12)|$ is the characteristic function of the triangular numbers, listed in the OEIS as A010054. The remaining single patterns of length 3 are then 123, 132, 213, 231, 312 and 321. We consider the reverse complement of a permutation: written out explicitly, this is $\pi_1 \dots \pi_n \mapsto (n+1-\pi_n)(n+1-\pi_{n-1}) \dots (n+1-\pi_1)$. Consider an inversion on $i < j$ in π . The values $\pi_i > \pi_j$ get mapped to $n+1-\pi_i, n+1-\pi_j$ at indices $n+1-i, n+1-j$ in the image. Thus the size of the elements is inverted, but so is their order. Therefore the number of inversions remains constant under reverse complement. We also see that π is decomposable if and only if its reverse complement is, so the reverse complement is an involution on I_k for every k . It maps the pattern 132 to the pattern 213, so from this we see that $|I_k(213)| = |I_k(132)|$, hence we only have to consider one of these patterns. This is similar to what happens when counting pattern avoiding permutations classically, however we must compose reversion and complement for the argument to work. Normally this line of reasoning shows that the number of 132 and 231 avoiding permutations are the same through reversion, but we will see that this is no longer the case when enumerating by inversions. A similar argument can be made to show that mapping to the inverse permutation preserves inversions and indecomposability as well, and in fact the symmetries are generated by these two maps.

This means we only have to consider the patterns 123, 132, 231 and 321. We will start with 132. To do this, we make use of a well known bijection on permutations. The inversion table of a permutation $\pi = \pi_1 \dots \pi_n$ is the sequence $b_1 b_2 \dots b_n$, where b_i is the number of values after π_i in π that are smaller than π_i . The image of this bijection is given by the set of all sequences such that the first value is $\leq n-1$, the next $\leq n-2$ and so on. We call such sequences subdiagonal. Furthermore, elements in a subdiagonal sequence that are equal to their maximum possible value are called diagonal elements.

Theorem 2 $|I_k(132)|$ counts partitions on k elements, listed as A000041 on the OEIS.

To work with 231-avoiding permutations we define the *skew-sum* of two permutations $\pi_1 \dots \pi_n$ and $\tau_1 \dots \tau_m$, as the permutation on $n+m$ elements given by $\pi \ominus \tau = (\pi_1 + m) \dots (\pi_n + m) \tau_1 \dots \tau_m$. We will say that π is skew-decomposable if it can be written as $\pi = \tau_1 \ominus \tau_2$ for non-empty τ_1, τ_2 , and skew-indecomposable otherwise.

Theorem 3 $|I_k(231)|$ counts fountains on k coins. A fountain of coins is an arrangement of coins in rows such that the bottom row is full (that is, there are no “holes”), and such that each coin in a higher row rests on two coins in the row below. This is listed as A005169 on the OEIS.

Enumerating 321-avoiding permutations by size and inversions (allowing decomposable permutations) has been investigated in [2]. A generating function is derived, but as we need a bijective map from $I_k(321)$ to our target set for what comes later, we will have to give a different proof.

Theorem 4 $|I_k(321)|$ counts parallelogram polyominoes with k cells, listed as A006958 on the OEIS.

This alternative view on parallelogram polyominoes opens up a formula for efficiently computing new terms of the series, taking $\mathcal{O}(k^2)$ time and space to compute $|I_k(321)|$.

Theorem 5 $|I_k(321)| = a_{k,1}$ where

$$a_{n,m} = \begin{cases} 1 & \text{if } n = 0 \\ \sum_{i=1}^n a_{n-i,i} & \text{if } m = 1 \\ a_{n,m-1} + \sum_{i=m}^n a_{n-i,i} & \text{otherwise} \end{cases}$$

Not only can this view help with computing new terms, but it also gives rise to new bijective correspondences. Consider fountains of coins where we only count coins in even rows. We can still place coins as we like, but when tallying the number we only count those in the bottom row, those 2 rows up, 4 rows up and so on. Such fountains with n counted coins will be called even fountains of size n . In [1] the problem of mapping parallelogram polyominoes to such fountains is tackled. It is shown that there are equally many by an algebraic argument, but it is left as an open question at the end whether there is any bijective proof. Using $I_k(321)$, a bijective proof can be found. Since Theorem 4 is proved by mapping parallelogram polyominoes bijectively to $I_k(321)$, it suffices to map $I_k(321)$ bijectively to even fountains of size k .

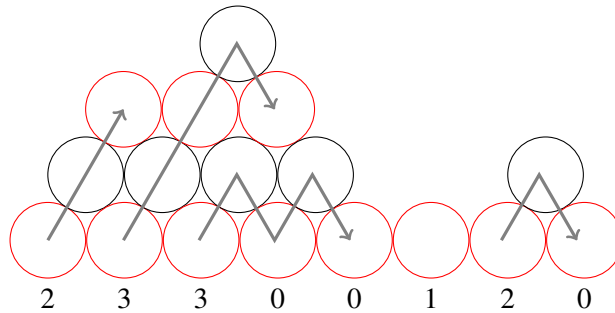
Theorem 6 $I_k(321)$ maps bijectively to even fountains of size k .

The proof of Theorem 5 is bijective, so it suffices to map the even fountains to the sequences described there. We will describe a map taking even fountains to such sequences.

Write out the fountain in the usual manner (see picture), calling coins in even rows red and the ones in odd rows black for convenience. For each coin in the bottom row, going from left to right, we do the following procedure repeatedly:

- If we are on a red coin, we remove it and move to the coin above and to the right. If there is no such coin we stop.
- If we are on a black coin, we remove it and move to the coin above and to the right. If there is no such coin we move to the coin below and to the right instead.

Once done with a coin in the bottom row, we write down the number of red coins removed during this procedure. This produces a sequence of numbers, which we then append a single zero to. We claim this produces a sequence of the desired form.



This leaves us only with $I_k(123)$, the only single pattern giving rise to a sequence not in the OEIS.

Theorem 7 $|I_k(123)|$ counts indecomposable subdiagonal sequences where non-diagonal elements are in decreasing order. Let

$$c_{n,m,k} = \begin{cases} 0 & \text{if } n < 0 \text{ or } k < 0 \\ 1 & \text{if } n = k = 0 \\ c_{n-1,m,l-n+1} + \sum_{i=0}^{\min(n-2,m-1)} c_{n-1,i,k-i} & \text{otherwise} \end{cases}$$

Then the total number of permutations with k inversions (including decomposable ones) avoiding 123 can be calculated as $\sum_{n=0}^{k+1} c_{n,n,k}$.

To obtain the number of such permutations that are indecomposable, subtract the k -th coefficient of $(\sum_{i \geq 0} x^{i(i+1)/2})^2$.

3 Several patterns

We now investigate permutations avoiding several patterns. Some groups of patterns are restrictive enough to make all supersets of those patterns trivially determined. For example $|I_k(123, 321)|$ quickly decays to zero by the Erdős-Szekeres theorem, making all supersets of 123, 321 easy to determine. We have two more such pairs of patterns.

Theorem 8 $|I_k(231, 321)| = 1$ and the unique permutation with k inversions is $k12 \dots (k-1)$.

Theorem 9 $|I_k(231, 312)| = |I_k(12)|$.

By utilizing the symmetries we have, the only pairs of patterns left to investigate are 123, 231 and pairs containing 132.

Theorem 10 $|I_k(123, 231)|$ counts fountains of k coins where the missing coins with respect to a full triangular fountain form a rectangle (removing no coins counts as a rectangle). The generating function is given by $\sum_{i \geq 1} x^{\binom{i}{2}} + \sum_{i \geq 1} \sum_{j \geq 1} \sum_{\ell=0}^{\min(i,j)-1} x^{\binom{i+1}{2} + \binom{j+1}{2} - \binom{\ell+1}{2}}$.

This leaves us with four pairs, pairing 132 with any of the patterns 123, 213, 231 and 321. We now tackle them in that order.

Theorem 11 $|I_k(132, 123)|$ enumerates the Pascal triangle with the first column removed, which is listed as A135278 on the OEIS. It has generating function $\sum_{n \geq 0} x^{n(n+3)/2} ((x+1)^{n+2} - x^{n+2})$.

Our next result involves a kind of partition called a Gorenstein partition. Gorenstein partitions are partitions whose maximal chains are all of the same size when regarded as order ideals of $\{1, 2, \dots\} \times \{1, 2, \dots\}$. This definition is rather unwieldy for our purposes, so we first translate this condition.

Lemma 12 A partition ρ is Gorenstein if and only if $\rho_i + i$ is constant across the indices i that satisfy $\rho_i \neq \rho_{i+1}$, letting $\rho_{|\rho|} = 0$.

Theorem 13 $|I_k(132, 213)|$ — counts Gorenstein partitions of k . This is listed as A117629 on the OEIS.

Theorem 14 Let $\mu \vDash s$ denote that μ is a composition of s . Then $|I_k(132, 213)|$ has generating function $\sum_{s \geq 0} \sum_{\mu \vDash s, |\mu| \neq 1} x^{\binom{s}{2} - \sum_{m \in \mu} \binom{m}{2}}$. This means $|I_k(132, 213)|$ also enumerates finite sequences of positive integers of length > 1 such that k equals the second elementary symmetric function of the values of the sequence, as noted in the OEIS entry.

The generating function is not useful for actually computing new terms in the sequence, but we can use the following recurrence instead. By ignoring all but the first and last \sqrt{n} summands in the recurrence below, as they are zero, we can compute the n -th value in $\mathcal{O}(n^{2.5})$ time and $\mathcal{O}(n^2)$ space.

Theorem 15 The number of Gorenstein partitions with sum n , and thus also the number of elements in $|I_n(132, 213)|$, is given by the sum $\sum_{d=0}^n f(n, d)$ where

$$f(n, d) = \begin{cases} 0 & \text{if } n < 0 \\ 1 & \text{if } n = 0 \\ \sum_{k=1}^d f(n - k(d+1-k), d-k) & \text{otherwise} \end{cases}$$

Theorem 16 $|I_k(132, 231)|$ counts partitions on k elements with distinct parts, this is listed as A000009 on the OEIS.

Theorem 17 $|I_k(132, 321)|$ counts partitions on k elements with equal values. This is in turn equal to the number of divisors of k . This is listed as A000005 on the OEIS.

4 More than two patterns

Most of the remaining pattern combinations are trivially deduced as some subset of the patterns forces the sequence to die out or contain only very specific permutations. We consider here the complement of those cases.

Theorem 18 $|I_k(123, 132, 231)| = 1$.

Theorem 19 $|I_k(123, 132, 213)|$ enumerates the Pascal triangle, read by diagonals, offset by two elements. This means it reads the binomials $\binom{n}{k}$ in increasing order by the sum $n+k$, with each set being read in increasing order by n , and $|I_0(123, 132, 213)|$ starts at $\binom{1}{1}$. Furthermore its generating function can be written as $1 + \sum_{d \geq 3} x^{\binom{d-1}{2}} \sum_{n=2}^d \binom{n}{d-n} x^{n-2}$.

Theorem 20 $|I_k(132, 213, 231)|$ counts the odd divisors of k , which is listed as A001227 on the OEIS.

Theorem 21 $|I_k(132, 213, 321)| = |I_k(132, 321)|$.

Theorem 22 $|I_k(123, 132, 213, 231)|$ enumerates the Pascal triangle, except all values > 1 are replaced by 0. This is listed as on A103451 on the OEIS and has the generating function $\sum_{i \geq 0} x^{i(i+1)/2} + x^{(i+1)(i+4)/2}$.


Acknowledgements

The algorithm in [4] was used to generate elements of all the sequences above, which helped tremendously in finding the formulas and other results in this paper.

References

- [1] Peter Bala (2019): *FOUNTAINS OF COINS AND SKEW FERRERS DIAGRAMS*. Available at https://oeis.org/A161492/a161492_1.pdf.
- [2] E. Barucci, A. Del Lungo, E. Pergola & R. Pinzani (2001): *Some permutations with forbidden subsequences and their inversion number*. *Discrete Mathematics* 234(1), pp. 1–15, doi:10.1016/S0012-365X(00)00359-9. Available at <https://www.sciencedirect.com/science/article/pii/S0012365X00003599>.
- [3] Anders Claesson, Vít Jelínek & Einar Steingrímsson (2012): *Upper bounds for the Stanley–Wilf limit of 1324 and other layered patterns*. *Journal of Combinatorial Theory, Series A* 119(8), pp. 1680–1691, doi:10.1016/j.jcta.2012.05.006. Available at <https://www.sciencedirect.com/science/article/pii/S0097316512000891>.
- [4] Scott Effler & Frank Ruskey (2003): *A CAT algorithm for generating permutations with a fixed number of inversions*. *Information Processing Letters* 86(2), pp. 107–112, doi:10.1016/S0020-0190(02)00481-7. Available at <https://www.sciencedirect.com/science/article/pii/S0020019002004817>.
- [5] Thomas E. Mason (1912): *On the Representation of an Integer as the Sum of Consecutive Integers*. *The American Mathematical Monthly* 19(3), pp. 46–50, doi:10.1080/00029890.1912.11997664.
- [6] Rodica Simion & Frank W. Schmidt (1985): *Restricted Permutations*. *European Journal of Combinatorics* 6(4), pp. 383–406, doi:10.1016/S0195-6698(85)80052-4. Available at <https://www.sciencedirect.com/science/article/pii/S0195669885800524>.
- [7] Richard P. Stanley (1978): *Hilbert functions of graded algebras*. *Advances in Mathematics* 28(1), pp. 57–83, doi:10.1016/0001-8708(78)90045-2. Available at <https://www.sciencedirect.com/science/article/pii/0001870878900452>.

Bijections between Variants of Dyck Paths and Integer Compositions

Manosij Ghosh Dastidar 

TU Wien
Wien, Austria

Institute of Discrete Mathematics and Geometry
manosij.dastidar@tuwien.ac.at

Michael Wallner 

TU Wien
Wien, Austria

Institute of Discrete Mathematics and Geometry
michael.wallner@tuwien.ac.at
dmg.tuwien.ac.at/mwallner

We give bijective results between several variants of lattice paths of length $2n$ (or $2n - 2$) and integer compositions of n , all enumerated by the seemingly innocuous formula 4^{n-1} . These associations lead us to make new connections between these objects, such as congruence results.

Keywords: Integer compositions, lattice paths, Dyck paths, bijections

1 Introduction

We explore several links between different variants of integer compositions and generalizations of Dyck paths. Let us first introduce these objects. First, an *integer composition* of a nonnegative integer n is a tuple (n_1, \dots, n_k) of nonnegative integers such that $n = n_1 + \dots + n_k$. Note that *integer partitions* are integer compositions such that $n_1 \geq n_2 \geq \dots \geq n_k$, (equivalently, the order of summands is not significant). Second, a *Dyck path* is a sequence of steps up $u = (1, 1)$ and down $d = (1, -1)$ that starts at the origin, ends on the x -axis, and never crosses the x -axis. All classes of paths we consider will start at the origin and consist of steps u and d , but the constraints will differ. Natural classes are *Dyck walks* that have no constraints, i.e., they may end anywhere and go below the x -axis, and *Dyck bridges* (also known as *grand Dyck paths*) that have to end on the x -axis but may go below it.

Our results reveal a series of bijections, shown in Figure 1, connecting these structures through a common enumeration formula

$$4^{n-1}.$$

This gives the corresponding integer sequence A000302 in the OEIS¹ many new combinatorial interpretations. Additionally, these bijections often map natural statistics onto each other, such as the height of peaks and the number of crossings of the x -axis.

2 Bijections involving integer compositions

We start with the simple initial bijection, connecting pairs of compositions and Dyck walks.

Proposition 2.1. *There exists a natural bijection between pairs of compositions of n and Dyck walks of length $2n - 2$.*

¹The On-Line Encyclopedia of Integer Sequences: <http://oeis.org/>

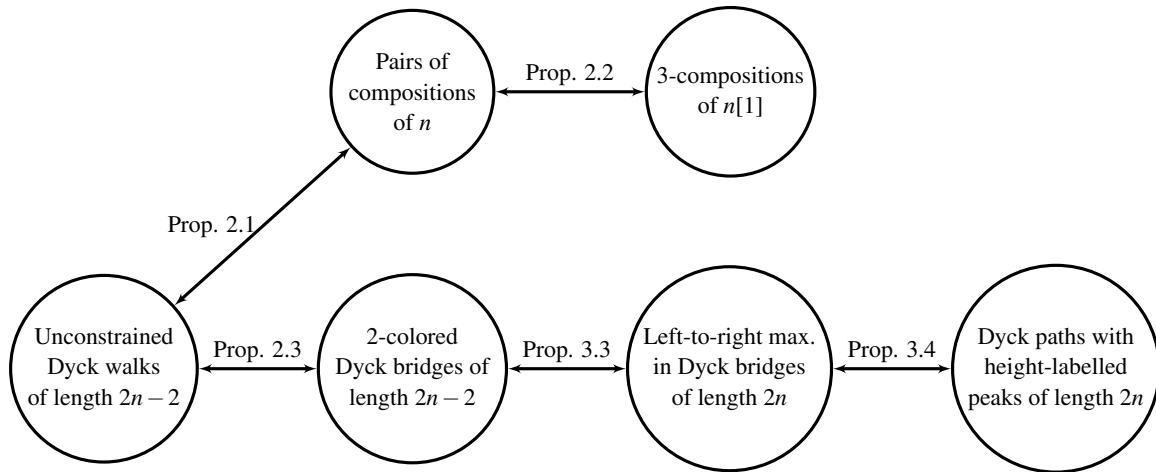


Figure 1: Bijections proved in this paper of classes of paths and integer compositions that are all enumerated by 4^{n-1} .

Proof. Let a pair (A, B) of two compositions of n be given. First, we convert each composition to a binary sequence: for each element k in a composition, append $k - 1$ zeros followed by a 1. By construction, both of these sequences have to end in 1. So we remove these ones and then concatenate the binary sequences, with A 's sequence coming first. Finally, after replacing each 0 by an up step u and each 1 by a down step d the claim follows. For the reverse direction cut the walk in the middle into two parts, and re-add the ones. \square

A k -composition is an integer compositions whose parts come in k different colors with the restriction that the last part of the composition is of the first color; see [1]. We will consider only 3-compositions.

Proposition 2.2. *There exists a natural bijection between 3-compositions of n and pairs of compositions of n .*

Proof. By definition, the parts of 3-compositions have three labels 1, 2, and 3. Anticipating the result, we introduce a notion of left and right: remove the labels of color 1, use label L for color 2, and label R for color 3.

Now, we describe a map from 3-compositions of n to pairs of compositions of n . First, we create two identical copies. In the first copy, we remove the labels R and add the parts labeled by L to the next part. If the next part has also a label L , then the addition continues to the next part, etc. This gives a composition A without any labels. Similarly, in the second copy, we remove the labels L and add the parts labeled by R to the next part. Again, if the next part has also a label R , then the addition continues, and we get a composition B without any labels. Observe that the size of both compositions has not changed. Therefore, (A, B) is a pair of compositions of n .

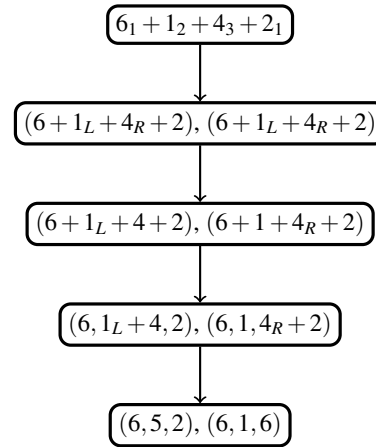
To prove that this map is in fact a bijection, let us consider an arbitrary pair (A, B) of compositions of n . The key statistic to consider is the *run of identical parts*: Let $A = (a_1, a_2, \dots, a_{\ell_A})$ and $B = (b_1, b_2, \dots, b_{\ell_B})$. A run is a sequence of maximal length such that $a_1 = b_1, a_2 = b_2$, and so on. If $a_1 \neq b_2$ we say the run has length 0.

For the inverse map, we will describe a recursive algorithm, which reduces the sizes of A and B in the pair (A, B) step-by-step and builds a 3-composition C . We start with an empty 3-composition C . Depending on the first parts of A and B we distinguish three cases:

1. If $a_1 = b_1$ then we attach a_1 with label 1 to C , remove a_1 from A , and remove b_1 from B .
2. If $a_1 > b_1$ then we attach b_1 with color 2 to C , replace a_1 in A by $a_1 - b_1$, and remove b_1 from B .
3. If $a_1 < b_1$ then we attach a_1 with color 3 to C , remove a_1 from A , and replace b_1 in B by $b_1 - a_1$.

We repeat this process with the new values of A , B , and C . As the sizes of A and B decrease in each step by at least one, this process terminates. Moreover, note that in each step both parts decrease by the same size. Hence, in the last step the process ends with case one where both parts are equal, and therefore the final part gets label 1, as required in the definition of 3-compositions. \square

Example 2.3. Consider the 3-composition $6_1 + 1_2 + 4_3 + 2_1$ of $n = 13$. First, we remove the labels of color 1, use label L for color 2, and label R for color 3. Second, we create two identical copies. In the first copy, we remove the labels R and add the parts labeled by L to the next part. If the next part has also a label L , then the addition to the next part continues, etc. Similarly, in the second copy, we remove the labels L and add the parts labeled by R to the next part. This gives a pair of compositions of n without any labels, and we have shown in Proposition 2.2 that this is in fact a bijection.



3 Bijections involving Dyck paths

Let us now consider more complicated classes of Dyck paths. All of them use the concept of a *peak*, which is a consecutive pattern ud . The first class we consider are *Dyck paths with a marked peak*, which are classical Dyck paths enriched by a marker on a distinguished peak. Two such paths are different, if the underlying paths differ, or, if the paths are the same then two different peaks are marked. Therefore, the number of these paths is equal to the number of peaks in all Dyck paths, whose enumeration is well-known; see, e.g., [3, Section 6.1].

Theorem 3.1. *There is an explicit bijection between Dyck paths with a marked peak of height h and Dyck bridges starting with a d step and $h - 1$ crossings of the x -axis preserving the length. Therefore, the number of peaks in all Dyck paths of length $2n$ is equal to $\binom{2n-1}{n}$; see A001700.*

Proof. Let D be a Dyck path with marked peak at height h . Using this peak, we decompose the path D into a left part L from the origin to this peak and a right part R from this peak to the end: $D = LR$ such that L ends with u and R starts with d . In L we perform a last-passage decomposition, cutting at the u leaving a certain altitude for the last time; while in R we perform a first-passage decomposition, cutting at the d bringing us down to a new altitude for the first time; see Figure 2. More formally, we have

$$L = L_1 u L_2 u \dots u L_h u, \quad R = d R_h d R_{h-1} d \dots d R_1, \tag{1}$$

where L_i and R_i for $i = 1, \dots, h$ are Dyck paths. Now, we pair the paths $L_i u$ and $d R_i$ with the same index and map them as follows to non-empty Dyck paths:

$$D_i = u L_i d R_i.$$

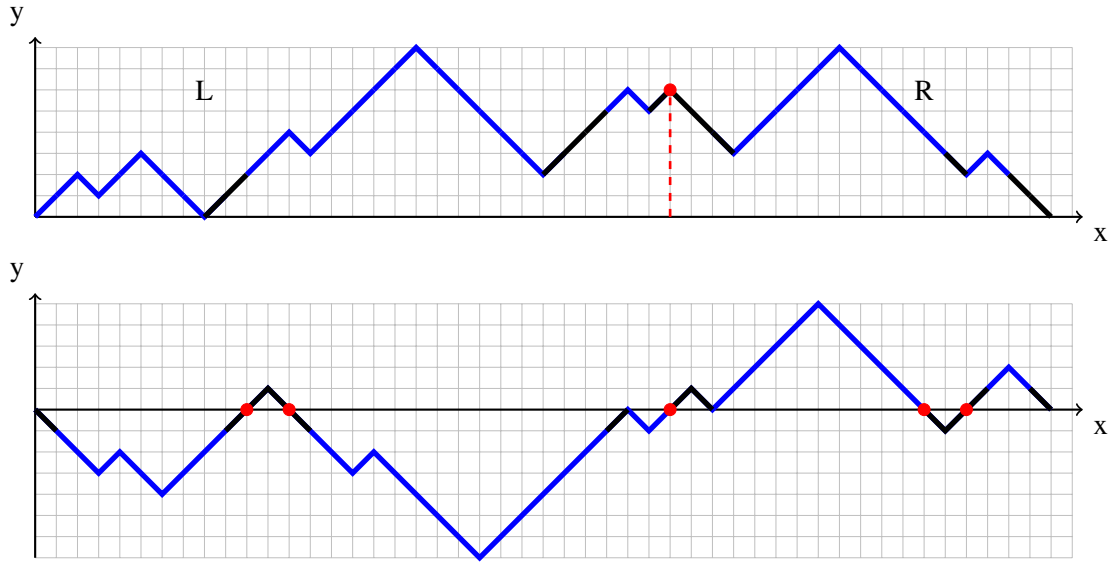


Figure 2: A Dyck path with a marked peak (red dot) at height 6 and its image under the bijection from Theorem 3.1 given by a Dyck bridge starting with a d step and $5 = 6 - 1$ crossings (red dots). The black steps are used in the last-passage (resp., first-passage) decomposition in the proof.

Then we concatenate these parts, after mapping each part with odd index to its image obtained by any fixed bijection φ between Dyck paths and negative Dyck paths. This gives the Dyck bridge

$$\varphi(D_1)D_2\varphi(D_3)D_4 \dots \varphi(D_{h-1})D_h \tag{2}$$

when h is even. For odd h it ends with $\varphi(D_h)$. This bridge starts with a down step d and crosses the x -axis $h - 1$ times, as claimed. Second, let a Dyck bridge starting with a d step be given. We cut at each crossing of the x -axis and recover the components D_i and $\varphi(D_i)$. Hence, it is straightforward to recover the components L_i and R_i and to rebuild the Dyck path D with marked peak.

Finally, bridges of length $2n$ are counted by $\binom{2n}{n}$, as there is an equal number of up and down steps. Since half of them start with a down step, we get

$$\frac{1}{2} \binom{2n}{n} = \binom{2n-1}{n}. \quad \square$$

We return now to the bijections of Figure 1 and we connect our results with *2-colored bridges*; see [2, Section 6.4]. They are defined as the concatenation of two bridges such that the first bridge is colored in color 1 and the second one in color 2. Note that contrary to [2], we allow each part to be empty. Hence, it is easy to see that its generating function is equal to the square of the generating function $B(z) = \frac{1}{\sqrt{1-4z^2}}$ of bridges:

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

Proposition 3.2. *There is an explicit bijection between 2-colored Dyck bridges and unconstrained Dyck walks of the same length $2n$.*

Proof. Recall the classical notion of a Dyck meander, which is defined as the prefix of a Dyck path, i.e., a path that starts at 0, never goes below the x -axis, but does not necessarily end on the x -axis. We will

use repeatedly that there is an explicit bijection between Dyck bridges and Dyck meanders of the same length $2n$; see [6]. Sometimes, it will be necessary to transform a meander further into a negative meander, by flipping all steps, i.e., exchanging u by d and vice versa.

We distinguish four cases. First, the first and second bridges are non-empty. The idea is that the change in color corresponds to the last crossing of the x -axis. For this purpose we transform the second bridge into a meander or negative meander and attach it to the first bridge such that the attached meander continues on the other side of the x -axis. We can easily reverse this procedure by cutting at the last crossing of the x -axis. All the other cases will have no crossings. Second, if the first bridge is non-empty and the second one is empty, we transform the first bridge into a meander. Third, if the first bridge is empty and the second one is non-empty, we transform the second bridge into a negative meander. Finally, if both bridges are empty, we map them to the empty walk. \square

We continue, with *Dyck bridges with marked strict left-to-right maximum*. A *strict left-to-right maximum* is any peak ud that has a greater height than all peaks to its left. We called it marked in the previous sense, when it is attached with a distinguished marker.

Proposition 3.3. *There is an explicit bijection between Dyck bridges of length $2n$ with marked strict left-to-right maximum at height h and 2-colored Dyck bridges of length $2n - 2$ with $h - 1$ crossings of the x -axis in color 1.*

Proof. Let us start with a Dyck bridge with marked strict left-to-right maximum of length $2n$. Then, we cut the bridge at the first return to the x -axis after this maximum. The second part to the right is a bridge, which we give color 2. Onto the first part we apply a similar idea as in the bijection of Theorem 3.1. As before, we cut the path at the marked left-to-right maximum into a left and right part given by LR , such that L ends with u and R starts with d . Now, we decompose it similar to (1) into

$$\begin{aligned} L &= \varphi(L_1)u\varphi(L_2)u\dots u\varphi(L_h)u, \\ R &= dR_hdR_{h-1}d\dots R_2d, \end{aligned}$$

where h is the height of the peak, and L_i and R_i are Dyck paths. Note that in this case R ends with a d step and contains only $h - 1$ Dyck paths R_i . As in the proof of Theorem 3.1 we form Dyck paths $D_i = uL_idR_i$ for $i = 2, \dots, h$. Finally, we remove the two steps of the marked peaks, and get the following bridges with two steps less (compare with Equation (2)):

$$\varphi(L_1)D_2\varphi(D_3)D_4\dots\varphi(D_{h-1})D_h,$$

when h is even. For odd h it ends with $\varphi(D_h)$.

The mapping may be reversed by repeating the aforementioned steps in reverse order. \square

We end the bijections shown in Figure 1 by the following link to *Dyck paths with height-labeled peak* that are Dyck paths in which one peak is associated with a label from $\{1, 2, \dots, h\}$, where h is the height of the specific peak.

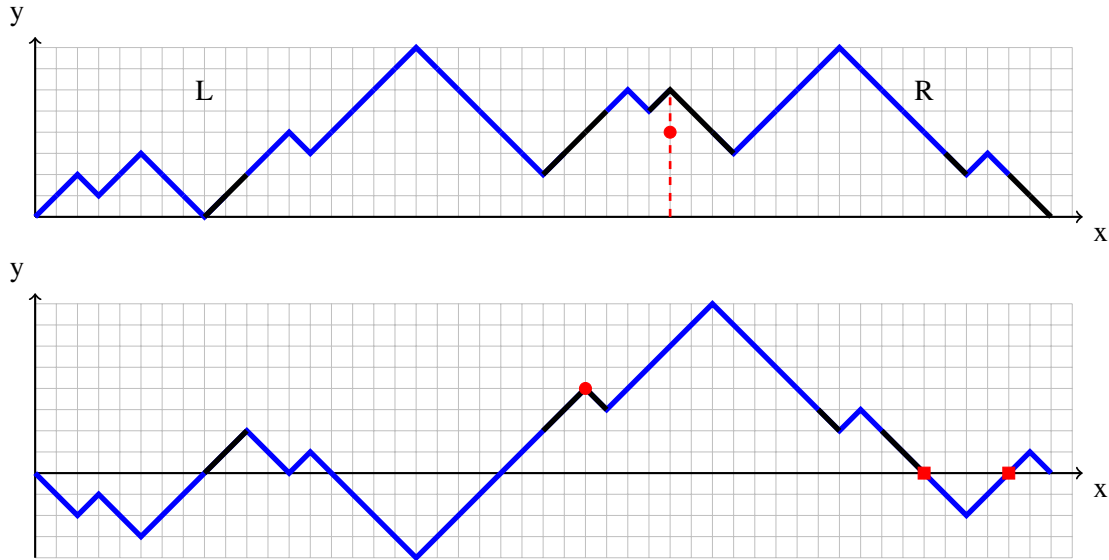


Figure 3: A Dyck path with a height-labeled peak (red dot) with label 4 at height 6 and its image under the bijection from Proposition 3.4 given by a Dyck bridge with marked strict left-to-right maximum (red dot) at height 4 and $2 = 6 - 4$ crossings (red squares).

Proposition 3.4. *There is an explicit bijection between Dyck paths with height-labeled peak with label μ at height h and Dyck bridges of the same length with marked strict left-to-right maximum at height μ and $h - \mu$ crossings of the x -axis after this maximum.*

Proof. This bijection follows directly from the one described in the proof of Theorem 3.1, whose notation we will use here. The difference is that here we concatenate the (positive and negative) Dyck paths differently; see Figure 3.

Let a Dyck path with height-labeled peak be given. Let h be the height of this peak and $\mu \in \{1, \dots, h\}$ be its label. First, we apply the bijection φ onto all parts L_i in (1). From that we get the following bridge in which the height-labeled peak is now a left-to-right maximum (underlined):

$$\varphi(L_1) \cup \varphi(L_2) \cup \dots \cup \varphi(L_h) \underline{d} R_h d R_{h-1} d \dots d R_1.$$

Next, we transform this bridge, such that in the end the height-label μ constitutes the height of the left-to-right maximum. For this purpose, we create and concatenate the paths D_i and $\varphi(D_i)$ in an alternating fashion at the end:

$$\varphi(L_1) \cup \varphi(L_2) \cup \dots \cup \varphi(L_\mu) \underline{d} R_\mu d R_{\mu-1} d \dots d R_1 \varphi(D_{\mu+1}) D_{\mu+2} \varphi(D_{\mu+3}) \dots D_h, \tag{3}$$

when $h - \mu$ is even. Otherwise, the last D_h is replaced by $\varphi(D_h)$.

For the reverse direction, let a Dyck bridge with marked left-to-right maximum be given. It is then straightforward to decompose it into (3) and to reverse the steps above to build a Dyck path. The left-to-right maximum becomes the height-labeled peak, labeled by its current height. Observe that the height-labeled peak is lifted by the number of crossings of the x -axis to the left of this peak. \square

4 Conclusion and Outlook

Having established links between the internal structures of Dyck paths and integer compositions, it is only natural to ask whether important theorems from one subject can be transported to the other. When we talk about integer compositions or partitions we are keen to see arithmetic properties in those structures. In the long version of this work [4], we give further bijective links and we show that such arithmetic results also exist in lattice paths. In particular, we are pleased to note that:

Theorem 4.1 ([4, Theorem 3.8]). *Let $D_r(n)$ be the number of Dyck paths with semi-length n and with exactly r peaks for every reached height. Then $D_r(n) \equiv 0 \pmod{r+1}$ for $n > r$.*

In the opposite direction we also want to see if important theorems in integer compositions and partitions can “generate” theorems in the world of lattice paths. In 2020, Kim, Kim, and Lovejoy [5] observed the phenomenon of *parity bias* in partitions, where they showed that: if $p_o(n)$ denotes the number of partitions of n with more odd parts than even parts and if $p_e(n)$ denotes the number of partitions of n with more even parts than odd parts, then $p_o(n) > p_e(n)$.

In a subsequent article we will show that the analogous theorem is true even for integer compositions. Furthermore, we will demonstrate that even for Dyck paths a similar result holds, when segregating paths with respect to whether they have more peaks at odd or even heights.

References

- [1] George E. Andrews (2007): *The Theory of Compositions, IV: Multicompositions*. *The Mathematics student* 76, p. 25.
- [2] Cyril Banderier, Markus Kuba & Michael Wallner (2024): *Phase transitions of composition schemes: Mittag-Leffler and mixed Poisson distributions*. To appear in *Ann. Appl. Probab.* arXiv:2103.03751.
- [3] Emeric Deutsch (1999): *Dyck path enumeration*. *Discrete Math.* 204(1-3), pp. 167–202, doi:10.1016/S0012-365X(98)00371-9.
- [4] Manosij Ghosh Dastidar & Michael Wallner (2024): *Bijections and congruences involving lattice paths and integer compositions*. arXiv:2402.17849.
- [5] Byungchan Kim, Eunmi Kim & Jeremy Lovejoy (2020): *Parity bias in partitions*. *Eur. J. Comb.* 89, p. 18, doi:10.1016/j.ejc.2020.103159. Id/No 103159.
- [6] Philippe Marchal (2003): *Constructing a sequence of random walks strongly converging to Brownian motion*. In: *Proceedings of Discrete Random Walks (DRW'03)*, *Discr. Math. Theor. Comput. Sci.*, pp. 181–190, doi:10.46298/dmtcs.3335.

Greedy Gray Codes for some Restricted Classes of Binary Words

Nathanaël Hassler

Vincent Vajnovszki

Laboratoire d'Informatique de Bourgogne
Dijon, France

nathanael.hassler@ens-rennes.fr

vincent.vajnovszki@u-bourgogne.fr

Dennis Wong

Macao Polytechnic University
Macao, China

cwong@uoguelph.ca

We investigate the existence of greedy Gray codes, based on the choice of the first element in the code, for two classes of binary words: generalized Fibonacci words and generalized Dyck words.

1 Introduction

1.1 Constrained binary words

1's run constrained binary words

Let $n, p \in \mathbb{N}$, $p \geq 2$, and $F_n(p)$ be the set of length n binary words with no p consecutive 1's. $F_n(2)$ is counted by the Fibonacci numbers f_n , and in general, $F_n(p)$ is counted by the p -order Fibonacci numbers $f_n^{(p)}$. Now let $k \in \mathbb{N}$, and $F_n(p, k)$ be the subset of words in $F_n(p)$ of weight k (i.e., with exactly k 1's). $F_n(p, k)$ is counted by the univariate p -nomial coefficient.

Prefix constrained binary words

Let $k, n, p \in \mathbb{N}$ with $(p+1)k \leq n$, and $C_n(p, k)$ be the set of length n binary words of weight k with the property that any prefix contains at least p times as many 0's as 1's. In particular:

- $C_n(0, k)$ is the set of length n binary words of weight k (combinations in binary word representation),
- $C_{2n}(1, n)$ is the set of length $2n$ Dyck words, and it is counted by the Catalan numbers $\binom{2n}{n} - \binom{2n}{n-1} = \frac{1}{n+1} \binom{2n}{n}$,
- $C_{3n}(2, n)$ is in bijection with size $3n$ ternary trees (see A001764 in [4]).

More generally, $C_{(p+1)n}(p, n)$ is counted by $\frac{1}{pn+1} \binom{(p+1)n}{n}$, known as the Pfaff–Fuss–Catalan numbers, and the cardinality of $C_n(p, k)$ is established for instance in [5, Equation (2)], using generating functions:

$$|C_n(p, k)| = \binom{n}{k} - p \binom{n}{k-1}. \quad (1)$$

1.2 Gray codes and greedy algorithms

A Gray code for a class of combinatorial objects is a list that contains each object from the class exactly once such that any two consecutive objects in the list differ only by a ‘small change’ [3]. In this paper we restrict ourselves to Gray codes for restricted classes of same length and same weight binary words (the weight of a binary word being its number of 1’s). The ‘small changes’ we consider here are homogeneous transpositions: two binary words differ by a *homogeneous* transposition if one can be obtained from the other by transposing a 1 with a 0, and there are no 1’s between the transposed bits. A Gray code is called homogeneous if consecutive words differ in a such a way. A list of words is *suffix partitioned* if words with the same suffix are consecutive in the list.

The next definition of the *greedy Gray code algorithm* is a specialisation of that introduced in [6] to particular cases of binary words, see also [7].

Definition 1.1. For a set S of same length and same weight binary words the *greedy Gray code algorithm* to obtain a Gray code list \mathcal{L} for the set S is:

1. Initialize \mathcal{L} with a particular word in S .
2. For the last word in \mathcal{L} , homogeneously transposes the leftmost possible 1 with the leftmost possible 0, such that the obtained word is in S but not in \mathcal{L} .
3. If at point 2. a new word is obtained, then append it to the list \mathcal{L} and return to point 2.

In the following we will say that the list \mathcal{L} is obtained by applying the greedy algorithm for S to α , where α is the initial word of \mathcal{L} . Depending on the choice of α , it can happen that the obtained list \mathcal{L} is not an exhaustive one for S .

2 Tail partitioned lists

The tail of a binary word is its unique suffix of the form $011 \cdots 1$, and the only words with no tail have the form $11 \cdots 1$. Note that 0 is also a tail, for the words ending by 0.

A list of binary words is *increasing (decreasing) tail partitioned* if words with tails of length ℓ appear before (after) words with tail of length $\ell + 1$, for any $\ell \geq 1$.

Definition 2.1. A list \mathcal{L} of same length binary words is *recursive tail partitioned* if it is empty, or

- it is increasing or decreasing tail partitioned, and
- for any tail t , the list obtained by: (i) considering the sublist of \mathcal{L} of words with tail t , then (ii) erasing the tail t in each word of this sublist, is in turn recursive tail partitioned.

In the following \cdot denotes the concatenation (of two words, or of each word in a list with a word) and the comma appends lists. With this notation, \mathcal{L} is a recursive tail partitioned list if it is empty or has the form

$$\mathcal{L} = \mathcal{L}_1 \cdot 01^u, \mathcal{L}_2 \cdot 01^{u+1}, \mathcal{L}_3 \cdot 01^{u+2}, \dots, \mathcal{L}_{\ell+1} \cdot 01^{u+\ell} \quad (2)$$

or the form

$$\mathcal{L} = \mathcal{L}_1 \cdot 01^{u+\ell}, \mathcal{L}_2 \cdot 01^{u+\ell-1}, \mathcal{L}_3 \cdot 01^{u+\ell-2}, \dots, \mathcal{L}_{\ell+1} \cdot 01^u \quad (3)$$

for some $u, \ell \geq 0$, and each list \mathcal{L}_i , is in turn recursive tail partitioned.

We remark that the list in (2) is not necessarily the reverse of that in (3). Clearly, a recursive tail partitioned list (r-t partitioned list for short) is a suffix partitioned list.

Theorem 2.2. *If the list \mathcal{L} is a homogeneous and suffix partitioned Gray code for a set of (same length and same weight) binary words, then \mathcal{L} is an r-t partitioned list.*

3 Main results

3.1 Fibonacci words $F_n(2, k)$

In this subsection we show that for any $\alpha \in F_n(2, k)$, by applying the greedy algorithm for $F_n(2, k)$ to α , a suffix partitioned list is obtained, and we characterize the words α such that the greedy algorithm yields an exhaustive list for $F_n(2, k)$. Moreover, for every $\alpha \in F_n(2, k)$ we characterize the last word in the obtained list.

For $n < 2k - 1$, $F_n(2, k)$ is empty and in two particular cases $F_n(2, k)$ is a singleton set.

Fact 3.1. If $k = 0$, then $F_n(2, k) = \{0^n\}$, and if $n = 2k - 1$, then $F_n(2, k) = \{1(01)^{k-1}\}$.

For $\alpha \in F_n(2, k)$ we denote by $\mathcal{F}(\alpha)$ the list obtained by applying the greedy algorithm for $F_n(2, k)$ to α .

For n, k with $n \geq 2k$, let $\alpha_{n,k}^i = 0^i 1(01)^{k-1} 0^{n-2k+1-i}$, for $0 \leq i \leq n - 2k + 1$. Furthermore, let $\gamma_{n,k} := \alpha_{n,k}^{n-2k+1} = 0^{n-2k}(01)^k$. We denote by $GenF(n, k)$ the set of words $\alpha \in F_n(2, k)$ such that $\mathcal{F}(\alpha)$ is a homogeneous Gray code for $F_n(2, k)$. Equivalently, $\alpha \in GenF(n, k)$ if and only if $\mathcal{F}(\alpha)$ contains every word of $F_n(2, k)$.

Theorem 3.2. Let $n \in \mathbb{N}^*$. Then for all k with $n \geq 2k$, we have

1. $\alpha_{n,k}^i \in GenF(n, k)$ for all $0 \leq i \leq n - 2k$, and $\mathcal{F}(\alpha_{n,k}^i)$ is a suffix partitioned list, with $\gamma_{n,k}$ as last word.
2. $\gamma_{n,k} \in GenF(n, k)$, $\mathcal{F}(\gamma_{n,k})$ is a suffix partitioned list and its last word is
 - (i) $\alpha_{n,k}^0$ if k is even,
 - (ii) $\alpha_{n,k}^{n-2k}$ if k is odd.

In the next lemma, we extend the result of 1. in Theorem 3.2 concerning the last element of $\mathcal{F}(\alpha)$ to each $\alpha \in F_n(2, k)$.

Lemma 3.3. Let $n \geq 2k$. For any $\alpha \in F_n(2, k)$ with $\alpha \neq \gamma_{n,k}$, the last word of $\mathcal{F}(\alpha)$ is $\gamma_{n,k}$.

Now we generalise the property for $\mathcal{F}(\alpha)$ to be suffix partitioned to each word of $F_n(2, n)$.

Lemma 3.4. Let $n \geq 2k - 1$. For any $\alpha \in F_n(2, k)$, $\mathcal{F}(\alpha)$ is a suffix partitioned list.

Now we are able to completely describe the set $GenF(n, k)$.

Proposition 3.5. Let $n \geq 2k - 1$. Then

$$GenF(n, k) = \{0^i 1(01)^{k-1} 0^{n-2k+1-i} \mid 0 \leq i \leq n - 2k + 1\}.$$

In particular, $|GenF(n, k)| = n - 2k + 2$.

3.2 $C_n(p, k)$

For n, p, k such that $n \geq (p + 1)k$ and $\alpha \in C_n(p, k)$, we denote by $\mathcal{D}_p(\alpha)$ the list obtained by applying the greedy algorithm for $C_n(p, k)$ to α . Let also $Gen_p(n, k)$ be the set of words $\alpha \in C_n(p, k)$ such that $\mathcal{D}_p(\alpha)$ is a homogeneous Gray code for $C_n(p, k)$. Equivalently, $\alpha \in Gen_p(n, k)$ if and only if $\mathcal{D}_p(\alpha)$ contains every word of $C_n(p, k)$. Since the situation is a bit more complicated than the one for Fibonacci words, for clarity we first investigate the case $p \in \mathbb{N}$, explaining every details of the proofs. Then we explain how the results are generalised to any $p \in \mathbb{R}$, not necessarily describing each proof.

3.2.1 $p \in \mathbb{N}$

We start by investigating the case $p \in \mathbb{N}$. We fix such a p throughout this section. For $n \geq (p+1)k$, let $\alpha_{n,k}^{i,j} := 0^{pj-i}1^{j-1}0^i1(0^p1)^{k-j}0^{n-(p+1)k}$ (resp. $\alpha_{n,k} = 1^k0^{n-k}$) for $i = 0, \dots, p-1$ and $j = 1, \dots, k$ if $p \geq 1$ (resp. if $p = 0$), and $\beta_{n,k}^i := 0^i1^k0^{n-i-k}$ for $i = pk+1, \dots, n-k$. Because it plays a special role, we will use different notation for $\beta_{n,k}^{n-k}$, so we set $\gamma_{n,k} := \beta_{n,k}^{n-k} = 0^{n-k}1^k$.

Theorem 3.6. *Let $n \geq (p+1)k$, with $k \geq 1$.*

1. *For all $0 \leq i \leq p-1$ and $1 \leq j \leq k$, we have $\alpha_{n,k}^{i,j} \in \text{Gen}_p(n,k)$ (resp. $\alpha_{n,k} \in \text{Gen}_0(n,k)$), and $\mathcal{D}_p(\alpha_{n,k}^{i,j})$ (resp. $\mathcal{D}_0(\alpha_{n,k})$) is a suffix partitioned list with $\gamma_{n,k}$ as last word.*
2. *For all j , $pk+1 \leq j \leq n-k-1$, we have $\beta_{n,k}^j \in \text{Gen}_p(n,k)$, and $\mathcal{D}_p(\beta_{n,k}^j)$ is a suffix partitioned list with $\gamma_{n,k}$ as last word.*
3. *$\gamma_{n,k} \in \text{Gen}_p(n,k)$, $\mathcal{D}_p(\gamma_{n,k})$ is a suffix partitioned list and its last word is*
 - (i) 0^p1 if $(n,k) = (p+1, 1)$,
 - (ii) $\beta_{n,k}^{n-k-1} = 0^{n-k-1}1^k0$ if k is odd and $n \neq (p+1)k$,
 - (iii) $\alpha_{n,k}^{1,k-1} = 0^{(k-1)p-1}1^{k-2}010^p1$ if $k \geq 3$ is odd and $n = (p+1)k$ (resp. $\alpha_{n,k} = 1^k$ if $p = 0$),
 - (iv) $\alpha_{n,k}^{1,k} = 0^{kp-1}1^{k-1}010^{n-(p+1)k}$ if k is even (resp. $\alpha_{n,k} = 1^k0^{n-k}$ if $p = 0$).

Lemma 3.7. *Let $n \geq (p+1)k$. For all $\alpha \in C_n(p,k)$, with $\alpha \neq \gamma_{n,k}$, $\mathcal{D}_p(\alpha)$ ends with $\gamma_{n,k}$.*

Lemma 3.8. *Let $n \geq 1$. Then for all k with $n \geq (p+1)k$, and for each $\alpha \in C_n(p,k)$, $\mathcal{D}_p(\alpha)$ is a suffix partitioned list.*

Proposition 3.9. *Let $n \geq (p+1)k$. If $p \geq 1$ then*

$$\begin{aligned} \text{Gen}_p(n,k) = \bigcup_{j=1}^k \{ & 0^{pj-i}1^{j-1}0^i1(0^p1)^{k-j}0^{n-(p+1)k} \mid 0 \leq i \leq p-1 \} \\ & \cup \{ 0^i1^k0^{n-i-k} \mid pk+1 \leq i \leq n-k \}. \end{aligned}$$

If $p = 0$ then $\text{Gen}_0(n,k) = \{ 0^i1^k0^{n-i-k} \mid 0 \leq i \leq n-k \}$. In particular,

$$|\text{Gen}_p(n,k)| = n - k + 1 - p.$$

Remark 3.10. Propositions 3.5 and 3.9 highlight the fact that the choice of the first element for the greedy algorithm is crucial. Indeed, only a few elements will produce a Gray code for $F_n(2,k)$ or $C_n(p,k)$ with this algorithm.

3.2.2 $p \in \mathbb{R}_+$

The previous results can be generalized to $C_n(p,k)$ for any $p \in \mathbb{R}_+$. In particular we have the following result:

Theorem 3.11. *Let $p \in \mathbb{R}_+$ and $n \geq (p+1)k$. Then*

$$|\text{Gen}_p(n,k)| = n - k + 1 - \lceil p \rceil.$$

3.3 Algorithmic considerations

Building on the previous results, the last part presents CAT algorithms that greedily generate Gray code for $C_n(p, k)$ and $F_n(2, k)$, which we omit in this abstract.

References

- [1] A. Bultena & F. Ruskey (1998): *An Eades-McKay algorithm for well-formed parentheses strings*. *Information Processing Letters* 68, pp. 255–259, doi:10.1016/S0020-0190(98)00171-9.
- [2] P. Eades & B. McKay (1984): *An algorithm for generating subsets of fixed size with a strong minimal change property*. *Information Processing Letters* 19, pp. 131–133, doi:10.1016/0020-0190(84)90091-7.
- [3] Torsten Mütze (2023): *Combinatorial Gray codes – an updated survey*. *Electronic Journal of Combinatorics* 30(3):DS26, doi:10.37236/11023.
- [4] N.J.A. Sloane: *The On-Line Encyclopedia of Integer Sequences*. Available at <http://oeis.org/>.
- [5] V. Vajnovszki & T. Walsh (2006): *A loop-free two-close Gray code algorithm for listing k -ary Dyck words*. *Journal of Discrete Algorithms* 4(4), pp. 633–648, doi:10.1016/j.jda.2005.07.003.
- [6] A. Williams (2013): *The greedy Gray code algorithm*. In: *Proceedings of the 13th international conference on Algorithms and Data Structures*, p. 525–536, doi:10.1007/978-3-642-40104-6_46.
- [7] D. Wong & V. Vajnovszki (2023): *Greedy Gray codes for Dyck words and ballot sequences*. In: *Computing and Combinatorics. COCOON 2023, Lecture Notes in Computer Science 14423*, Springer, pp. 29–40, doi:10.1007/978-3-031-49193-1_3.

Morphic Sequences: Complexity and Decidability

Raphael Henry

I2M

Marseille, France

raphael.henry@univ-amu.fr

In this work we recall Pansiot's result on the complexity of pure morphic sequences and we use the tools developed by Devyatov for morphic sequences to prove the decidability of the complexity class of pure morphic sequences.

1 Introduction

In symbolic dynamics, a natural way to generate right infinite words (indexed by $\mathbb{N} = \{0, 1, 2, \dots\}$) on a finite alphabet A is to iterate a morphism $\varphi : A^* \rightarrow A^*$ on a letter a , a process which converges to a fixed point of φ . We call such a word a **pure morphic sequence** and we denote it by $\varphi^\infty(a)$. More generally, applying a coding ψ , that is a letter-to-letter morphism, to a pure morphic word gives a **morphic sequence** denoted by $\psi(\varphi^\infty(a))$.

A major tool of symbolic dynamics is the **factor complexity**: the function $P_\alpha : \mathbb{N} \rightarrow \mathbb{N}$ counting the number of rows (factors) of length n appearing in the sequence α . An important result linking complexity and the structure of sequences is the following:

Theorem 1.1 (Morse-Hedlund, 1938). *A sequence α is ultimately periodic if and only if $P_\alpha(n) \leq n$ for some $n \in \mathbb{N}^*$ if and only if $P_\alpha(n)$ is bounded.*

In this work we study the characterization of the complexity of pure morphic sequences. For example, with D0L-systems, Ehrenfeucht, Lee et Rozenberg showed in 1975 that the complexity of pure morphic sequences is $\mathcal{O}(n^2)$. Other lower and upper bounds were obtained in particular cases until Pansiot gave the complete classification in [5] using criteria on the morphism:

Theorem 1.2 (J.J. Pansiot, 1984). *The complexity of pure morphic sequences belongs to one of the five classes:*

$$\Theta(1), \Theta(n), \Theta(n \log \log n), \Theta(n \log n), \Theta(n^2).$$

Applying a coding will either permute letters or merge some of them, which can only decrease the complexity. Doing so, new complexity classes appear:

Proposition 1.3 (J.J. Pansiot, 1985). *For every $k \in \mathbb{N}^*$, there exists a morphic sequence α such that $P_\alpha(n) = \Theta(n^{1+1/k})$.*

This result is stated in [6], and the example of a pure morphic sequence of complexity $\Theta(n^{1+1/k})$ is detailed in [1]:

- $A = \{a, b_0, b_1, \dots, b_k\}$
- $\varphi(a) = ab_k, \varphi(b_0) = b_0$ and $\varphi(b_i) = b_i b_{i-1}$ for $i \in \llbracket 1, k \rrbracket$
- $\psi(a) = 0, \psi(b_i) = 0$ for $i \in \llbracket 0, k-1 \rrbracket$ and $\psi(b_k) = 1$

In [2], Devyatov shows that they are the only classes between $\Theta(n \log n)$ et $\Theta(n^2)$:

Theorem 1.4 (R. Devyatov, 2015). *The complexity of morphic sequences is either:*

$$\Theta(n^{1+1/k}) \text{ for some } k \in \mathbb{N}^* \text{ or } \mathcal{O}(n \log n).$$

2 Result

In [6], Pansiot mentions the following decidability problem:

PMClass: **Input:** A pure morphic sequence $\alpha = \varphi^\infty(a)$
Question: What is the complexity class of α ?

Theorem 1.2 states that there are five possible answers and its proof exhibits criteria for each complexity class, which we will formulate in an algorithm. By deciding every criterion, we prove the following result:

Theorem 2.1. *PMClass is decidable.*

To achieve that, we use the detailed proof of Theorem 1.2 in [1] and a decidability result from Pansiot [7] and Harju-Linna [4], and we adapt some parts of Devyatov's proof.

3 Sketch of the proof

3.1 Growth of morphism

If $a \in A$ is a letter, the **growth rate** of a is the function associated to the asymptotic behaviour of $|\varphi^k(a)|$ when k tends to ∞ . The following theorem stated in [8] gives its precise form:

Theorem 3.1 (A. Salomaa, M. Soittola). *For each morphism $\varphi : A^* \rightarrow A^*$ and each letter $a \in A$, there exist $(\beta, \alpha) \in (\mathbb{R}_{\geq 1} \times \mathbb{N}) \cup \{(0, 0)\}$ such that*

$$|\varphi^k(a)| = \Theta(k^\alpha \beta^k).$$

We say a letter is **bounded** if its growth rate is bounded ($(\beta_a, \alpha_a) \in \{(0, 0), (1, 0)\}$), and **growing** in the other case. We denote by B the set of bounded letters and C the set of growing letters. If every letter is growing, φ is said to be **growing**. The case $(\beta_a, \alpha_a) = (0, 0)$ means that φ erases the letter a ($\varphi(a) = \varepsilon$ the empty word).

We say φ is **quasi-uniform** if every letter has the same rate of the form β^k with $\beta > 1$. We say φ is **polynomially divergent** if every letter a has a rate of the form $k^{\alpha_a} \beta^k$ with $\beta > 1$, and at least one of the α_a is not 0. We say φ is **exponentially divergent** if there are two letters a and b of rate $k^{\alpha_a} \beta_a^k$ et $k^{\alpha_b} \beta_b^k$ with $1 < \beta_a < \beta_b$ and $\beta_c > 1$ for all $c \in A$.

These three classes of morphisms are mutually exclusive, and a morphism is growing if and only if it belongs to one of them.

3.2 Pansiot criteria

Given a finite alphabet A , a morphism $\varphi : A^* \rightarrow A^*$ and a pure morphic sequence, the proof of Theorem 1.2 gives the criteria to determine its complexity class.

In particular, the case where φ is not growing and the factors of α in B^* have bounded length boils down to computing the complexity of another pure morphic sequence:

Proposition 3.2. *If φ is not growing and the factors of α in B^* have bounded length, one can explicitly compute an alphabet Σ , a growing morphism $\sigma : \Sigma^* \rightarrow \Sigma^*$, a letter $b \in \Sigma$ and a non-erasing morphism $\psi : \Sigma^* \rightarrow A^*$ such that*

$$\alpha = \psi(\sigma^\infty(b)).$$

Moreover α and $\sigma^\infty(b)$ are in the same complexity class.

We formulate the classification with the following algorithm:

```

PMClass( $\alpha = \varphi^\infty(a)$ ):
  if  $\alpha$  is eventually periodic:
    return " $\Theta(1)$ "
  if  $\varphi$  is growing:
    if  $\varphi$  is quasi-uniform:
      return " $\Theta(n)$ "
    if  $\varphi$  is polynomially divergent:
      return " $\Theta(n \log \log n)$ "
    if  $\varphi$  is exponentially divergent:
      return " $\Theta(n \log n)$ "
  else:
    if the factors of  $\alpha$  in  $B^*$  have bounded length:
      compute  $\Sigma, \sigma, \psi$  et  $b$  such that  $\alpha = \psi(\sigma^\infty(b))$ 
      return PMClass( $\sigma^\infty(b)$ )
    else:
      return " $\Theta(n^2)$ "

```

When φ is not growing and the factors of α in B^* have bounded length, the algorithm is recursive but the new morphism σ is growing so there is only one more iteration. Each complexity class is non-empty, here are examples for each one:

- $\Theta(1)$: with $\varphi : a \mapsto ab, b \mapsto c, c \mapsto b, \varphi^\infty(a) = abcbcbcbcb\dots$ is eventually periodic.
- $\Theta(n)$: with the Thue-Morse morphism $\varphi : a \mapsto ab, b \mapsto ba, |\varphi^k(a)| = |\varphi^k(b)| = 2^k$, φ is quasi-uniform and $\varphi^\infty(a) = abbabaabbaababba\dots$
- $\Theta(n \log \log n)$: with $\varphi : a \mapsto aba, b \mapsto bb, |\varphi^k(b)| = 2^k$ and $|\varphi^k(a)| = k2^{k-1} + 2^k$, φ is polynomially divergent and $\varphi^\infty(a) = ababbababbbbabababba\dots$
- $\Theta(n \log n)$: with $\varphi : a \mapsto abc, b \mapsto bb, c \mapsto ccc, |\varphi^k(b)| = 2^k, |\varphi^k(c)| = 3^k$ and $|\varphi^k(a)| = 2^k + (3^k - 1)/2$, φ is exponentially divergent and $\varphi^\infty(a) = abcbbcccbbbbcccccccc\dots$
- $\Theta(n^2)$: with $\varphi : a \mapsto ab, b \mapsto bc, c \mapsto c, B = \{c\}, \varphi^\infty(a) = abc^1bc^2bc^3\dots$ and for all $i \ c^i \sqsubset \alpha$.
- $\Theta(n)$ in two iterations: with $\varphi : a \mapsto acb, b \mapsto bca, c \mapsto c, B = \{c\}$ but the only factors of α in B^* are ε and c . Actually $\varphi^\infty(a) = \psi(\sigma^\infty(a))$ with σ being the Thue-Morse morphism on $\{a, b\}$ and $\psi : a \mapsto ac, b \mapsto bc$.

4 Decidability

We prove that each condition of the algorithm is decidable.

4.1 α is eventually periodic

Pansiot and Harju-Linna proved simultaneously in [7] and [4] that the eventual periodicity can be reduced to properties on factors and prove the decidability:

Theorem 4.1. *The eventual periodicity of pure morphic sequences is decidable.*

4.2 φ is growing

Deciding if φ is growing, quasi-uniform, polynomially or exponentially divergent can be done by algorithmically computing and comparing eigenvalues of integer matrices.

4.3 φ is not growing

In order to decide if the factors of α in B^* have bounded length, we prove that it is equivalent to a constructive property on the images of letters. Let us remark that this property appears with a non-constructive form in [5] (proof of Theorem 4.1) and in [3] (Lemma 3.15). To achieve that we use the notion of k -blocks developed by Devyatov in [2]:

An occurrence of α is a factor associated to the position of its letters in α , denoted $\alpha_{i\dots j}$. A **1-block** is an occurrence of α in B^* surrounded by two growing letters that we call the **left border** and the **right border**. Then α can be split into an alternation of (possibly empty) 1-blocks and growing letters.

If u is a 1-block, then $\varphi(u)$ is an occurrence of α containing only bounded letters so it is contained in a unique 1-block that we call the **descendant** of u and we denote it by $Dc_1(u)$.

Also if there exists a 1-block v such that $Dc_1(v) = u$, then v is unique, we call it the **ancestor** of u and we denote it by $Dc_1^{-1}(u)$. If a 1-block has no ancestor, which is equivalent to the fact that the 1-block and its borders are contained in the image of a letter under φ , we say it is an **origin**.

An **evolution** of 1-blocks is a sequence \mathcal{E} of 1-blocks such that \mathcal{E}_0 is an origin and, for every integer l , $\mathcal{E}_l = Dc_1^l(\mathcal{E}_0)$. In particular every 1-block belongs to an evolution of 1-blocks.

For every word u containing a growing letter, $LB(u)$ (resp. $RB(u)$) denotes the longest prefix (resp. suffix) of u in B^* , and $LC(u)$ (resp. $RC(u)$) denotes the first (resp. last) growing letter in u . With these notations we get a first idea of the structure of 1-blocks.

Lemma 4.2. *Let $\alpha = \varphi^\infty(a)$ a pure morphic sequence, \mathcal{E}_l a 1-block of index l in its evolution \mathcal{E} and α_i, α_j the borders of \mathcal{E}_0 . Then*

$$\mathcal{E}_l = RB(\varphi^l(\alpha_i)) \varphi^l(\mathcal{E}_0) LB(\varphi^l(\alpha_j)).$$

For every growing letter c , we also define the following objects:

$$\begin{aligned} LE(c) &= \varphi(LB(\varphi(c))) & RE(c) &= \varphi(RB(\varphi(c))) \\ LK(c) &= LB(\varphi(LC(\varphi(c)))) & RK(c) &= RB(\varphi(RC(\varphi(c)))) \\ LP(c) &= \varphi(LK(c)) & RP(c) &= \varphi(RK(c)) \end{aligned}$$

In order to refine the structure, we replace φ by a large enough power of itself, which does not modify α nor the properties of the morphism, so that the morphism is **strongly 1-periodic**. We must note that this power can be bounded using the size of A , which makes this process effective.

Lemma 4.3. *Let φ a strongly 1-periodic morphism. Then for every growing letter c and for every $l \geq 2$*

$$\begin{aligned} LB(\varphi^l(c)) &= LE(c) LP(c)^{l-2} LK(c) \\ RB(\varphi^l(c)) &= RK(c) RP(c)^{l-2} RE(c) \end{aligned}$$

These two lemmas lead us to state an equivalent condition which is clearly decidable:

Proposition 4.4. *Let φ a strongly 1-periodic morphism and $\alpha = \varphi^\infty(a)$ a non-eventually-periodic pure morphic sequence. Then the factors of α in B^* have bounded length if and only if*

$$\text{for every } c \in C, LP(c) = RP(c) = \varepsilon.$$

References

- [1] J. Cassaigne & F. Nicolas (2010): *Factor complexity*, p. 163–247. Encyclopedia of Mathematics and its Applications, Cambridge University Press, Cambridge, doi:10.1017/CB09780511777653.005.
- [2] R. Devyatov (2015): *On Subword Complexity of Morphic Sequences*. Available at <https://arxiv.org/abs/1502.02310v1>.
- [3] F. Durand (2011): *Decidability of the HD0L ultimate periodicity problem*. *RAIRO - Theoretical Informatics and Applications* 47, doi:10.1051/ita/2013035.
- [4] T. Harju & M. Linna (1986): *On the periodicity of morphisms on free monoids*. *RAIRO Theor. Informatics Appl.* 20, pp. 47–54, doi:10.1051/ita/1986200100471.
- [5] Pansiot J.J. (1984): *Complexité des facteurs des mots infinis engendrés par morphismes itéré*. *RAIRO Theor. Informatics Appl.*, doi:10.1007/3-540-13345-3_34.
- [6] J.J Pansiot (1985): *Subword complexities and iteration*. *Bulletin of the EATCS* 26.
- [7] J.J Pansiot (1986): *Decidability of Periodicity for Infinite Words*. *RAIRO Theor. Informatics Appl.* 20, doi:10.1051/ita/1986200100431.
- [8] A. Salomaa & M. Soittola (1978): *Automata: Theoretic Aspects of Formal Power Series*. Springer-Verlag, Berlin, Heidelberg, doi:10.1007/978-1-4612-6264-0.

Detecting Isohedral Polyforms with a SAT Solver

Craig S. Kaplan

School of Computer Science
University of Waterloo
csk@uwaterloo.ca

I show how to express the question of whether a polyform tiles the plane isohedrally as a Boolean formula that can be tested using a SAT solver. This approach is adaptable to a wide range of polyforms, requires no special-case code for different isohedral tiling types, and integrates seamlessly with existing software for computing Heesch numbers of polyforms.

1 Introduction

The study of algorithms for computing tiling-theoretic properties of shapes is a rich and fascinating branch of computational geometry. Implementations of these algorithms can also serve as useful tools in the experimental side of tiling theory, as part of the search for new shapes with interesting properties. For example, Myers systematically computed isohedral numbers (the minimum number of transitivity classes in any tiling by a given shape) for many simple polyforms [7]. Building on Myers’s work, I computed Heesch numbers (the maximum number of times that a non-tiling shape can be surrounded by layers of copies of itself) for simple polyforms [4]. Our tools did not contribute to Smith’s initial discovery of the “hat” aperiodic monotile, but they played a central role in our subsequent analysis of the hat and our proof (with Goodman-Strauss) of its aperiodicity [9].

An *isohedral tiling* is a tiling by congruent copies of some *prototile* T , such that for any two tiles T_1 and T_2 there exists a symmetry of the tiling mapping T_1 to T_2 . Isohedral tilings are some of the simplest periodic tilings, in that all tiles belong to a single transitivity class relative to the symmetries of the tiling. A complete theory of isohedral tilings, including their classification into 81 tiling types with unmarked tiles, was worked out by Grünbaum and Shephard [3, Chapter 6].

Given a simple shape such as a polyform, does it admit any isohedral tilings? This question offers interesting opportunities for the development of new algorithms. It is also of practical interest as part of any software for computing the tiling-theoretic properties of shapes. Myers’s software [7] can detect isohedral prototiles quickly, but formal questions of computational complexity are more or less peripheral to his work. The current state of the art, at least for the special case of polyominoes, is the quasilinear-time algorithm by Langerman and Winslow [5].

In this paper I present a new technique for checking whether a polyform tiles isohedrally. The algorithm is based on expressing the question as a Boolean formula that can be checked by a SAT solver, and was motivated by my desire to integrate such a test into my existing SAT-based framework for computing Heesch numbers [4]. I will explain the mathematical basis for this approach (Section 2), followed by its expression in Boolean logic (Section 3), and then conclude with a few final observations (Section 4).

2 Identifying prototiles based on surrounds

In order to determine whether a shape admits any isohedral tilings of the plane, it suffices to examine the ways that the shape can be surrounded by copies of itself. That is, if there exists a surround with a particular structure that will be explained here, then the shape is guaranteed to tile isohedrally.

Let T be a shape, which in full generality can be any topological disk, but which for my purposes is typically a polygon. Without loss of generality, I assume here that T is asymmetric. (A symmetric shape can always be decorated with an asymmetric marking, with the meaning of congruence expanded to preserve markings.)

A *patch* is a finite collection of congruent copies of T , with pairwise disjoint interiors, whose union is a topological disk. In particular, if exactly one copy of T lies in the interior of the patch, then we refer to the patch as a *1-patch*, to the interior tile as the patch's *centre*, and to the remaining tiles as a *surround* of T .

The fact that every two tiles in an isohedral tiling are related by a symmetry of the tiling implies that every tile is the centre of a congruent 1-patch, or more loosely that tiles have congruent surrounds. Grünbaum and Shephard use this fact to develop a complete enumeration of isohedral tiling types, based on an “incidence symbol” that expresses a prototile's relationships to its neighbours [3]. In fact, the converse holds as well: Dolbilin and Schattschneider showed that if the tiles in a tiling have congruent surrounds, then the tiling must be isohedral [2].

Let $\mathcal{S} = \{T_1, \dots, T_n\}$ be a surround of a shape T . The surround is made up of congruent copies of T , meaning that each $T_i = g_i(T)$ for some rigid motion g_i . Fix one shape T_i in the surround, and construct $\mathcal{S}_i = \{g_i \circ g_j(T)\}_{j=1}^n$, a congruent copy of \mathcal{S} placed around T_i . I call T_i *extendable* if this transformed surround does not “conflict” with T_i 's neighbours in the original 1-patch centred at T . More precisely, T_i is extendable if for every $A \in \{T, T_1, \dots, T_n\}$ and every $B \in \mathcal{S}_i$, either $A = B$ or A and B have disjoint interiors.

Suppose that T has a surround in which every T_i is extendable. The transformed surrounds \mathcal{S}_i must all be compatible with the 1-patch around T and with each other, meaning that their union will surround \mathcal{S} with a second layer of tiles. In this manner we can continue outward layer by layer, each time completing the surrounds of the tiles along the boundary of the growing patch. (This construction is similar to one used by Grünbaum and Shephard [3, Theorem 6.1.1].) In the limit we obtain a tiling of the plane in which every tile has a congruent surround, which must therefore be isohedral by The Local Theorem of Dolbilin and Schattschneider [2]. I summarize this argument with a proposition.

Proposition 1 *A shape T admits an isohedral tiling if and only if T has a surround $\mathcal{S} = \{T_1, \dots, T_n\}$ in which every T_i is extendable (in which case every tile in the tiling is surrounded by a congruent copy of \mathcal{S}).*

3 SAT formulation

In previous work I showed how to use a SAT solver to compute Heesch numbers of simple polyforms [4]. My software constructs a sequence of Boolean formulas equivalent to the questions “Can T be surrounded at least once?”, “Can T be surrounded at least twice?”, and so on, and passes them to a SAT solver. It halts as soon as one of these questions is false (or after a predetermined maximum number of levels, to avoid looping forever when given a shape that tiles).

Here I show that it is possible to incorporate the mathematical ideas of the previous section into my Heesch number computation, by interposing the question “Can T tile isohedrally?” immediately

after “Can T be surrounded at least once?”. Indeed, the new question is a simple restriction of the surroundability formula already being used, taking the form “Can T be surrounded at least once, in a way that witnesses its ability to tile isohedrally?”.

Let \mathcal{T} be a tiling of the plane. A *poly- \mathcal{T} -tile* is a shape created by gluing together a finite connected set of tiles from \mathcal{T} . Informally, I refer to a poly- \mathcal{T} -tile as a “polyform”, to \mathcal{T} as “the grid”, and to the tiles of \mathcal{T} as “cells”. In any patch or tiling by a polyform, I will also require that every tile be a union of cells from the grid; that is, every tile must be “aligned” to the grid.

Let T be a poly- \mathcal{T} -tile. Define the *halo* of T to be all grid cells not in T that are neighbours of cells in T . Compute the set $\{T_1, \dots, T_n\}$ of all transformed copies of T that can be neighbours of T in a surround. Each T_i will have the form $g_i(T)$ for a rigid motion g_i . Any legal surround must be a subset of the T_i that collectively occupy every halo cell without overlapping each other. We can express these criteria using a Boolean formula, a simplified version of the one I used for Heesch number computation. Abusing notation slightly, create Boolean variables T_1, \dots, T_n for each potential member of the surround. Now construct a formula with the following clauses:

- For every cell in the halo, a conjunction of all the T_i that use that cell (every cell in the halo must be occupied);
- For every pair T_i and T_j that overlap in one or more cells, a clause of the form $(\neg T_i \vee \neg T_j)$ (overlapping tiles are mutually exclusive).

If a satisfying assignment is found for this formula, then a candidate surround will correspond to the subset of variables set to true. It is possible, however, for the resulting set of tiles to enclose holes; if a hole is detected, then a clause is added to suppress this solution and the SAT solver is restarted. This process iterates until either a simply connected solution is found, or no more candidate surrounds remain.

If T is surroundable, we can check whether it tiles isohedrally before trying to surround it with more layers. I do so by augmenting the formula above with new clauses. Let $T_i = g_i(T)$ and $T_j = g_j(T)$ be two neighbours of T that are also themselves neighbours. If T_i and T_j are used together in a surround \mathcal{S} , then they must both be extendable by that surround. Note that $g_i(T_j) = g_i \circ g_j(T)$ will be one of the shapes in \mathcal{S}_i , the copy of \mathcal{S} surrounding T_i , and must therefore avoid conflicts with the shapes in \mathcal{S} . We can enforce this condition by finding the member $T_k = g_i \circ g_j(T)$, if it exists, and adding a clause of the form $(\neg T_i \vee \neg T_j \vee T_k)$ (if T_i and T_j are both used in a surround, then T_k must be used too). By symmetry, we perform the same steps for $g_j \circ g_i$.

We can add clauses to this formula that further restrict the space of possible solutions the SAT solver must explore, potentially improving performance. Suppose $T_i = g_i(T)$ is part of an isohedral surround, and g_i is not an involution. Then because T is a neighbour of $g_i(T)$, it follows that $g_i^{-1}(T)$ is a neighbour of T , meaning that it must also appear in the surround. We therefore find $T_k = g_i^{-1}(T)$ and add a clause of the form $(\neg T_i \vee T_k)$, which forces T_k to be used if T_i is. Similarly, in the joint cases above we also add clauses for $g_i \circ g_j^{-1}$ and $g_j \circ g_i^{-1}$, if those transformations correspond to neighbours of T .

This augmented formula has a satisfying assignment if and only if it corresponds to a surround of T for which every T_i in the surround is extendable, or in other words, if and only if T tiles the plane isohedrally.

4 Discussion

I implemented the augmented Boolean formula described above within the framework of my existing software for computing Heesch numbers of polyforms [4]. In my implementation, transformed copies of

a polyform T are represented via their affine transformation matrices (and not their boundaries or cells). A matrix effectively also serves as an asymmetric marker, thereby preventing any issues from arising with symmetric shapes.

As a simple validation, my software produces counts of isohedral polyforms that agree with the figures tabulated by Myers [7], up to the size limits I tested (12-ominoes, 12-hexes, 13-diamonds, and 12-kites).

When resigning oneself to the black box of a SAT solver, questions of asymptotic complexity become largely moot. Therefore, a theoretical comparison with, say, the quasilinear-time algorithm of Langerman and Winslow [5] is not particularly meaningful. My approach is slower than what would be possible with an efficient implementation of their algorithm, and is certainly slower than Myers's lightning-fast hand-optimized C code. In the context of my software, the extra time required for checking isohedral tiling as part of computing Heesch numbers is minimal. Furthermore, this approach is remarkably convenient—the original program for computing Heesch numbers required a few thousand lines of C++ code, and fewer than 100 lines were added for this enhancement. It is also quite general: it adapts seamlessly to arbitrary polyform grids, and does not require any special-purpose code for different isohedral tiling types (in fact, it uses the definition of isohedral tiling directly, and does not rely on any information about tiling types at all).

My enhanced implementation still cannot resolve the tiling-theoretic status of every polyform. In particular, it is unable to compute the isohedral number of any k -isohedral polyform (which admits only tilings containing at least k transitivity classes of tile) for $k \geq 2$. It would be interesting to explore further methods based on discrete optimization that can expand to cover these more complex, but equally important shapes. And of course, no software can currently detect aperiodic monotiles, for which no general procedures are known.

Acknowledgements

Thanks to Joseph Myers and Doris Schattschneider for helpful feedback during the course of this work and the preparation of this paper.

References

- [1] Bojan Bašić (2021): *A figure with Heesch number 6: pushing a two-decade-old boundary*. *Math. Intelligencer* 43(3), pp. 50–53, doi:10.1007/s00283-020-10034-w.
- [2] Nikolai Dolbilin & Doris Schattschneider (1998): *The Local Theorem for Tilings*. In Jiří Patera, editor: *Quasicrystals and discrete geometry*, 10, American Mathematical Soc., pp. 193–199, doi:10.1090/fim/010/06.
- [3] Branko Grünbaum & G.C. Shephard (2016): *Tilings and Patterns*, second edition. Dover.
- [4] Craig S. Kaplan (2022): *Heesch numbers of unmarked polyforms*. *Contributions to Discrete Mathematics* 17(2), pp. 150–171, doi:10.55016/ojs/cdm.v17i2.72886.
- [5] Stefan Langerman & Andrew Winslow (2016): *A Quasilinear-Time Algorithm for Tiling the Plane Isohedrally with a Polyomino*. In: *32nd International Symposium on Computational Geometry (SoCG 2016)*, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, pp. 50:1–50:15, doi:10.4230/LIPIcs.SocG.2016.50.
- [6] Stefan Langerman & Andrew Winslow (2016): *A Quasilinear-Time Algorithm for Tiling the Plane Isohedrally with a Polyomino*. In Sándor P. Fekete & Anna Lubiw, editors: *32nd International Symposium on*

- Computational Geometry (SoCG 2016)*, LIPIcs 51, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, pp. 50:1–50:15, doi:10.4230/LIPIcs.SoCG.2016.50.
- [7] Joseph Myers (2000–2024): *Polyform tiling*. Available at <https://www.polyomino.org.uk/mathematics/polyform-tiling/>. Accessed: May 15th, 2024.
- [8] Michael Rao (2017): *Exhaustive search of convex pentagons which tile the plane*, doi:10.48550/arXiv.1708.00274.
- [9] David Smith, Joseph Samuel Myers, Craig S. Kaplan & Chaim Goodman-Strauss (2023): *An aperiodic monotile*, doi:10.48550/arXiv.2303.10798.
- [10] Mate Soos, Karsten Nohl & Claude Castelluccia (2009): *Extending SAT Solvers to Cryptographic Problems*. In Oliver Kullmann, editor: *Theory and Applications of Satisfiability Testing - SAT 2009, 12th International Conference, SAT 2009, Swansea, UK, June 30 - July 3, 2009. Proceedings, Lecture Notes in Computer Science 5584*, Springer, pp. 244–257, doi:10.1007/978-3-642-02777-2_24.

A Symmetry Property of Christoffel Words

Yan Lanciault

LACIM, Université du Québec à Montréal,
Montréal, Québec
lanciault.yan@courrier.uqam.ca

Christophe Reutenauer*

LACIM, Université du Québec à Montréal,
Montréal, Québec
reutenauer.christophe@uqam.ca

Motivated by the theory of trapezoidal words, whose sequences of cardinality of factors by length are symmetric, we introduce a bivariate variant of this symmetry. We show that this symmetry characterizes Christoffel words, and prove other related results.

1 Introduction

Trapezoidal words were considered by Aldo de Luca in [10]; for such a word, w say, of length n , the graph of the discrete function $\{0, 1, \dots, n\} \rightarrow \mathbb{N}$, giving the number of factors of length k of w is an isosceles trapezoid, with successive values $1, 2, \dots, J, J + 1, \dots, J + 1, J, \dots, 2, 1$. He showed that Sturmian words are trapezoidal, but the converse does not necessarily hold. The terminology “trapezoidal” was introduced by Flavio d’Alessandro in [7], who studied these words, giving in particular a condition for which a trapezoidal word is Sturmian. In [4], Michelangelo Bucci, Alessandro De Luca and Gabriele Fici gave many equivalent conditions for a word to be trapezoidal; one of them is that the number of factors of length k is at most $k + 1$ (also see the work of Florence Levé and Patrice Séébold [9], and that of Mira-Cristiana Anisiu and Julien Cassaigne [1]). Remind that a factor of a word is a contiguous subword.

A remarkable property of trapezoidal words is, as mentioned above, that the sequence of the lengths of the factors of these words, from length 0 to length n , is symmetric. We may call such a word *factor-symmetric*.

In the present work, we present a generalization of this symmetry property. Let w be a word over the alphabet $\{a, b\}$, with p occurrences of the letter a and q occurrences of the letter b ; in other words, the Parikh image of w is (p, q) . We say that w is *strongly factor-symmetric* if for any i, j , w has as many distinct factors with Parikh image (i, j) as distinct factors of Parikh image $(p - i, q - j)$. Note that in that case, the notion of symmetry does not necessarily mean invariant under reversal.

We show that each Christoffel word is strongly factor-symmetric (Theorem 3.1). Conversely, each finite primitive Sturmian word which is strongly factor-symmetric is a Christoffel word (Theorem 3.2). Note that $aabb$ is strongly factor-symmetric, so that the hypothesis “Sturmian” is not superfluous.

These results are interesting, in part because one obtain a characterization of Christoffel words among all Sturmian words. Indeed, in the literature there exist many characterizations of conjugate of Christoffel words ([6, 11, 3, 13, 12, 14]), which do not distinguish between Christoffel words and their conjugates. However, another notable characterization of Christoffel words is that a Sturmian word is a Christoffel word if and only if it is a Lyndon word [2], if and only if it is unbordered [5] (see also [8]).

Concerning nonprimitive words, we show that if w is a nontrivial power of a primitive word u , then w is strongly factor-symmetric if and only if u is a Christoffel word (Theorem 3.3). The hypothesis “Sturmian” is not necessary here. In particular, $(aabb)^2$ is not strongly factor-symmetric.

*Christophe Reutenauer was partially supported by NSERC

As a byproduct, we obtain that, with the notation of the previous paragraph, that w is factor-symmetric if and only if u is the conjugate of some Christoffel word (Theorem 4.3).

Concerning the strong factor symmetry of a Christoffel word w , we give an explicit bijection between the factors of w of Parikh image (i, j) and those of Parikh image $(p - i, q - j)$ (Theorem 4.1); it relies on the notion of *attractor* and *circular attractor* [12]. Moreover, the support of the function of pairs of integers that counts the numbers of factors of w for each Parikh image, which is a subset of the discrete plane, is the set of integer points on the two paths defined by w and its reversal \bar{w} , and between them (Theorem 4.2): see, for example (1) and Figure 1.

This work was partially supported by NSERC, Canada.

2 Christoffel words and Sturmian words

Among several equivalent definitions of Christoffel words, we choose the following: a Christoffel word on the alphabet $\{a, b\}$ is either a or b , or a word of the form amb or bma , such that m is a palindrome, and w is a product of two palindromes. For other characterizations, see for example the book of the second author [13]. Christoffel words are primitive, that is, are not equal to a nontrivial power of another word.

It is known that the factorization into two palindromes is unique, and it is called the *palindromic factorization*.

Given a word w , we define the function $\delta_w : \mathbb{N}^2 \rightarrow \mathbb{N}$ by $\delta_w(i, j) =$ the number of factors of w whose Parikh image is (i, j) . We say that a word w of Parikh image (p, q) is *strongly factor-symmetric* if for any i, j , $\delta_w(i, j) = \delta_w(p - i, q - j)$. For example, the distinct factors of the Christoffel word $aabab$ are $1, a, b, aa, ab, ba, aab, aba, bab, aaba, abab, aabab$ so that δ_w is represented by the array whose i, j -coordinate is $\delta_w(i, j)$ (coordinates are as in the Cartesian plane, and this array is embedded in the plane):

$$\begin{array}{cccc} 0 & 1 & 1 & 1 \\ 1 & 2 & 2 & 1 \\ 1 & 1 & 1 & 0 \end{array} \quad (1)$$

This array has a central symmetry, which means that w is strongly factor-symmetric. We call this array the *factor array* of w .

A word w is called *factor-symmetric* if the sequence of length of factors, which turns out to be $\sum_{i+j=k} \delta_w(i, j)$, $k = 0, \dots, |w|$ is symmetric; in other words, w has as many factors of length i as factors of length $n - i$, for all i , with $n = |w|$. Clearly, a strongly factor-symmetric word is factor-symmetric.

Trapezoidal words are factor-symmetric words ([10] Proposition 4.7, [4] Definition 2.5); and conversely, each factor-symmetric word w is trapezoidal: indeed, if $|w| = n$, then w has $n - i + 1$ occurrences of factors of length i , so that it has at most $n - i + 1$ such factors; but the factor symmetry implies that it has at most $n - i + 1$ factors of length $n - i$, and hence it is trapezoidal by the cited proposition.

3 Main results

Theorem 3.1. *Each Christoffel word is strongly factor-symmetric.*

We have a converse. Note that a Sturmian word is a factor of a Christoffel word.

Theorem 3.2. *If the support of δ_w is symmetric (and in particular, if w is strongly factor-symmetric) and if w is primitive and Sturmian, then w is a Christoffel word.*

Note that the factor array of the word $aabb$ is

$$\begin{matrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{matrix}$$

which has a central symmetry, so that $aabb$ is strongly factor-symmetric; this word is not a Christoffel word, but is not Sturmian either, since aa and bb cannot be both factors of a Sturmian word.

Theorem 3.3. *Let $w = u^k$, u primitive, $k \geq 2$. Then w is strongly factor-symmetric if and only if u is a Christoffel word.*

Here, the hypothesis “factor-symmetric” suffices for the “only if” part. And the hypothesis “Sturmian” is no more necessary.

4 Byproducts

An *attractor* of a word $w = w_1 \cdots w_n$, with w_i letters of the alphabet, is a subset K of $\{1, \dots, n\}$ such that every factors of w has an occurrence that meets one of the letters indexed by one of the numbers in K . A *circular attractor* is defined similarly, but with the notion of circular factors, that is factors of a conjugate of w . Using these concepts, we have a bijection that explains Theorem 3.1.

Theorem 4.1. *Let $w = uv$ be a Christoffel word of length n with its palindromic factorization. Suppose $k, 0 \leq k \leq n$. Consider all factors of length k of w that intersect the cut of the factorization, and order them from left to right: f_1, f_2, \dots, f_r . Consider all factors of length $n - k$ of w that intersect this cut, and order them from right to left: g_1, g_2, \dots, g_s . Then $r = s$, the words f_i are distinct, the words g_i are distinct, and the mapping $f_i \mapsto g_i$ is a bijection from the set of factors of length k of w to the set of factors of length $n - k$ of w , which complements the Parikh image $\gamma(w)$ of w ; that is: $\gamma(f_i) + \gamma(g_i) = \gamma(w)$.*

An example: let $w = aababab$, $u \cdot v = aa \cdot babab$, $k = 4$, $f_1 = aaba, f_2 = abab, f_3 = baba$, $g_1 = bab, g_2 = aba, g_3 = aab$.

In the following, with each word on the alphabet $\{a, b\}$, we associate the path in the discrete plane starting from the origin, where a represents an horizontal step towards East, and b a vertical step towards North.

Theorem 4.2. *Let w be a lower Christoffel word, \tilde{w} the corresponding upper Christoffel word, and S_w the set of integer points on the paths corresponding to w and \tilde{w} . Then S_w is the support of the function δ_w .*

See for example Figure 1.

Theorem 4.3. *Let $w = u^k$, u primitive, $k \geq 2$. Then w is factor-symmetric if and only if u is the conjugate of some Christoffel word.*

Open question: which primitive trapezoidal words are strongly factor-symmetric? We know that if a word is primitive, Sturmian, and strongly factor-symmetric, it must be a Christoffel word. Hence the question is really: which primitive trapezoidal words, that are not Sturmian, are strongly factor-symmetric? An example is the word $aabb$. The work of [7] might help.

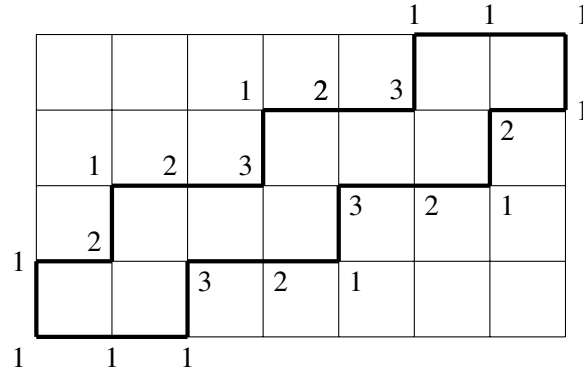


Figure 1: Paths of lower and upper Christoffel words $w = aabaabaabab$ and \tilde{w} and the function δ_w

5 Sketch of proofs

Proving Theorem 3.1 amounts to proving that the bivariate (commutative) polynomial $\sum_{i,j} \delta_w(i, j) a^i b^j \in \mathbb{N}[a, b]$ is reciprocal, with an appropriate (but evident) definition of “reciprocal”. One shows that this property is preserved by product. Then one shows, using the notion of attractor and circular attractor [12] that the factors of w , which intersect the cut in the palindromic factorization uv of w , are all the factors of w ; moreover they are distinct [3]. Hence the set of factors is the unambiguous product of the set of suffixes of u by the set of prefixes of v . Making the letters commute, the previous polynomial is the product of two polynomials; these are reciprocal, by palindromicity of u and v .

To prove Theorem 3.2, it is enough to prove that w is unbordered. One show that w has a nontrivial period p if and only if the intersection of the support of δ_w and of the line of equation $x + y = p$ is a singleton. Hence, by symmetry of the support, if w has this period, w also has the period $n - p$, hence is not primitive, by a Fine-Wilf lemma.

Let us sketch the proof of Theorem 3.3. Suppose that u is a Christoffel word. Clearly, all circular factors of u are factors of w . An ad hoc construction then allows one to enumerate all factors of w , relating them to the circular factors of u , and implying that δ_w has the required symmetry.

Conversely, one shows that the hypothesis implies that u has at most $k + 1$ circular factors of length k , for $k = 0, 1, \dots, |u| - 1$; being primitive, it must have exactly $k + 1$ factors. Hence u is the conjugate of a Christoffel word. We conclude the result using periodicity as above.

For Theorem 4.1, a closer look at the combinatorics behind the algebraic proof using polynomials gives the bijection.

For Theorem 4.2, one notes that since w is balanced, there at most two points in the intersection of the support of δ_w and the line of equation $x + y = p$. One of them is given by the intersection of the lower path and the line, and corresponds to the prefix of length p of w . The other to the suffix of length p of w , since $w = amb$, m palindrome.

Finally, if $w = u^k$ is factor symmetric, then one shows as above that u is the conjugate of a Christoffel word. Conversely, each power of a conjugate of a Christoffel word is Sturmian, hence trapezoidal, hence factor-symmetric. This proves Theorem 4.3.

References

- [1] Mira-Cristiana Anisiu & Julien Cassaigne (2004): *Properties of the complexity function for finite words*. *Rev. Anal. Numér. Théor. Approx.* 33(2), pp. 123–139, doi:10.33993/jnaat332-767.
- [2] Jean Berstel & Aldo de Luca (1997): *Sturmian words, Lyndon words and trees*. *Theoret. Comput. Sci.* 178(1-2), pp. 171–203, doi:10.1016/S0304-3975(96)00101-6.
- [3] Jean-Pierre Borel & Christophe Reutenauer (2006): *On Christoffel classes*. *Theor. Inform. Appl.* 40(1), pp. 15–27, doi:10.1051/ita:2005038.
- [4] Michelangelo Bucci, Alessandro De Luca & Gabriele Fici (2013): *Enumeration and structure of trapezoidal words*. *Theoret. Comput. Sci.* 468, pp. 12–22, doi:10.1016/j.tcs.2012.11.007.
- [5] Wai-fong Chuan (1998): *Unbordered factors of the characteristic sequences of irrational numbers*. *Theoret. Comput. Sci.* 205(1-2), pp. 337–344, doi:10.1016/S0304-3975(98)00104-2.
- [6] Wai-Fong Chuan (1999): *Sturmian morphisms and α -words*. *Theoret. Comput. Sci.* 225(1-2), pp. 129–148, doi:10.1016/S0304-3975(97)00239-9.
- [7] Flavio D’Alessandro (2002): *A combinatorial problem on trapezoidal words*. *Theoret. Comput. Sci.* 273(1-2), pp. 11–33, doi:10.1016/S0304-3975(00)00431-X. WORDS (Rouen, 1999).
- [8] Tero Harju & Dirk Nowotka (2004): *Minimal Duval extensions*. *Internat. J. Found. Comput. Sci.* 15(2), pp. 349–354, doi:10.1142/S0129054104002467.
- [9] Florence Levé & Patrice Séébold (2001): *Proof of a conjecture on word complexity*. *Bull. Belg. Math. Soc. Simon Stevin* 8(2), pp. 277–291, doi:10.36045/bbms/1102714173. Available at <http://projecteuclid.org/euclid.bbms/1102714173>. Journées Montoises d’Informatique Théorique (Marne-la-Vallée, 2000).
- [10] Aldo de Luca (1999): *On the combinatorics of finite words*. *Theoret. Comput. Sci.* 218(1), pp. 13–39, doi:10.1016/S0304-3975(98)00248-5. WORDS (Rouen, 1997).
- [11] S. Mantaci, A. Restivo & M. Sciortino (2003): *Burrows-Wheeler transform and Sturmian words*. *Inform. Process. Lett.* 86(5), pp. 241–246, doi:10.1016/S0020-0190(02)00512-4.
- [12] Sabrina Mantaci, Antonio Restivo, Giuseppe Romana, Giovanna Rosone & Marinella Sciortino (2021): *A combinatorial view on string attractors*. *Theoret. Comput. Sci.* 850, pp. 236–248, doi:10.1016/j.tcs.2020.11.006.
- [13] Christophe Reutenauer (2019): *From Christoffel words to Markoff numbers*. Oxford University Press, Oxford.
- [14] Christophe Reutenauer (2021): *Christoffel words and weak Markoff theory*. *Adv. in Appl. Math.* 127, pp. Paper No. 102179, 15, doi:10.1016/j.aam.2021.102179.

Combinatorics on Social Configurations

Dylan Laplace Mermoud

UMA, ENSTA Paris, Institut Polytechnique de Paris,
Paris, France.

dylan.laplace@ensta-paris.fr

Pierre Popoli

Department of Mathematics, ULiège
Liège, Belgium.

Pierre.Popoli@uliege.be

In cooperative game theory, the social configurations of players are modeled by balanced collections [2, 3]. A balanced collection is a set system defined on the set N of players in the game, together with a system of weights such that each player belongs to coalitions whose weights sum to 1. The Bondareva–Shapley theorem, perhaps the most fundamental theorem in cooperative game theory, characterizes the existence of solutions to the game that benefit everyone using balanced collections. Roughly speaking, if the trivial set system $\{N\}$ is one of the most efficient balanced collections for the game, then the set of solutions from which each coalition benefits, the so-called *core*, is non-empty.

In the following, we discuss some interactions between combinatorics and cooperative game theory that are still relatively unexplored. First, we study the similarities between balanced collections on the one hand and regular or uniform hypergraphs on the other. Second, we present some results leading to the construction of the combinatorial species of structures of uniform hypergraphs, from which we aim to construct the species of regular hypergraphs by duality. Finally, we investigate the possibility of expressing some “minimality” properties of regular or uniform hypergraphs in the language of combinatorial species, hoping to obtain new properties of minimal balanced collections.

1 Cooperative Game Theory

Cooperative game theory aims to study the emergence of cooperative behavior between rational players whose actions affect each other’s well-being. It was introduced in the seminal book *Theory of Games and Economic Behavior* by von Neumann and Morgenstern [6], written during the Second World War, motivated by von Neumann’s desire to study the stability of social organizations.

Definition 1 (von Neumann and Morgenstern [6]). A *cooperative game with transferable utility*, hereafter called *game*, is an ordered pair (N, v) where

- N is a non-empty finite set of *players*, called the *grand coalition*,
- v is a set function $v : 2^N \rightarrow \mathbb{R}$ such that $v(\emptyset) = 0$.

The non-empty subsets of N are called *coalitions*, and their set is denoted by \mathcal{N} . For each coalition $S \in \mathcal{N}$, the number $v(S)$, called the *worth* of S , can be interpreted as the amount of *utility* or *satisfaction* that the players forming S can obtain through full cooperation. When a coalition is formed, a non-trivial task is to allocate among its players the utility acquired by the coalition among its players. To prevent the coalition from splitting, the allocation of each of its subcoalitions must at least pay off its value, otherwise the coalitions would defect to obtain more utility. A necessary condition for the formation of the grand coalition is therefore that the following set

$$C(v) = \left\{ x \in \mathbb{R}^N \mid \sum_{i \in N} x_i = v(N), \text{ and } \sum_{i \in S} x_i \geq v(S), \forall S \in \mathcal{N} \right\}$$

is not empty. The set $C(v)$ is called the *core* of the game and is one of the essential objects studied in cooperative game theory. Each vector $x \in \mathbb{R}^N$ represents a payment to the players when player $i \in N$ receives a payment from x_i . The payment of a coalition is the sum of the payments of its players. Thus, the vectors in the core are exactly the payments that allocate the utility acquired by the large coalition in such a way that each coalition is satisfied with its payment.

A closely related object is the balanced collection. Formally, a balanced collection \mathcal{B} is a set of coalitions such that there exists a map $\lambda : \mathcal{B} \rightarrow \mathbb{R}_{>0}$ satisfying $\sum_{S \in \mathcal{B}, S \ni i} \lambda(S) = 1$ for each player $i \in N$. For example, the set partitions of N are balanced collections with unit weights. We measure the efficiency of a balanced collection \mathcal{B} by taking the weighted sum of the worths of the coalitions in \mathcal{B} , that is $\sum_{S \in \mathcal{B}} \lambda(S)v(S)$.

Theorem 1 (Bondareva [2], Shapley [3]). *The core of a game is nonempty if and only if $\{N\}$ belongs to the set of maximally efficient balanced collections.*

The Bondareva-Shapley theorem provides a useful characterization of the core nonemptiness, from which the first author, Grabisch and Sudhölter [5] developed an algorithm. This algorithm is based on an improved characterization of core nonemptiness, often called the sharp Bondareva-Shapley theorem, which differs from the previously mentioned theorem only in that the balanced collections are replaced by the *minimal balanced collections*. The minimal balanced collections are the balanced collections for which no proper subcoalitions are balanced. Moreover, the set of minimal balanced collections is the minimal, with respect to inclusion, set of balanced collections for which the Bondareva-Shapley theorem holds. In the same paper, the first author, Grabisch and Sudhölter [5] have generated the minimal balanced collections up to 7 players. The sequence of the numbers of the minimal balanced collections is stored as [A355042](#) in the *Online Encyclopedia of Integer Sequences* [7]. The method used in the aforementioned paper is inefficient when the number of players is greater than 7, and this work aims to find another way to generate it.

n	2	3	4	5	6	7
k	2	6	42	1,292	200,214	132,422,036

Table 1: Number k of minimal balanced collections according to the number n of players.

2 Hypergraphs

The cornerstone of this work is the striking similarity between the balanced collections and the regular hypergraphs. An (*undirected*) *hypergraph* \mathcal{H} is a pair $\mathcal{H} = (N, E)$, where N is a set of *nodes* and E is a spanning collection of non-empty subsets of N , called *hyperedges* or simply *edges*.

A hypergraph \mathcal{H} is called *k-regular* if for each node $x, \in N$ the *degree* of x is k , i.e. $\delta(x) := |\{e \in E \mid e \ni x\}| = k$. The underlying set of the multiset of edges of a regular hypergraph is a balanced collection. Indeed, the weight of a given coalition is the multiplicity of the edge in the collection E divided by the regularity of the hypergraph. If each edge has cardinality d , the hypergraph is said to be *d-uniform*. Therefore, the dual of a d -regular hypergraph is d -uniform and vice versa.

One of the main interests of uniform hypergraphs lies in the fact that writing a program that generates uniform hypergraphs of a certain size, i.e. with a certain number of edges, is extremely simple. It is sufficient to take arbitrary sets of equal cardinality and relabel their elements so that they fit into the notation $N = \{1, \dots, n\}$. If an edge needs to be added to the hypergraph, any set of nodes with the

appropriate cardinality can be used. However, if we want to add a node in a regular hypergraph, it is not easy to add it while maintaining regularity. Note that adding an edge to a uniform hypergraph is the same operation as adding a node to its dual regular hypergraph.

We believe that this approach is a possible route to a more efficient method for generating minimal balanced collections. Let $\mathcal{H} = (N, E)$ be a hypergraph, let $A \subseteq N$ and $X \subseteq E$. The hypergraph denoted by \mathcal{H}_A and defined by $\mathcal{H}_A = (A, \{S \cap A \mid S \in E\})$ is the *subhypergraph* of \mathcal{H} induced by A . The hypergraph $\mathcal{H}^X = (N, X)$ is the *partial hypergraph* of \mathcal{H} induced by X . Note that the subhypergraph of a hypergraph corresponds to a partial hypergraph of its dual.

Similarly to minimal balanced collections, we say that a hypergraph is *minimally uniform* if it is uniform and no proper subhypergraph is uniform, and we say that a hypergraph is *minimally regular* if it is regular and no proper partial hypergraph is regular.

Proposition 1. *The dual of a minimally uniform hypergraph is minimally regular and vice versa.*

Note that our definition of a subhypergraph does not eliminate the edges that become empty when taking the intersection with the subset of nodes. This definition is not common in the literature, but it is a natural one in our context, and the proposition above illustrates this fact.

3 Species of structures

Our goal now is to generate the objects we mentioned. To do this, we use the theory of *species of structures* and the corresponding operations on formal power series developed by Joyal [4].

A species of structures is a rule F that assigns to each finite set U a finite set $F[U]$ that is “independent of the nature” of the elements of U . The members of $F[U]$, called *F-structures*, are interpreted as combinatorial structures on the set U given by the rule F . The fact that the rule is independent of the nature of the elements of U is expressed by the invariance under relabeling. More precisely, to any bijection $\sigma : U \rightarrow V$ the rule F associates a bijection $F[\sigma] : F[U] \rightarrow F[V]$ that transforms each F-structures on U into an (isomorphic) F structure on V .

Each species is associated with a formal power series, which refers to the enumeration of F structures and is denoted by $F(x)$. There are a myriad of operations on species of structures such as addition, multiplication, functorial and partitional composite, see [1] for more details and further operations. The main interest of these operations is to provide a new description of a species of structures and to extract formulas over the generating series.

Example 1. Let \wp denote the species of subsets associating to each finite set U the set of subsets of U , and $\wp^{[2]}$ the species of the 2-subsets, or unordered pairs, defined similarly. Their generating series are, respectively, $\wp^{[2]}(x) = \sum_{n \geq 0} \binom{n}{2} \frac{x^n}{n!}$ and $\wp(x) = \sum_{n \geq 0} 2^n \frac{x^n}{n!} = e^{2x}$. Thanks to these two species and the composition of species, we have the following combinatorial identity

$$\text{GR} = \wp \square \wp^{[2]} \tag{1}$$

where GR is the species of simple graphs. From this formula, we obtain the generating series of simple graph, namely $\text{GR}(x) = \sum_{n \geq 0} 2^{\binom{n}{2}} \frac{x^n}{n!}$. An illustration of this identity is pictured in Figure 1.

Let us denote E the species of sets and $\zeta^{[p]}$ the species of k -subsets. Similarly to the combinatorial identity (1), we have proved the following formula.

Theorem 2. *The species of k -uniform hypergraphs of size p , which we denote by $\text{UNI}_{k,p}$, satisfies the following combinatorial equation:*

$$E \cdot \text{UNI}_{k,p} = \zeta^{[p]} \square \wp^{[k]}.$$



Figure 1: Construction of the species GR of simple graphs.

Let us denote $\bar{n} = n(n+1) \cdots (n+p)$. Using the formalism of virtual species, see [1] again for more details, we have the following corollary.

Corollary 1. *The generating series of the species $\text{UNI}_{k,p}$ is*

$$\text{UNI}_{k,p}(x) = \sum_{n \geq 0} \left(\sum_{i=0}^n (-1)^{n-i} \binom{n}{i} \frac{\binom{i}{k}^{\bar{p}}}{p!} \right) \frac{x^n}{n!}.$$

Example 2. Let us count the 2-uniform hypergraphs of size 3, with no more than three nodes. Since the hypergraphs are 2-uniform, n only goes from 2 to 3. Note that the number of hypergraphs is not counted up to an isomorphism. The number we are looking for is therefore

$$\begin{aligned} \sum_{n=2}^3 \left(\sum_{i=2}^n (-1)^{n-i} \binom{n}{i} \frac{\binom{i}{2}^{\bar{p}}}{p!} \right) &= (-1)^0 \binom{2}{2} \frac{\binom{2}{2}^{\bar{3}}}{3!} + (-1)^1 \binom{3}{2} \frac{\binom{2}{2}^{\bar{3}}}{3!} + (-1)^0 \binom{3}{3} \frac{\binom{3}{2}^{\bar{3}}}{3!} \\ &= 1 - 3 + 10 = 8. \end{aligned}$$

We represent them in the following. Notice that among these 8 uniform hypergraphs, only one is minimal, that is the triangle.

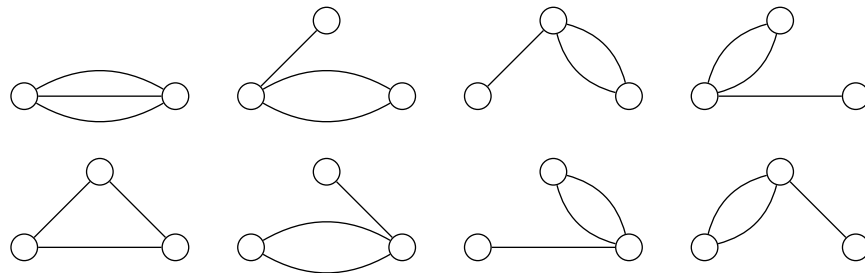


Figure 2: All 2-uniform hypergraphs of size 3 with no more than 3 nodes.

One can define the species of minimal balanced collections which are the underlying sets of the multisets of edges of minimally regular hypergraphs, which we construct from the minimally uniform hypergraphs, thanks to Proposition 1. For now, we simply have constructed the species of uniform hypergraphs.

Problem 1. *Express minimality in terms of species of structures.*

4 Decompositions in minimally uniform hypergraphs

Our first approach to study Problem 1 was based on the idea that minimally uniform hypergraphs (resp. minimally regular hypergraphs) are the building blocks of uniform hypergraphs (resp. regular hypergraphs). In this work, we also assume that the number of edges remains the same since the goal is that they represent the number of players, and this should be fixed. The following proposition states that a uniform hypergraph can be partitioned into smaller minimally uniform hypergraph.

Proposition 2. *Let $\mathcal{H} = (N, E)$ be a uniform hypergraph of size p . Then there exists a partition π of N such that for each element $B \in \pi$ the subhypergraph \mathcal{H}_B is minimally uniform of size p .*

One can expect that the aforementioned partition is unique, up to a permutation, and therefore leads to a combinatorial identity via structures of species. However, the next example shows that such a partition is not unique.

Example 3. Let us consider \mathcal{H} the 4-uniform hypergraph of order 7 and size 4 defined by

$$\mathcal{H} = (\{v_1, \dots, v_7\}, \{\{v_1, v_2, v_3, v_4\}, \{v_1, v_5, v_6, v_7\}, \{v_3, v_4, v_5, v_6\}, \{v_3, v_4, v_6, v_7\}\}).$$

The hypergraph \mathcal{H} can be partitioned in the two following ways

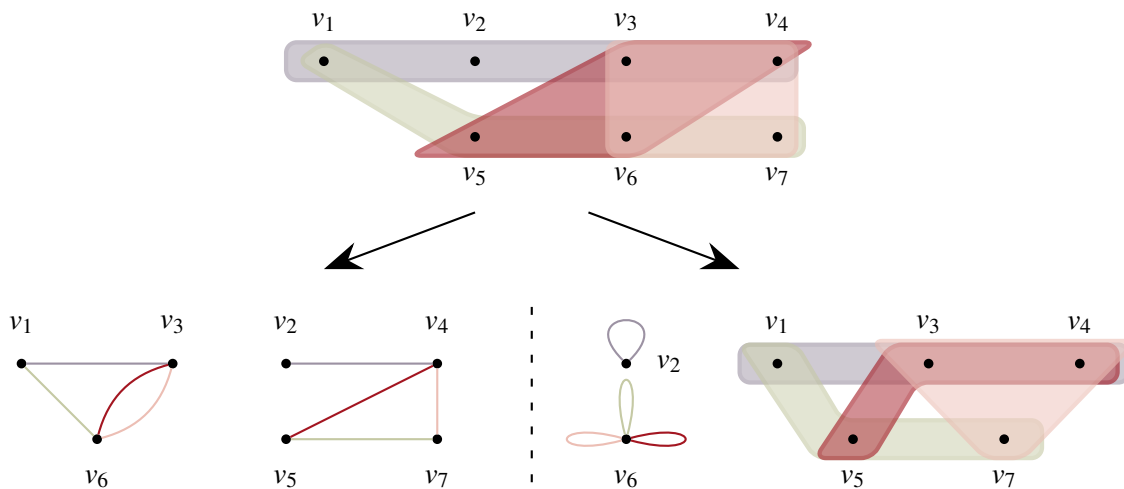


Figure 3: Example of a non-unique partition.

We can easily verify that

- The two hypergraphs on the left side $\mathcal{H}_1 = (\{v_1, v_3, v_6\}, \{\{v_1, v_3\}, \{v_1, v_6\}, \{v_3, v_6\}, \{v_3, v_6\}\})$ and $\mathcal{H}_2 = (\{v_2, v_4, v_5, v_7\}, \{\{v_2, v_4\}, \{v_4, v_5\}, \{v_4, v_7\}, \{v_5, v_7\}\})$ are minimally 2-uniform hypergraphs that will merge to \mathcal{H} .
- The hypergraphs on the right side $\mathcal{H}'_1 = (\{v_2, v_6\}, \{\{v_2\}, \{v_6\}, \{v_6\}, \{v_6\}\})$ and $\mathcal{H}'_2 = (\{v_1, v_3, v_4, v_5, v_7\}, \{\{v_1, v_3, v_4\}, \{v_1, v_5, v_7\}, \{v_3, v_4, v_5\}, \{v_3, v_4, v_7\}\})$ are respectively minimally 1-uniform and minimally 3-uniform hypergraphs that will merge to \mathcal{H} .

Therefore, such a decomposition in the state cannot lead to a combinatorial identity such as Theorem 2, which contains the species of minimally uniform hypergraph.

References

- [1] F. Bergeron, G. Labelle & P. Leroux (1998): *Combinatorial species and tree-like structures*. *Encyclopedia of Mathematics and its Applications* 67, Cambridge University Press, Cambridge. Translated from the 1994 French original by Margaret Readdy, With a foreword by Gian-Carlo Rota.
- [2] O. N. Bondareva (1963): *Some applications of the methods of linear programming to the theory of cooperative games*. *Problemy Kibernet.* 10, pp. 119–139.
- [3] A. Charnes & K. Kortanek (1967): *On balanced sets, cores, and linear programming*. *Cahiers Centre Études Rech. Opér.* 9, pp. 32–43.
- [4] André Joyal (1981): *Une théorie combinatoire des séries formelles*. *Adv. in Math.* 42(1), pp. 1–82, doi:10.1016/0001-8708(81)90052-9.
- [5] Dylan Laplace Mermoud, Michel Grabisch & Peter Sudhölter (2023): *Minimal balanced collections and their application to core stability and other topics of game theory*. *Discrete Appl. Math.* 341, pp. 60–81, doi:10.1016/j.dam.2023.07.025.
- [6] John von Neumann & Oskar Morgenstern (1944): *Theory of Games and Economic Behavior*. Princeton University Press, Princeton, NJ.
- [7] Neil J. A. Sloane & The OEIS Foundation Inc. (2024): *The Online Encyclopedia of Integer Sequences*. Available at <http://oeis.org/>.

Perfectly Clustering Words and Iterated Palindromes over a Ternary Alphabet

Mélodie Lapointe

Nathan Plourde-Hébert

Université de Moncton
Moncton, Canada

melodie.lapointe@umoncton.ca

enp0579@umoncton.ca

Recently, a new characterization of Lyndon words that are also perfectly clustering was proposed by Lapointe and Reutenauer (2024). A word over a ternary alphabet $\{a, b, c\}$ is called perfectly clustering Lyndon if and only if it is the product of two palindromes and it can be written as $a\pi_1 b\pi_2 c$ where π_1 and π_2 are palindromes. We study the properties of palindromes appearing as factors π_1 and π_2 and their links with iterated palindromes over a ternary alphabet.

1 Introduction

The Burrows-Wheeler transform of a word w , denoted $\text{bw}(w)$, is obtained from w by first listing the conjugates of w in lexicographic order, then concatenating the final letters of the conjugates in this order. For example, the Burrows-Wheeler transform of *apartment* is *tpmteaanr*. It was introduced in [2] as a tool in data compression. After applying the Burrows-Wheeler transform to a word, the occurrence of a given letter tend to occur in clusters. This clustering effect is optimal when all occurrences of each letter are group together. Words showing that optimal properties are thus called π -clustering. The permutation π represent the order in which the cluster of similar letters appear. The word aluminium, for example, is 451623-clustering since $\text{bw}(\text{aluminium}) = \text{mmnauuiil}$. A word w is *perfectly clustering* if its Burrows-Wheeler transform is a decreasing word, i.e., the clusters of letters appear from highest to lowest with respect to the alphabet order. This terminology was introduced by Ferenczi and Zamboni [4].

Perfectly clustering words were proposed in [11, 4] as a generalization of Christoffel words. Recently, Reutenauer and the first author [6] showed that a primitive word w is a perfectly clustering Lyndon word if and only if it is a product of two palindromes and has a palindromic special factorization, i.e., $w = a_1\pi_1 a_2\pi_2 \cdots \pi_{k-1} a_k$, where the letters in w are in $\{a_1 < a_2 < \cdots < a_k\}$ and $\pi_1, \pi_2, \dots, \pi_k$ are palindromes. This is also a generalization of characterization of Christoffel word due to de Luca and Mignosi [8]; a binary word amb is a Christoffel word if and only if the word amb is a product of two palindromes and m is also a palindrome called a central word. Hence, the palindromic special factorization of a Christoffel word in $\{a, b\}^*$ is simply amb where m is a palindrome. Central words have many properties (see [1, 10] for more information). We recall only one of them: a central word is the image of a mapping called iterated palindromization [7].

In this extended abstract, we discuss results about the palindromes in the palindromic special factorization of perfectly clustering Lyndon words over ternary alphabet. In Section 2, we recall some definitions about these words. In Section 3, we explore some relationships between the palindromes appearing in the special factorization. In Section 4, we describe the iterated palindromes that are factors of this factorization of perfectly clustering Lyndon words.

2 Definition

2.1 Words

For the rest of the paper, let $A = \{a, b, c\}$ be a totally ordered alphabet, where $a < b < c$. Let $w = w_1 w_2 \cdots w_n$ be a word in the free monoid generated by A . The length of $w = w_1 \cdots w_n$ (with $w_i \in A$), denoted by $|w|$, is n . The number of occurrences of a letter x in w is denoted by $|w|_x$. The *Parikh vector* of w is the integer vector $(|w|_a, |w|_b, |w|_c)$. The function Alph is defined by $\text{Alph}(w) = \{x \in A \mid |w|_x \geq 1\}$.

A word w is called *primitive* if it is not the power of another word; that is, for any word z such that $w = z^n$, one has $n = 1$. The *conjugates* of a word w are the words $w_i \cdots w_n w_1 \cdots w_{i-1}$. In other words, two words $u, v \in A^*$ are *conjugate* if for some words $x, y \in A^*$, one has $u = xy$ and $v = yx$. The *conjugation class* of a word is the set of its conjugates. If a word w of length n is primitive, then it has exactly n distinct conjugates. A word w is called a *Lyndon word* if it is primitive, and it is the minimal word in lexicographic order among its conjugates.

The *reversal* of $w = w_1 \cdots w_n$, denoted by $R(w)$, is the word $R(w) = w_n \cdots w_1$. A *palindrome* is a word w such that $w = R(w)$. A word u is a *factor* of w if there exists two words $x, y \in A^*$ such that $w = xuy$. The set of factors of w is denoted by $\text{Fact}(w)$ and $\text{Fact}_k(w)$ denotes the set of factors of length k of w .

2.2 Perfectly Clustering Lyndon Words

The *special factorization* of a word w over A is a factorization of w of the form $w = a\pi_1 b\pi_2 c$, where $\pi_1, \pi_2 \in A^*$. If π_1 and π_2 are both palindromes, then the special factorization is called *palindromic*. A *perfectly clustering Lyndon word* on A^* is a word w such that w is a product of two palindromes and has a palindromic special factorization. For example, the word $acbcbbcbc$ is a perfectly clustering Lyndon word since it is the product of the palindromes a and $cbcbcbcb$ and it has the palindromic special factorization $a \cdot cbc \cdot b \cdot bcb \cdot c$. Moreover, the palindromic special factorization of a perfectly clustering Lyndon word is unique [6].

This is not the original definition of perfectly clustering words, but of a characterization of perfectly clustering Lyndon word given in [6]. Usually, a word w is called perfectly clustering if its Burrows-Wheeler transform is $c^{|w|_c} b^{|w|_b} a^{|w|_a}$ (see [11] for a complete definition). If a primitive word is perfectly clustering, then all its conjugates are. Consequently, there is no loss of generality in studying only perfectly clustering Lyndon words. The set of perfectly clustering Lyndon words is denoted by \mathcal{P} .

It was proved by Mantaci, Restivo and Sciortino [9, Theorem 9] that perfectly clustering words on a binary alphabet are Christoffel words and their conjugates. Let recall the following lemma describing the possible sets of factors of length 2 of a perfectly clustering word.

Lemma 1 ([11]). *Let w be a perfectly clustering word in $\{a, b, c\}^*$. Then $\text{Fact}_2(w)$ is a subset of one of the sets below:*

- $\{ab, ac, ba, bb, ca\}$
- $\{ac, bb, bc, ca, cb\}$
- $\{aa, ab, ac, ba, ca\}$
- $\{ac, bc, ca, cb, cc\}$

2.3 Iterated Palindromes

The (*right*) *palindromic closure* of w , denoted by $w^{(+)}$, is the shortest unique palindrome having w as a prefix, i.e., if $w = ps$ where s is the longest palindromic suffix of w , then $w^{(+)} = psR(p)$. We define the mapping $\text{Pal}(w)$ from a free monoid to itself, called iterated palindromization, as follows: $\text{Pal}(\varepsilon) = \varepsilon$ and

for each letter x , $\text{Pal}(ux) = (\text{Pal}(u)x)^{(+)}$. A word w such that $w = \text{Pal}(u)$ is called an *iterated palindrome* and u is called the *directive word* of w . For example, the word $ababaababa$ is an iterated palindrome and its directive word is $abba$. A word amb is a Christoffel word if and only if m is an iterated palindrome on a binary alphabet [7]. Therefore, iterated palindromes on a binary alphabet are the only palindromes in the palindromic special factorization of Christoffel words.

3 Sets of palindromes

From the set \mathcal{P} , let define two sets of words P_1 and P_2 as follows.

$$P_1 = \{\pi_1 \mid a\pi_1 b\pi_2 c \in \mathcal{P}\}$$

$$P_2 = \{\pi_2 \mid a\pi_1 b\pi_2 c \in \mathcal{P}\}.$$

By definition, we know that all the words in P_1 and P_2 are palindromes. However, these sets are not equal, as shown in the next proposition.

Proposition 2. $P_1 \neq P_2$

Proof. One can check that $a \cdot bcb \cdot b \cdot bcb \cdot c$ is a perfectly clustering Lyndon word with the given palindromic special factorization. Hence, the palindrome $bcb \in P_2$.

It is sufficient to show that $bcb \notin P_1$, i.e., that for any word $u \in A^*$, the word $a \cdot bcb \cdot b \cdot u \cdot c$ is not a perfectly clustering Lyndon word. The set of factors $\{ab, bc, cb, bb\} \subseteq \text{Fact}_2(abcbuc)$ but $\{ab, bc, cb, bb\}$ is not a subset of the one of the set in Lemma 1. Thus, the word $a \cdot bcb \cdot b \cdot u \cdot c$ is not a perfectly clustering Lyndon word and $bcb \notin P_1$. This means that $P_1 \neq P_2$. \square

Some palindromes are in both sets. For example, the words $a \cdot cac \cdot b \cdot c$ and $a \cdot c \cdot b \cdot cac \cdot c$ are both perfectly clustering Lyndon words. Thus, $cac \in P_1 \cap P_2$. The intersection between P_1 and P_2 is discussed in Section 4.

There is a relationship between P_1 and P_2 . Let θ be the morphism exchanging the letter a and c defined by $\theta(a) = c$, $\theta(b) = b$, $\theta(c) = a$. The antimorphism, ω defined as $\omega = R \circ \theta$, send perfectly clustering Lyndon word to perfectly clustering Lyndon word [5].

Lemma 3. $P_1 = \theta(P_2)$

Proof. Let $p \in P_1$ be an arbitrary palindrome. There exist a perfectly clustering Lyndon word w and a palindrome $u \in A^*$ such that $w = apbuc$. Then

$$\omega(w) = (R \circ \theta)(apbuc) = R(c\theta(p)b\theta(u)a) = a(R \circ \theta)(u)b(R \circ \theta)(p)c.$$

The word $\omega(w)$ is a perfectly clustering Lyndon word with the given palindromic special factorization. Since p is a palindrome, $(R \circ \theta)(p) = \theta(p)$ and $\theta(p) \in P_2$. Similarly, we show that $\theta(P_2) \subseteq P_1$. Therefore $P_1 = \theta(P_2)$. \square

4 Iterated palindromes in the previous sets

Some iterated palindromes appears in P_1 and P_2 , but those sets also contain words which are not iterated palindromes. For example, $bacab$ is a palindrome in P_1 which is not an iterated palindrome since the word $a \cdot bacab \cdot b \cdot a \cdot c$ is a perfectly clustering Lyndon word.

Proposition 4. *Let $u \in A^*$ be a word. The iterated palindrome $\text{Pal}(u) \in P_1$ if and only if $u \in \{a, c\}^* \cdot \{a, b\}^*$.*

The proof of Proposition 4 uses induction and the construction of perfectly clustering Lyndon word proposed in [5]. We defined four automorphisms of the free group $F(A)$ by $\lambda_a(a) = a$, $\lambda_a(b) = ab$ and $\lambda_a(c) = ac$; $\lambda_b(a) = ab^{-1}$, $\lambda_b(b) = b$ and $\lambda_b(c) = bc$; $\rho_b(a) = ab$, $\rho_b(b) = b$ and $\rho_b(c) = b^{-1}c$ and $\rho_c(a) = ac$, $\rho_c(b) = bc$ and $\rho_c(c) = c$. It was proved in [5] that for each perfectly clustering word $w \in A^*$ of length at least 3, there exists a shorter perfectly clustering word $u \in A^*$ and an automorphism $f \in \{\lambda_a, \lambda_b, \rho_b, \rho_c\}$ such that $w = f(u)$. Since the word $abac$ is a perfectly clustering Lyndon word, we only need to show that $f(a\varepsilon bac) = a\text{Pal}(u)bqc$ where $f = f_{x_1} \circ f_{x_2} \circ \dots \circ f_{x_n}$, $f_{x_i} \in \{\lambda_a, \rho_b, \rho_c\}$, $u = x_1x_2 \dots x_n$ and $q \in P_2$. Moreover, the following lemma means that no other iterated palindrome can be in P_1 .

Lemma 5. *Let $u \in A^*$ be a word such that $\text{Alph}(u) = A$. The iterated palindrome $\text{Pal}(bu)$ is not in P_1 , nor in P_2 .*

Proof. Let $xv \in A^*$ be a word. In [3], it is shown that the first letter of a directive word is separating for $\text{Pal}(xv)$, i.e., the letter x appears in each factor of length 2 of $\text{Pal}(xv)$. Hence, the letter b appears in each factor of length 2 of $\text{Pal}(xv)$ and $ac \notin \text{Fact}_2(\text{Pal}(bu))$. However, ac is a factor in each set given in Lemma 1. Thus, $\text{Fact}_2(\text{Pal}(bu))$ cannot be a factor of a perfectly clustering word. \square

Using Lemma 3 and Proposition 4, one may describe the iterated palindromes which are elements of P_2 .

Proposition 6. *Let $u \in A^*$ be a word. The iterated palindrome $\text{Pal}(u) \in P_2$ if and only if $u \in \{a, c\}^* \cdot \{b, c\}^*$.*

From the previous proposition one may deduce which iterated palindrome are in $P_1 \cap P_2$.

Proposition 7. *Let $u \in A^*$ be a word. The iterated palindrome $\text{Pal}(u) \in P_1 \cap P_2$ if and only if $u \in \{a, c\}^* \cdot b^*$.*

Following computer exploration, we believe that the conjecture below is valid.

Conjecture 8. *A word $w \in P_1 \cap P_2$ if and only if $\text{Pal}(u) = w$ and $u \in \{a, c\}^* \cdot b^*$.*

Iterated palindromes represent a small subset of the palindromes in P_1 and P_2 . Those results are a step in the characterization of these sets that the authors intend to pursue. A more general question is to characterize the palindromes in the special factorization of perfectly clustering Lyndon words on any alphabets.

References

- [1] Jean Berstel (2007): *Sturmian and episturmian words (a survey of some recent results)*. In: *Algebraic informatics, Lecture Notes in Comput. Sci.* 4728, Springer, Berlin, pp. 23–47. doi:10.1007/978-3-540-75414-5_2.
- [2] M. Burrows & D. J. Wheeler (1994): *A block-sorting Lossless data compression algorithm*. Technical report, Digital System Research Center, p. 18.
- [3] Xavier Droubay, Jacques Justin & Giuseppe Pirillo (2001): *Episturmian words and some constructions of de Luca and Rauzy*. *Theoret. Comput. Sci.* 255(1-2), pp. 539–553. doi:10.1016/S0304-3975(99)00320-5.
- [4] Sébastien Ferenczi & Luca Q. Zamboni (2013): *Clustering words and interval exchanges*. *J. Integer Seq.* 16(2), pp. Article 13.2.1, 9.

- [5] Mélodie Lapointe (2020): *Combinatoire des mots: mots parfaitement amassants, triplets de Markoff et graphes chenilles*. Ph.D. thesis, Université du Québec à Montréal, Montréal (Québec, Canada). Available at <https://archipel.uqam.ca/14120/>.
- [6] Mélodie Lapointe & Christophe Reutenauer (2024): *Characterizations of perfectly clustering words*. Submitted to *Electron. J. Combin.*
- [7] Aldo de Luca (1997): *Sturmian words: structure, combinatorics, and their arithmetics*. *Theoret. Comput. Sci.* 183(1), pp. 45–82. doi:10.1016/S0304-3975(96)00310-6
- [8] Aldo de Luca & Filippo Mignosi (1994): *Some combinatorial properties of Sturmian words*. *Theoret. Comput. Sci.* 136(2), pp. 361–385. doi:10.1016/0304-3975(94)00035-H
- [9] S. Mantaci, A. Restivo & M. Sciortino (2003): *Burrows-Wheeler transform and Sturmian words*. *Inform. Process. Lett.* 86(5), pp. 241–246. doi:10.1016/S0020-0190(02)00512-4
- [10] Christophe Reutenauer (2019): *From Christoffel words to Markoff numbers*. Oxford University Press, Oxford. doi:10.1093/oso/9780198827542.003.0003.
- [11] Jamie Simpson & Simon J. Puglisi (2008): *Words with simple Burrows-Wheeler transforms*. *Electron. J. Combin.* 15(1), pp. Research Paper 83, 17. doi:10.37236/807.

On the Orthogonality of Generalized Pattern Sequences

Shuo Li

Department of Mathematics & Statistics
The University of Winnipeg
Winnipeg, Canada
sh.li@uwinnipeg.ca

The partial sums of integer sequences that count the occurrences of a specific pattern in the binary expansion of positive integers have been investigated by different authors since the 1950s. In this note, we introduce generalized pattern sequences, which count the occurrences of a finite number of different patterns in the expansion of positive integers in any integer base, and analyze their partial sums.

1 Introduction, definitions and notation

Let b be a positive integer larger than 1. Define $[[b]] = \{0, 1, 2, \dots, b-1\}$ and $[[b]]^*$ as the set of finite words composed of letters from $[[b]]$. A finite weighted subset S of $[[b]]^*$ is a set of the form

$$\{(n_{S,w}, w) \mid n_{S,w} \in \mathbf{R}, w \in [[b]]^*\},$$

such that $|\{w \mid n_{S,w} \neq 0\}| < \infty$. For any $w \in [[b]]^* \setminus [[1]]^*$ and any non-negative integer n , let $e_{b,w}(n)$ denote the total number of occurrences of the word w in the b -expansion of n . In this article, by b -expansion of integers, we mean the canonical b -expansion of integers with infinitely many leading zeros. For example, $e_{2,0011}(6) = 1$, $e_{2,0011}(51) = 2$. For any weighted subset S of $[[b]]^*$ and any non-negative integer n , define

$$e_{b,S}(n) = \sum_{(n_{S,w}, w) \in S} n_{S,w} e_{b,w}(n).$$

For any positive integer m larger than 1, define $a_{b,m,w}(n) = \exp \frac{2\pi i e_{b,w}(n)}{m}$ and $a_{b,m,S}(n) = \exp \frac{2\pi i e_{b,S}(n)}{m}$ for all non-negative n . Both sequences $(a_{b,m,S}(n))_{n \in \mathbf{N}}$ and $(e_{b,m,S}(n))_{n \in \mathbf{N}}$ are well studied in the literature. The sequences $(e_{b,w}(n))_{n \in \mathbf{N}}$ are called *block-counting sequences* and $(e_{b,S}(n))_{n \in \mathbf{N}}$ are called *digital sequences* from [5, Chapter 3.3]. The analytical and combinatorial properties of these sequences in the case of $b = n = 2$ have been well studied since Thue. The sequences $(a_{2,2,S}(n))_{n \in \mathbf{N}}$ are called *pattern sequences* in [15, 11, 16]. For some special examples, the ± 1 -Thue-Morse sequence can be defined as $(a_{2,2,1}(n))_{n \in \mathbf{N}}$ (see, for example, [5, P. 15], the sequence defined there is actually the $\{0, 1\}$ -Thue-Morse sequence, the sequence $(a_{2,2,1}(n))_{n \in \mathbf{N}}$ can be obtained by changing 0 to 1 and 1 to -1 from the previous sequence) and the ± 1 -Rudin-Shapiro sequence can also be defined as $(a_{2,2,11}(n))_{n \in \mathbf{N}}$ (see, for example, [5, Example 3.3.1]). The asymptotic and combinatorial properties of $(a_{2,2,S}(n))_{n \in \mathbf{N}}$ and $(e_{2,2,S}(n))_{n \in \mathbf{N}}$ are studied in [2, 12, 8, 3, 4, 6].

Let $(f_n)_{n \in \mathbf{N}}$ and $(g_n)_{n \in \mathbf{N}}$ be two real sequences. They are *orthogonal* if

$$\lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=0}^N f_n g_n = 0.$$

The orthogonality between $(a_{2,2,w}(n))_{n \in \mathbf{N}}$ and periodic sequences was first studied by Rudin and Shapiro; they proved separately in [13] and [14] that

$$\max_{0 \leq \theta < 1} \left| \frac{1}{N} \sum_{n=0}^N a_{2,2,11}(n) e^{2\pi i n \theta} \right| \leq C \sqrt{N}, N \geq 1.$$

A similar result involving $(a_{2,2,1}(n))_{n \in \mathbf{N}}$ was obtained by Gelfond in [10]. Moreover, from [7, Proposition 3.1] and [5, Theorem 16.1.5], the sequences of the form $(a_{2,2,S}(n))_{n \in \mathbf{N}}$ are actually in the class of *automatic sequences* (see [5]) and the sequences of the form $(e^{2\pi i n \theta})_{n \in \mathbf{N}}$ are *multiplicative sequences* (see [9]). The orthogonality between automatic sequences and multiplicative sequences was studied in [9]. In a series of recent articles [15, 11, 16], the correlations of $(a_{2,2,S}(n))_{n \in \mathbf{N}}$ were studied from a viewpoint of dynamical systems. The focus of this paper is two-fold. First, we study the orthogonality among the sequences of the form $(a_{b,m,S}(n))_{n \in \mathbf{N}}$. We later prove that it amounts to study the partial sums of $(a_{b,m,S}(n))_{n \in \mathbf{N}}$. Second, we generalize the result in [8] concerning the sequences $(a_{2,2,w}(n))_{n \in \mathbf{N}}$ to $(a_{b,m,S}(n))_{n \in \mathbf{N}}$ for arbitrary b, m and S by using a recent result on the combinatorial structure of $(a_{b,m,w}(n))_{n \in \mathbf{N}}$ introduced in [1]. We give a necessary and sufficient condition for

$$\lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=0}^N a_{b,m,S}(n) = 0. \quad (\star)$$

The main results are announced in Theorem 3 and Theorem 5.

2 Window functions and $(a_{b,m,S}(n))_{n \in \mathbf{N}}$

A finite weighted subset S of $[[b]]^*$ is called *proper* if $n_{S,w} \neq 0$ implies w does not have leading zeros. For any two finite weighted subsets S_1, S_2 of $[[b]]^*$, define

$$S_1 \oplus S_2 = \{(n_{S_1,w} + n_{S_2,w}, w) \mid w \in [[b]]^*\}.$$

Any finite word w in $[[b]]^*$ is written $w = w[1]w[2] \cdots w[|w|]$ where $|w|$ is its *length*. For later use we denote $w' = w[2]w[3] \cdots w[|w|]$, and $(w)_b = \sum_{i=1}^{|w|} w[i]b^{|w|-i}$.

Let $\exp[[b]] = \{e^{\frac{2\pi i n}{b}} \mid n \in [[b]]\}$ and $\exp[[b]]^* = \{e^{\frac{2\pi i n}{b}} \mid n \in [[b]]^*\}$ be the set of finite words composed of letters from $\exp[[b]]$.

Let us generalize the definition of the *window function* in [1]. For any integers b, m larger than 1 and any $w \in [[b]]^*$, let $\alpha_w^1 = \frac{(w')_b}{b^{|w|-1}}$ and $\alpha_w^2 = \frac{(w')_{b+1}}{b^{|w|-1}}$. The *window function* $\phi_{b,m,w} : \exp[[b]]^* \rightarrow \exp[[b]]^*$ is such that for any $v \in \exp[[b]]^*$:

$$\phi_{b,m,w}(v)[j] = \begin{cases} e^{\frac{2\pi i}{m}} v[j], & \text{if } \alpha_w^1 |v| < j \leq \alpha_w^2 |v|; \\ v[j], & \text{otherwise.} \end{cases}$$

It is extended to finite weighted subset S of $[[b]]^*$, by setting

$$\phi_{b,m,S} = \prod_{(n_{S,w}, w) \in S} (\phi_{b,m,w})^{n_{S,w}}.$$

Proposition 1 Let b, m be two integers larger than 1 and let S_1, S_2 be two finite weighted subsets of $[[b]]^*$. For any non-negative integer n , one has

$$a_{b,m,S_1}(n)a_{b,m,S_2}(n) = a_{b,m,S_1 \oplus S_2}(n).$$

As a corollary, the orthogonality of two generalized pattern sequences is equivalent to the (\star) property of some generalized pattern sequence.

Proposition 2 Let b, m be two integers larger than 1 and let S be a finite weighted subset of $[[b]]^*$. There exists a finite proper weighted subset S' of $[[b]]^*$ such that for any non-negative integer n , one has $a_{b,m,S}(n) = a_{b,m,S'}(n)$.

An analog of Proposition 3 in [1] is stated in our first claimed result:

Theorem 3 Let b, m be two integers larger than 1 and let S be a finite weighted subset of $[[b]]^*$. Let $l = \max\{|w| \mid n_{S,w} \neq 0\}$. There exist $k-1$ proper weighted subset S_1, S_2, \dots, S_{b-1} of $[[b]]^*$ and a sequence $(u_t)_{t \in \mathbb{N}}$ in $\exp[[b]]^*$ such that:

1. $|u_0| = b^l$;
2. $u_{t+1} = u_t \phi_{b,m,S_1}(u_t) \phi_{b,m,S_2}(u_t) \cdots \phi_{b,m,S_{b-1}}(u_t)$ for all $t \geq 0$;
3. $(a_{b,m,S}(n))_{n \in \mathbb{N}} = \lim_{t \rightarrow \infty} u_t$.

3 Application

From Theorem 3, one can associate a generalized pattern sequence $(a_{b,m,S}(n))_{n \in \mathbb{N}}$ to a matrix $M_{b,m,S}$ in the following way:

1. let S_1, S_2, \dots, S_{b-1} and l be the same as in Theorem 3;
2. let V_0 be the constant sequence of 1 with a length of p^l , and let $V_k = \phi_{b,m,S_k}(V_0)$ for all $k \in \{1, 2, \dots, b-1\}$;
3. let $M_{b,m,S} \in \mathcal{M}_{p^l \times p^l}(\mathbb{C})$ such that for any integers $1 \leq r \leq b^{l-1}$, $0 \leq s \leq b-1$ and $1 \leq t \leq b$,

$$M_{b,m,S}(x,y) = \begin{cases} V_s[(r-1)b+t], & \text{if } (x,y) = (sb^{l-1}+r, (r-1)b+t); \\ 0, & \text{otherwise.} \end{cases}$$

Theorem 4 Let b, m be two integers larger than 1, let S be a finite weighted subset of $[[b]]^*$ and let $(a_{b,m,S}(n))_{n \in \mathbb{N}}$ and $M_{b,m,S}$ be respectively the associated generalized pattern sequence and the matrix. Let $(A_{b,m,S}(t))_{t \in \mathbb{N}}$ be a sequence of column vectors of dimension p^l such that for any integers $t \geq 0$ and $1 \leq j \leq p^l$,

$$A_{b,m,S}(t)(j) = \sum_{n=(j-1)p^l}^{jp^l-1} a_{b,m,S}(n).$$

Then for any integer $t \geq 0$, one has

$$A_{b,m,S}(t+1) = M_{b,m,S}A_{b,m,S}(t).$$

And we have our second claimed result.

Theorem 5 Let b, m be two integers larger than 1, let S be a finite weighted subset of $[[b]]^*$. The sequence $(a_{b,m,S}(n))_{n \in \mathbf{N}}$ satisfies the property (\star) if and only if at least one of the following conditions holds:

1. b is not an eigenvalue of $M_{b,m,S}$;
2. $M_{b,m,S}^{b-1} A_{b,m,S}(0) = \mathbf{0}$, where $A_{b,m,S}(0)$ is as the same as in Theorem 4.

Example 6 Let $b = m = 3$ and let $S = \{(n_{S,w}, w) | n_{S,w} \in \mathbf{R}, w \in \{0, 1\}^*\}$ satisfying

$$n_{S,w} = \begin{cases} 1, & \text{if } w = 1, 10, 12; \\ 2, & \text{if } w = 11, 22; \\ 0, & \text{otherwise.} \end{cases}$$

From Proposition 3 in [1], the window functions associated to $w = 1, 10, 12, 11, 22$ are respectively:

$$\begin{aligned} \phi_{3,3,1}(v)[j] &= e^{\frac{2\pi i}{q} v[j]}, & \phi_{3,3,12}(v)[j] &= \begin{cases} e^{\frac{2\pi i}{q} v[j]}, & \text{if } \frac{2}{3}|v| < j \leq |v|; \\ v[j], & \text{otherwise.} \end{cases}, \\ \phi_{3,3,10}(v)[j] &= \begin{cases} e^{\frac{2\pi i}{q} v[j]}, & \text{if } 0 < j \leq \frac{1}{3}|v|; \\ v[j], & \text{otherwise.} \end{cases} & \phi_{3,3,11}(v)[j] &= \begin{cases} e^{\frac{4\pi i}{q} v[j]}, & \text{if } \frac{1}{3}|v| < j \leq \frac{2}{3}|v|; \\ v[j], & \text{otherwise.} \end{cases}, \\ \phi_{3,3,22}(v)[j] &= \begin{cases} e^{\frac{4\pi i}{q} v[j]}, & \text{if } \frac{2}{3}|v| < j \leq |v|; \\ v[j], & \text{otherwise.} \end{cases}. \end{aligned}$$

From Theorem 3, one can find S_1, S_2 satisfying

$$n_{S_1,w} = \begin{cases} 1, & \text{if } w = 1, 10, 12; \\ 2, & \text{if } w = 11; \\ 0, & \text{otherwise.} \end{cases} \quad n_{S_2,w} = \begin{cases} 2, & \text{if } w = 22; \\ 0, & \text{otherwise.} \end{cases}$$

Thus,

$$\phi_{3,3,S_1}(v)[j] = \begin{cases} e^{\frac{2\pi i}{q} v[j]}, & \text{if } \frac{1}{3}|v| < j \leq \frac{2}{3}|v|; \\ e^{\frac{4\pi i}{q} v[j]}, & \text{otherwise.} \end{cases} \quad \phi_{3,3,S_2}(v)[j] = \begin{cases} e^{\frac{4\pi i}{q} v[j]}, & \text{if } \frac{2}{3}|v| < j \leq |v|; \\ v[j], & \text{otherwise.} \end{cases}.$$

From Theorem 3, set $u_0 = 1, 1, 1, e^{\frac{4\pi i}{3}}, 1, e^{\frac{4\pi i}{3}}, 1, 1, e^{\frac{4\pi i}{3}}$, and define a sequence of words $(u_n)_{n \in \mathbf{N}}$ such that $u_{m+1} = u_m \phi_{3,3,S_1}(u_m) \phi_{3,3,S_2}(u_m)$ for all integers $m \geq 0$, then $(a_{3,3,S}(n))_{n \in \mathbf{N}} = \lim_{m \rightarrow \infty} u_m$. Moreover, the associated matrix

$$M_{3,3,S} = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ e^{\frac{4\pi i}{q}} & e^{\frac{4\pi i}{q}} & e^{\frac{4\pi i}{q}} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & e^{\frac{2\pi i}{q}} & e^{\frac{2\pi i}{q}} & e^{\frac{2\pi i}{q}} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & e^{\frac{4\pi i}{q}} & e^{\frac{4\pi i}{q}} & e^{\frac{4\pi i}{q}} \\ e^{\frac{2\pi i}{q}} & e^{\frac{2\pi i}{q}} & e^{\frac{2\pi i}{q}} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & e^{\frac{2\pi i}{q}} & e^{\frac{2\pi i}{q}} & e^{\frac{2\pi i}{q}} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & e^{\frac{4\pi i}{q}} & e^{\frac{4\pi i}{q}} & e^{\frac{4\pi i}{q}} \end{pmatrix}$$

Since 3 is not an eigenvalue of $M_{3,3,S}$, we have

$$\lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=0}^N a_{3,3,S}(n) = 0.$$

Example 7 Let $b = m = 2$ and let U be a finite weighted subset of $\{0, 1\}^*$ satisfying

$$n_{U,w} = \begin{cases} 1, & \text{if } w = 1, 10, 11; \\ 0, & \text{otherwise.} \end{cases}$$

The associated matrix

$$M_{2,2,U} = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}.$$

One can easily verify that 2 is an eigenvalue of $M_{2,2,U}$. However, since $A_{2,2,U}(0) = (1, -1, 1, -1)^t$, and $M_{2,2,U}A_{2,2,U}(0) = \mathbf{0}$, one has

$$\lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=0}^N a_{2,2,U}(n) = 0.$$

In fact, one can verify $(a_{2,2,U}(n))_{n \in \mathbb{N}}$ is the $(1, -1)$ -periodic sequence.

Example 8 Let us consider $(a_{3,3,002}(n))_{n \in \mathbb{N}}$. One has, for any non-negative integer n ,

$$\begin{aligned} e_{3,3,002}(n) &= e_{3,3,02}(n) - e_{3,3,102}(n) - e_{3,3,202}(n) \\ &= e_{3,3,2}(n) - e_{3,3,12}(n) - e_{3,3,22}(n) - e_{3,3,102}(n) - e_{3,3,202}(n). \end{aligned}$$

Thus,

$$a_{3,3,002}(n) = a_{3,3,2}(n)a_{3,3,12}(n)a_{3,3,22}(n)a_{3,3,102}(n)a_{3,3,202}(n)^{-1}.$$

Define two finite weighted subsets P, Q of $\{0, 1, 2\}^*$ such that

$$n_{P,w} = \begin{cases} -1, & \text{if } w = 12, 102; \\ 0, & \text{otherwise.} \end{cases} \quad n_{Q,w} = \begin{cases} 1, & \text{if } w = 2; \\ -1, & \text{if } w = 22, 202; \\ 0, & \text{otherwise.} \end{cases}$$

From Theorem 3, set $u_0 = a_{3,3,002}(0), a_{3,3,002}(1), \dots, a_{3,3,002}(26)$, and define a sequence of words $(u_n)_{n \in \mathbb{N}}$ such that $u_{m+1} = u_m \phi_{3,3,P}(u_m) \phi_{3,3,Q}(u_m)$ for all integer $m \geq 0$, then $(a_{3,3,002}(n))_{n \in \mathbb{N}} = \lim_{m \rightarrow \infty} u_m$.

References

- [1] Antoine Abram, Yining Hu & Shuo Li (2024): *Block-counting sequences are not purely morphic*. *Advances in Applied Mathematics* 155, p. 102673, doi:10.1016/j.aam.2024.102673.
- [2] Jean-Paul Allouche & Pierre Liardet (1991): *Generalized Rudin-Shapiro sequences*. *Acta Arithmetica* 60(1), pp. 1–27, doi:10.4064/aa-60-1-1-27.
- [3] Jean-Paul Allouche & Jeffrey Shallit (1992): *The ring of k -regular sequences*. *Theoretical Computer Science* 98(2), pp. 163–197, doi:10.1016/0304-3975(92)90001-V.

- [4] Jean-Paul Allouche & Jeffrey Shallit (1999): *The ubiquitous Prouhet-Thue-Morse sequence*. In C. Ding, T. Helleseht & H. Niederreiter, editors: *Sequences and their Applications*, Springer London, London, pp. 1–16, doi:10.1007/978-1-4471-0551-0_1.
- [5] Jean-Paul Allouche & Jeffrey Shallit (2003): *Automatic Sequences: Theory, Applications, Generalizations*. Cambridge University Press, doi:10.1017/CB09780511546563.
- [6] Jean-Paul Allouche & Jeffrey Shallit (2003): *The ring of k -regular sequences, II*. *Theoretical Computer Science* 307(1), pp. 3–29, doi:10.1016/S0304-3975(03)00090-2.
- [7] Emmanuel Cateland (1992): *Suites digitales et suites k -régulières*. Theses, Université Sciences et Technologies - Bordeaux I. Available at <https://theses.hal.science/tel-00845511>.
- [8] Patrick Morton David W. Boyd, Janice Cook (1989): *On sequences of ± 1 's defined by binary patterns*. Instytut Matematyczny Polskiej Akademi Nauk. Available at <http://eudml.org/doc/268620>.
- [9] Nikos Frantzikinakis & Bernard Host (2019): *Furstenberg Systems of Bounded Multiplicative Functions and Applications*. *International Mathematics Research Notices* 2021(8), pp. 6077–6107, doi:10.1093/imrn/rnz037.
- [10] Alexandre O. Gelfond (1968): *Sur les nombres qui ont des propriétés additives et multiplicatives données*. *Acta Arithmetica* 13(3), pp. 259–265, doi:10.4064/aa-13-3-259-265.
- [11] Jakub Konieczny (2021): *Algorithmic classification of noncorrelated binary pattern sequences*. *Journal of Number Theory* 223, pp. 229–254, doi:10.1016/j.jnt.2020.10.008.
- [12] Morton Patrick (1990): *Connections between binary patterns and paperfolding*. *Séminaire de théorie des nombres de Bordeaux Ser. 2*, 2(1), pp. 1–12, doi:10.5802/jtnb.16.
- [13] Walter Rudin (1959): *Some theorems on Fourier coefficients*. *Proceedings of the American Mathematical Society* 10, pp. 855–859, doi:10.1090/S0002-9939-1959-0116184-5.
- [14] Harold Seymour Shapiro (1968): *Extremal problems for polynomials and power series*. Master's thesis, M.I.T. Available at <hdl.handle.net/1721.1/12198>.
- [15] Yu Zheng, Li Peng & Teturo Kamae (2018): *Characterization of noncorrelated pattern sequences and correlation dimensions*. *Discrete and Continuous Dynamical Systems* 38(10), pp. 5085–5103, doi:10.3934/dcds.2018223.
- [16] Yu Zheng, Li Peng & Teturo Kamae (2021): *Spectral properties of pattern sequences of general degrees*. *Nonlinearity* 34(5), p. 3411, doi:10.1088/1361-6544/abe91a.

Counting Polyominoes in a Rectangle $b \times h$

Louis Marin

UQAM LACIM

Montreal, Canada

marin.louis@courrier.uqam.ca

1 Introduction

The problem of enumerating polyominoes is known to be a difficult problem since their introduction by Golomb [2] and so far, only exhaustive generation computer programs are providing numerical answers. Nevertheless, restricted classes of these objects have been successfully enumerated [1, 4].

We focus on the problem of counting polyominoes inscribed in a rectangle of size $b \times h$. By *inscribed*, we mean a polyomino that is included in a rectangle of size $b \times h$ and that has at least one cell touching each side of the rectangle. If we fix b and increase h , the number of inscribed polyominoes satisfies a linear recurrence. The sequences for $b = 2, 3, 4$ are registered on the Online Encyclopedia of Integer Sequences (OEIS) ([A034182](#) [3], [A034184](#) [3] and [A034187](#) [3], respectively). The values for $5 \leq b \leq 12$ and $h = 24 - b$ are also available ([A292357](#) [3]).

The recurrence for $b = 2$ is known ([A034182](#) [3]) and can be proved using simple combinatorial argument.

$$G_2 = \frac{2x^3 + 3x^2 - 2x + 1}{(x-1)(x^2 + 2x - 1)}$$

Where G_b is the generating function for the number of polyominoes in a rectangle of size $b \times h$. Recurrences for $b = 3, 4$ have been discovered empirically but, to the best of our knowledge, no proof seems to be available.

In this extended abstract, we show how we obtained the formulas for $b = 3, 4, 5, 6$ and we design a method for obtaining the formulas for any b . To do so, we adapt methods described in previous works by Zeilberger [4] and by Bousquet-Mélou and Brak [1] in order to build an automaton \mathcal{A}_b recognizing exactly the polyominoes inscribed in a rectangle of fixed width b and any height h .

2 Building \mathcal{A}_b

A polyomino P is a set of edge-connected cells in the square lattice. If P is inscribed in a rectangle of size $b \times h$, it can equivalently be described as a stack of h rows of b cells, where each cell is either selected or not and the selected cells are edge-connected.

Each possible configuration in a row can be encoded by a unique word $u \in \{0, 1\}^b$, $|u|_1 > 0$, a 0 represents an empty cell, a 1 represents a selected cell, and $|u|_1$ denotes the number of 1's occurring in the word u . A stack of height h of such words encodes a unique polyomino [1]. In the next paragraphs, we use the expressions *stack of words* and *stack of rows* interchangeably.

We wish to build an automaton \mathcal{A}_b having the property that, given a stack of words, \mathcal{A}_b accepts the stack if and only if it encodes a valid polyomino. Without loss of generality, we can assume that the

automaton \mathcal{A}_b operates by reading the stack of words from top to bottom. In order to accept only valid stacks, \mathcal{A}_b needs to keep track of the connexity with each of its states. If we extend the alphabet of the words u to $\{0, 1, \dots, \lceil b/2 \rceil\}$, we can label each cell of the row with a 0 if the cell is empty and with the value $i \in \{1, \dots, \lceil b/2 \rceil\}$ if the cell belongs to the i^{th} connected component of the polyomino, where the connected components are labeled with increasing values from left to right.

We also want each state to encode whether the stack read so far has yet touched the left side or the right side of the rectangle. For this purpose, we introduce two boolean variables l and r that have value **T** if the leftmost (resp. rightmost) column has at least one selected cell, and **F** otherwise.

From these conventions, we have that each state is of the form:

$$(w, l, r) \quad \text{with} \quad w = w_1 \cdots w_b \in \{0, 1, \dots, \lceil b/2 \rceil\}^b, \quad l, r \in \{\mathbf{T}, \mathbf{F}\}.$$

However, not every triplet of this form can suitably represent a valid polyomino row-configuration. We introduce the following additional conditions such that only the necessary triplets are kept:

Empty row: If $w = 0^b$, then $l = \mathbf{F}$ and $r = \mathbf{F}$. This row is forbidden in a polyomino but it is convenient to use this encoding as an initial state.

Inscription: If $w_1 \neq 0$, then $l = \mathbf{T}$, and if $w_b \neq 0$, then $r = \mathbf{T}$. This means that P is adjacent to the left (resp. right) side of the bounding rectangle.

Separation: If $w_i \neq 0$, then $w_{i-1}, w_{i+1} \in \{0, w_i\}$. In other words, if the i^{th} cell of the current row is in P , each of its adjacent cells w_{i-1} and w_{i+1} is either empty or belongs to the same connected component, in virtue of the edge-connectedness requirement.

Non-crossing: Let $i, j, k, l \in \{1, 2, \dots, \lceil b/2 \rceil\}$ with $i < k < j < l$. Then the conditions $w_i = w_j$ and $w_k = w_l$ imply $w_i = w_k = w_j = w_l$. This condition comes from the fact that two distinct connected components cannot have crossed earlier in P .

We denote by \mathcal{T}_b the set of triplets (w, l, r) that meet those conditions.

Now, we still have the possibility that two distinct words in \mathcal{T}_b represent the same row-configuration in P . For example the words 10201 and 20102 both represent the state where the first cell is connected above to the fifth cell, the third is in P but disconnected from the first and fifth cells while the second and fourth cells are empty.

To ensure injectivity, we introduce an equivalence relation. Let $w = w_1 w_2 \dots w_b \in A^b$ be a word on the alphabet A . For $a \in A$, we denote by $Pos_a(w)$ the set of all indices i such that $w_i = a$. Then w and w' are called *equivalent*, and we write $w \equiv w'$, if the following two conditions are verified:

- (i) $Pos_0(w) = Pos_0(w')$
- (ii) There exists $\sigma \in \mathfrak{S}_{\lceil b/2 \rceil}$ such that $Pos_i(w) = Pos_{\sigma(i)}(w')$, for $i \in \{1, 2, \dots, \lceil b/2 \rceil\}$.

where $\mathfrak{S}_{\lceil b/2 \rceil}$ is the symmetric group on $\lceil b/2 \rceil$ elements. For each equivalence class of \equiv , we choose as representative the minimum element with respect to the lexicographic order, denoted by $[w]$.

We are now ready to define our automaton. More formally, let $\mathcal{A}_b = (\Sigma_b, Q_b, q_{0,b}, F_b, \delta_b)$, where

- (1) $\Sigma_b = \{u \in \{0, 1\}^b : |u|_1 > 0\}$ is the set of possible words in a stack;
- (2) $Q_b = \{([w], l, r) : [w] \in \mathcal{T}_b / \equiv\}$ is the set of states;
- (3) $q_{0,b} = (0^b, \mathbf{F}, \mathbf{F})$ is the initial state;
- (4) $F_b = \{([w], l, r) \in \mathcal{T}_b : w \in \{0, 1\}^b, l = r = \mathbf{T}\}$ is the set of accepting states and
- (5) $\delta_b : Q_b \times \Sigma_b \rightarrow Q_b$ is the transition function described in the next paragraphs.

Observe that a state $([w], l, r) \in Q_b$ is also element of F_b (i.e. is an accepting state) if and only if it has a single connected component (i.e. $w \in \{0, 1\}^b$) and is inscribed in the rectangle (i.e. $l = r = \mathbf{T}$). The transition function δ_b is defined only on the pairs $(([w], l, r), u)$, with $([w], l, r) \in Q_b$, $u \in \Sigma_b$, such that, for all $a \in \{1, 2, \dots, \lceil b/2 \rceil\}$, $|Pos_a(w)| > 0$ implies $Pos_a(w) \cap Pos_1(u) \neq \emptyset$. This ensures that, in the polyomino P , no connected component is “lost”. Since u represents the configuration of the next row of the polyomino, $Pos_0(u) = Pos_0([w'])$, and

$$\delta_b(([w], l, r), u) = ([w'], l', r'),$$

where w' is obtained by adding a row subject to the following constraints:

Vertical connexity: From left to right, we read u and write a new word x .

If $u_i = 0$ then $x_i = 0$. If $u_i = 1$ and $w_i \neq 0$, then $x_i = w_i$. If $u_i = 1$ and $w_i = 0$, then $x_i = N + 1$ where $N = \max\{w_1, w_2, \dots, w_b, x_1, x_2, \dots, x_{i-1}\}$. This step keeps the connexity between the cells in the next row and the cells in the current row that are directly above them. For the cells in the next row with no cells directly above them in the current row, we create new connected components for each of them. Observe that x does not meet the *separation condition*.

Horizontal connexity: From x , we create w' .

We first obtain every factor of x that is between 0's together with the largest prefix of x that has no 0's and the largest suffix of x that has no 0's. We call these factors the horizontally connected components of the new row. We say that two horizontally connected components c and c' are *directly linked* if they share a letter and *indirectly linked* if there exists a set $E = \{e_i \mid e_i \text{ is a horizontally connected component}\}$ such that $c = e_0$, $c' = e_k$ and e_i is *directly linked* to e_{i+1} for all $0 \leq i \leq n - 1$ (i.e. there exists a chain of *directly linked* connected components that links c to c'). The components c and c' are said to be *linked* if they are either *directly linked* or *indirectly linked*.

The second step in the construction of w' consists in collecting every *linked* horizontally connected components into different sets. In other words, we take the transitive closure of the “is linked to” relation. Each set receives a letter in $\{1, 2, \dots, \lceil b/2 \rceil\}$. The letters $\{1, 2, \dots, \lceil b/2 \rceil\}$ are assigned incrementally in increasing order to the set that contains the left-most horizontally connected component among the sets that have not yet been assigned a letter.

Finally, we obtain w' from x by looking at every horizontally connected component c in x and replacing each letter of c by the letter assigned to the set to which c belongs.

Adjacency: The values l' and r' are obtained by the computations $l' = l \vee u_1$ and $r' = r \vee u_b$.

Example 1. Let

$$w = 10203020104 \quad \text{and} \quad u = 10111011101.$$

The first step yields the word $x = 10253026104$. The horizontally connected components are 1, 253, 261 and 4. The linked horizontally connected components are then joined together, forming the sets $\{1, 253, 261\}$ and $\{4\}$. The letters are associated with each set in order with the mapping

$$1 \rightarrow \{1, 253, 261\}, \quad 2 \rightarrow \{4\}.$$

Finally, w' is obtained by replacing each letter in every horizontally connected components by the assigned letter, yielding $w' = 10111011102$.

Observe that every step in the construction of \mathcal{A}_b is automatic and δ_b is unambiguous. \mathcal{A}_b is thus deterministic.

Theorem 1. *The set of stacks of size h of words recognized by \mathcal{A}_b is in bijection with the set of polyominoes inscribed in a rectangle of size $b \times h$.*

As an example, Figure 1 shows a stack of length 9 recognized by \mathcal{A}_5 and the associated polyomino.

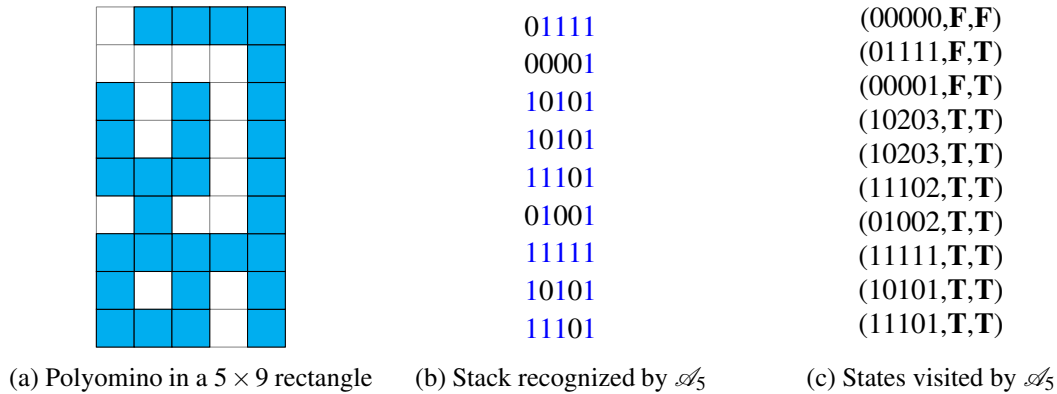


Figure 1: Example of a word recognized by \mathcal{A}_5 and the associated polyomino

3 Generating the automaton \mathcal{A}_b .

We can manually compute the automata \mathcal{A}_2 , \mathcal{A}_3 and, with a bit of courage, \mathcal{A}_4 , \mathcal{A}_5 and \mathcal{A}_6 . Using the Maple software, we were able to compute the generating functions for values of b up to 6. However, for $b \geq 7$, we need to turn to computers with more power, as the number of states grows rapidly. Thankfully, every step in the building of \mathcal{A}_b can be automatized easily.

With these methods, we computed the generating functions G_b for $b = 3, 4, 5, 6$. We also could compute G'_b for $b = 3, 4$ where G'_b is the generating function for the number of polyominoes of area n inscribed in a rectangle of size $b \times h$. The rational expression of those generating functions are of too large degree to include in this paper. Indeed, G_3 is of degree 9, G_4 is of degree 20, G_5 is of degree 49 and G_6 is of degree 112.

The case $b = 7$ was generated but the calculations to produce the generating function still need to be done. We are still working on the implementation to obtain the automata for greater values of b . The challenge comes from the fact that the systems grow exponentially in the number of states. Indeed, the number of states of \mathcal{A}_b can be computed from the formula

$$\# \text{ of states of } \mathcal{A}_b = 1 + \sum_{k=1}^{2^b-1} C_{f(k)} \cdot 2^{\llbracket k \equiv 0 \pmod{2} \rrbracket} \cdot 2^{\llbracket k < 2^{b-1} \rrbracket}, \tag{1}$$

where C_m is the m^{th} Catalan number, $f(k)$ is the number of runs of 1's in the binary expansion of k (A069010 [3]) and $\llbracket p \rrbracket$ is the indicator function taking value 1 if p is true and 0 otherwise. Surprisingly, the sequence of the number of states of \mathcal{A}_b for increasing b is not registered on OEIS. However, the sequence A140662 [3], also counting states in an automaton recognizing polyominoes, has been reported. Avenues for immediate improvement include using a higher efficiency programming language and using a more suitable data type to store the states of the automaton.

b	0	1	2	3	4	5	6	7	8	9	10	11
number of states in \mathcal{A}_b	1	2	6	16	40	99	247	625	1605	4178	11006	29292

Figure 2: Number of states in \mathcal{A}_b for $b = 0, \dots, 11$

References

- [1] Mireille Bousquet-Mélou & Richard Brak (2009): *Exactly solved models of polyominoes and polygons*. In: *Polygons, Polyominoes and Polycubes*, Lecture Notes in Physics, Springer Berlin / Heidelberg, pp. 43–78, doi:10.1007/978-1-4020-9927-4. Available at <https://hal.science/hal-00342024>.
- [2] Solomon W. Golomb (1994): *Polyominoes: Puzzles, Patterns, Problems, and Packings - Revised and Expanded Second Edition*, new edition. 111, Princeton University Press, doi:10.2307/j.ctv10vm1sc. Available at <http://www.jstor.org/stable/j.ctv10vm1sc>.
- [3] OEIS Foundation Inc. (2024): *The On-Line Encyclopedia of Integer Sequences*. Published electronically at <http://oeis.org>.
- [4] Doron Zeilberger (2000): *Symbol-crunching with the transfer-matrix method in order to count skinny physical creatures*. *Integers* 0. Available at <https://sites.math.rutgers.edu/~zeilberg/mamarim/mamarimPDF/animals.pdf>.

LLT Polynomials and Hecke Algebra Traces

Alejandro H. Morales*

LACIM
Université du Québec à Montréal
Montréal QC, Canada

morales_borrero.alejandro@uqam.ca

Mark A. Skandera

Department of Mathematics
Lehigh University
Bethlehem PA, USA

mas906@lehigh.edu

Jiayuan Wang[†]

Department of Mathematics
Lehigh University
Bethlehem PA, USA

jiw922@lehigh.edu

We show that coefficients in unicellular LLT polynomials are evaluations of Hecke algebra traces at Kazhdan–Lusztig basis elements. We express these in terms of traditional trace bases, induction, and Kazhdan–Lusztig R -polynomials.

1 Introduction

The study of proper colorings of a graph G is a fundamental topic in discrete mathematics. Stanley [9] defined the chromatic symmetric function $X_{G,q}$ which is a symmetric function generalization of the chromatic polynomial. This function was generalized by Shareshian and Wachs [8]: for a graph $G = (V, E)$ let $X_{G,q} := \sum_{\kappa} q^{\text{asc}(\kappa)} x_{\kappa(1)} x_{\kappa(2)} \cdots$, where the sum is over all proper colorings $\kappa : V(G) \rightarrow \mathbb{N}$ of G and $\text{asc}(\kappa)$ denotes the number of ascents of κ , pairs (i, j) with $i < j$ such that $\kappa(i) < \kappa(j)$. When $G = \text{inc}(P)$ is an incomparability graph of a unit interval order P , $X_{G,q}$ is a symmetric function. There are important positivity conjectures about $X_{\text{inc}(P),q}$ like the e -positivity conjecture of Stanley–Stembridge–Shareshian–Wachs.

In another context, the functions $X_{\text{inc}(P),q}$ appeared in the study of the space of *diagonal harmonics*. Let $\text{LLT}_{\text{inc}(P),q} := \sum_{\kappa} q^{\text{asc}(\kappa)} x_{\kappa(1)} x_{\kappa(2)} \cdots$, where the sum is over arbitrary vertex colorings of $\text{inc}(P)$. This is also a symmetric function called a *unicellular LLT polynomial*, a special case of a family of symmetric functions introduced by Lascoux–Leclerc–Thibon in 1997 in a different context. These functions $\text{LLT}_{\text{inc}(P),q}$ appear in the *Shuffle conjecture* of diagonal harmonics [6] proved by Carlsson–Mellit [2]. In their proof of the shuffle conjecture, they show that both these symmetric functions are related by a *plethystic substitution*:

$$X_{\text{inc}(P),q}[X] = (q-1)^{-n} \text{LLT}_{\text{inc}(P),q}[(q-1)X],$$

where n is the size of P . From work of Grojnowski and Haiman [5], $\text{LLT}_{\text{inc}(P)}$ are Schur positive and it is an open question to find a combinatorial interpretation for this expansion.

An important basis of the Hecke algebra $H_n(q)$ is the (modified, signless) *Kazhdan–Lusztig basis* defined by $\tilde{C}_w(q) := q^{\frac{\ell(w)}{2}} C'_w(q) = \sum_{u \leq v} P_{u,w}(q) T_w$, where $\{T_w \mid w \in \mathfrak{S}_n\}$ is the natural basis of $H_n(q)$, $P_{v,w}(q)$ are the *Kazhdan–Lusztig polynomials* and \leq denotes the Bruhat order of \mathfrak{S}_n . It is known [3] that the various expansions of the chromatic quasi-symmetric function $X_{\text{inc}(P),q}$ can be viewed as evaluations of traces at $\{\tilde{C}_w(q) \mid w \text{ avoiding } 312\}$, when $P = P(w)$ is a unit interval order corresponding to w ,

$$X_{\text{inc}(P(w)),q} = \sum_{\lambda \vdash n} \varepsilon_q^\lambda(\tilde{C}_w(q)) m_\lambda = \sum_{\lambda \vdash n} \eta_q^\lambda(\tilde{C}_w(q)) f_\lambda = \sum_{\lambda \vdash n} \chi_q^{\lambda^\top}(\tilde{C}_w(q)) s_\lambda = \cdots,$$

*AHM is partially supported by NSF grant DMS-2154019.

[†]JW is partially supported by an AMS-Simons travel grant.

where $\varepsilon_q^\lambda, \eta_q^\lambda, \chi_q^{\lambda^\top}$ are induced sign, induced trivial, and irreducible characters of $H_n(q)$. In this context, the e -positivity conjecture of $X_{\text{inc}(P),q}$ is part of a more general conjecture of Haiman [7] for symmetric functions associated to $\tilde{C}_w(q)$ for any w in the context of immanants. In [1, §11.3], Abreu and Nigro used the plethystic relation to define analogs of unicellular LLT polynomials for all permutations.

The coefficients of various expansions of LLT polynomials can also be viewed as evaluations of traces at $\{\tilde{C}_w(q) \mid w \text{ avoiding } 312\}$.

$$\text{LLT}_{\text{inc}(P),q} = \sum_{\lambda} \varepsilon_{q,\text{LLT}}^\lambda(\tilde{C}_w(q))m_{\lambda},$$

where $\varepsilon_{q,\text{LLT}}^\lambda(\tilde{C}_w(q))$ is a certain LLT-analog of the trace ε_q^λ .

We describe similar analogs of induced trivial characters $\eta_{q,\text{LLT}}^\lambda$ and power sum traces $\psi_{q,\text{LLT}}^\lambda$. The evaluations at $\tilde{C}_w(q)$ were known as expansions of the LLT but now we obtain evaluations at the natural basis T_w , which were not known before.

We also give change of basis equations between $\psi_{q,\text{LLT}}^n, \varepsilon_{q,\text{LLT}}^n$, and $\eta_{q,\text{LLT}}^n$ and known traces that resemble the Cauchy identity of symmetric functions after a principal specialization.

2 Background

Symmetric functions. We mostly use the notation from [10, Ch. 7]. We denote by Λ_n the ring of symmetric functions of degree n and $m_{\lambda}, e_{\lambda}, h_{\lambda}, p_{\lambda}, s_{\lambda}, f_{\lambda}$ denote the *monomial, elementary, complete, power sum, Schur, and forgotten* symmetric functions. Also ω denotes the standard involution in Λ_n .

Hecke algebra and traces. The Hecke algebra $H_n(q)$ is a noncommutative $\mathbb{Z}[q^{\frac{1}{2}}, q^{-\frac{1}{2}}]$ -algebra generated by *natural generators* $\{T_{s_i} \mid 1 \leq i \leq n-1\}$ subject to the relations

$$\begin{aligned} T_{s_i}^2 &= (q-1)T_{s_i} + q, & \text{for } i &= 1, \dots, n-1, \\ T_{s_i}T_{s_j}T_{s_i} &= T_{s_j}T_{s_i}T_{s_j}, & \text{if } |i-j| &= 1, \\ T_{s_i}T_{s_j} &= T_{s_j}T_{s_i}, & \text{if } |i-j| &\geq 2. \end{aligned}$$

Specializing $H_n(q)$ at $q^{\frac{1}{2}} = 1$, we obtain the classical group algebra $\mathbb{Z}[\mathfrak{S}_n]$ of the symmetric group.

Let $\mathcal{T}(H_n(q))$ be the $\mathbb{Z}[q^{\frac{1}{2}}, q^{-\frac{1}{2}}]$ -module of $H_n(q)$ -traces, linear functionals $\theta_q : H_n(q) \rightarrow \mathbb{Z}[q^{\frac{1}{2}}, q^{-\frac{1}{2}}]$ satisfying $\theta_q(DD') = \theta_q(D'D)$ for all $D, D' \in H_n(q)$. For any trace $\theta_q : T_w \mapsto a(q)$ in $\mathcal{T}(H_n(q))$, the $q^{\frac{1}{2}} = 1$ specialization $\theta : w \mapsto a(1)$ belongs to the space $\mathcal{T}(\mathfrak{S}_n) := \mathcal{T}(H_n(1))$ of $\mathbb{Z}[\mathfrak{S}_n]$ -traces from $\mathbb{Z}[\mathfrak{S}_n] \rightarrow \mathbb{Z}$ (\mathfrak{S}_n -class functions). Like the \mathbb{Z} -module Λ_n of homogeneous degree- n symmetric functions, the trace spaces $\mathcal{T}(H_n(q))$ and $\mathcal{T}(\mathfrak{S}_n)$ have dimension equal to the number of *integer partitions* of n , the weakly decreasing positive integer sequences $\lambda = (\lambda_1, \dots, \lambda_r)$ satisfying $\lambda_1 + \dots + \lambda_r = n$.

It can be useful to record trace evaluations in a symmetric generating function. In particular, for $D \in \mathbb{Q}(q) \otimes H_n(q)$, we record induced sign character evaluations by defining

$$Y_q(D) := \sum_{\lambda \vdash n} \varepsilon_q^\lambda(D)m_{\lambda} \in \mathbb{Q}(q) \otimes \Lambda_n. \tag{2.1}$$

This symmetric generating function in fact gives us all of the standard trace evaluations.

Proposition 2.1. *The symmetric function $Y_q(D)$ is equal to*

$$\sum_{\lambda \vdash n} \eta_q^\lambda(D)f_{\lambda} = \sum_{\lambda \vdash n} \frac{\text{sgn}(\lambda)\psi_q^\lambda(D)}{z_{\lambda}}p_{\lambda} = \sum_{\lambda \vdash n} \chi_q^{\lambda^\top}(D)s_{\lambda} = \sum_{\lambda \vdash n} \phi_q^\lambda(D)e_{\lambda} = \sum_{\lambda \vdash n} \gamma_q^\lambda(D)h_{\lambda},$$

where $\text{sgn}(\lambda) := (-1)^{n-\ell(\lambda)}$; equivalently, $\omega Y_q(D)$ is equal to

$$\sum_{\lambda \vdash n} \varepsilon_q^\lambda(D) f_\lambda = \sum_{\lambda \vdash n} \eta_q^\lambda(D) m_\lambda = \sum_{\lambda \vdash n} \frac{\psi_q^\lambda(D)}{z_\lambda} p_\lambda = \sum_{\lambda \vdash n} \chi_q^\lambda(D) s_\lambda = \sum_{\lambda \vdash n} \phi_q^\lambda(D) h_\lambda = \sum_{\lambda \vdash n} \gamma_q^\lambda(D) e_\lambda.$$

Quantum matrix bialgebra and immanants. An important computational tool in the evaluation of $H_n(q)$ -traces is the *quantum matrix bialgebra* $\mathcal{A}_n(q)$, the noncommutative ring generated as a $\mathbb{Z}[q^{\frac{1}{2}}, q^{-\frac{1}{2}}]$ -algebra by the n^2 variables $t = (t_{i,j})_{i,j \in [n]}$ subject to relations

$$\begin{aligned} t_{i,\ell} t_{i,k} &= q^{\frac{1}{2}} t_{i,k} t_{i,\ell}, & t_{j,k} t_{i,\ell} &= t_{i,\ell} t_{j,k} \\ t_{j,k} t_{i,k} &= q^{\frac{1}{2}} t_{i,k} t_{j,k} & t_{j,\ell} t_{i,k} &= t_{i,k} t_{j,\ell} + (q^{\frac{1}{2}} - q^{-\frac{1}{2}}) t_{i,\ell} t_{j,k}, \end{aligned} \tag{2.2}$$

for all indices $1 \leq i < j \leq n$ and $1 \leq k < \ell \leq n$. As a $\mathbb{Z}[q^{\frac{1}{2}}, q^{-\frac{1}{2}}]$ -module, $\mathcal{A}_n(q)$ has a natural basis of monomials $t_{\ell_1, m_1} \cdots t_{\ell_r, m_r}$ in which index pairs appear in lexicographic order. The relations (2.2) allow one to express other monomials in terms of this natural basis.

To state immanant generating functions for $H_n(q)$ -traces, it will be convenient to express monomials in $\mathcal{A}_n(q)$ as follows. Given $u = u_1 \cdots u_n, v = v_1 \cdots v_n \in \mathfrak{S}_n$, define

$$t^{u,v} := t_{u_1, v_1} \cdots t_{u_n, v_n}.$$

For any linear function $\theta_q : H_n(q) \rightarrow \mathbb{Z}[q^{\frac{1}{2}}, q^{-\frac{1}{2}}]$, define the θ_q -immanant in $\mathcal{A}_n(q)$ to be

$$\text{Imm}_{\theta_q}(t) = \sum_{w \in \mathfrak{S}_n} q^{-\frac{\ell(w)}{2}} \theta_q(T_w) t^{e,w}.$$

Proposition 2.2. *Given Hecke algebra traces*

$$\theta_1 \in \mathcal{T}(H_k(q)), \quad \theta_2 \in \mathcal{T}(H_{n-k}(q)), \quad \theta = (\theta_1 \otimes \theta_2) \uparrow_{H_k(q) \times H_{n-k}(q)}^{H_n(q)} \in \mathcal{T}(H_n(q)),$$

we have

$$\text{Imm}_\theta(t) = \sum_{I \text{ where } |I|=k} \text{Imm}_{\theta_1}(t_{I,I}) \text{Imm}_{\theta_2}(t_{\bar{I},\bar{I}}).$$

Since Hecke algebra traces are determined by the values on minimum length representatives (see [4, Cor. 8.2.6]), then the following result will be useful.

Lemma 2.3. *Let $w \in \mathfrak{S}_n$ be of minimum length in its conjugacy class, then w avoids the patterns 3412 and 4231. Furthermore, each $v \leq w$ also avoids the patterns 3412 and 4231, and also is of minimum length in its conjugacy class.*

3 Plethystically defined characters

Suppose that a certain plethystic substitution transforms symmetric functions written $\{Y_q(D) \mid D \in H_n(q)\}$ into symmetric functions $\{Z_q(D) \mid D \in H_n(q)\}$, i.e.

$$Z_q(D) := r(q) Y_q(D) [s(q) X] \tag{3.1}$$

for some rational functions $r(q)$ and $s(q)$. This substitution yields a transformation of $H_n(q)$ traces $\theta_q \mapsto \theta_{q,Z}$ as well. Namely, we define $\varepsilon_{q,Z}^\lambda$ to be the $H_n(q)$ -trace that maps D to the coefficient of m_λ in the monomial expansion of $Z_q(D)$,

$$Z_q(D) = \sum_{\lambda \vdash n} \varepsilon_{q,Z}^\lambda(D) m_\lambda. \tag{3.2}$$

Then we extend linearly over $\mathbb{Z}[q^{\frac{1}{2}}, q^{-\frac{1}{2}}]$, mapping

$$\theta_q = \sum_{\lambda \vdash n} b_\lambda \varepsilon_q^\lambda \mapsto \theta_{q,Z} := \sum_{\lambda \vdash n} b_\lambda \varepsilon_{q,Z}^\lambda. \tag{3.3}$$

Observation 3.1. *The symmetric function $Z_q(D)$ is equal to*

$$\sum_{\lambda \vdash n} \eta_{q,Z}^\lambda(D) f_\lambda = \sum_{\lambda \vdash n} \frac{\text{sgn}(\lambda) \psi_{q,Z}^\lambda(D)}{z_\lambda} p_\lambda = \sum_{\lambda \vdash n} \chi_{q,Z}^{\lambda^\top}(D) s_\lambda = \sum_{\lambda \vdash n} \phi_{q,Z}^\lambda(D) e_\lambda = \sum_{\lambda \vdash n} \gamma_{q,Z}^\lambda(D) h_\lambda.$$

Proposition 3.2. *For any plethystically defined map $Y \mapsto Z$ (3.1) of symmetric functions, if θ_q is a trace function, then so is $\theta_{q,Z}$.*

Furthermore by (3.3), the change of basis matrix which relates two symmetric function bases (and necessarily the traces which correspond by the Frobenius map), also relates the Z -analogs of those traces.

For example when $Z = \text{LLT}_{\text{inc}(P),q}$ and $w = w(P)$ avoiding the patterns 3412 and 4231, we have

$$\varepsilon_{q,\text{LLT}}^n(\tilde{C}_w(q)) = 1. \tag{3.4}$$

This is because by Prop. 3.4 there is only one column-strict Young tableau U of shape 1^n , the tableau consisting one column with entries $1, 2, \dots, n$ in order, with $\text{INV}_P(U) = 0$ where $P = P(w)$.

It is possible to describe LLT analogs of induced sign characters, induced trivial characters, and power sum traces in terms of character induction.

LLT analogs of power sum traces The LLT analogs of the power sum trace can be expressed simply in terms of the ordinary power sum trace.

Proposition 3.3. *We have*

$$\psi_{q,\text{LLT}}^\lambda = (q-1)^n \prod_i \frac{1}{q^{\lambda_i} - 1} \cdot \psi_q^\lambda.$$

Proof. Omitted. □

LLT analogs of induced sign characters and induced trivial characters The evaluations of $\varepsilon_{q,\text{LLT}}^\lambda$ and $\eta_{q,\text{LLT}}^\lambda$ at $\{\tilde{C}_w(q) \mid w \text{ avoiding } 312\}$ have simple combinatorial interpretations.

Proposition 3.4. *Fix $w \in \mathfrak{S}_n$ avoiding the pattern 312, and let $P = P(w)$. For all $\lambda \vdash n$ we have*

$$\varepsilon_{q,\text{LLT}}^\lambda(\tilde{C}_w(q)) = \sum_U q^{\text{INV}_P(U)},$$

where the sum is over all column-strict Young tableaux U of shape λ^\top , and

$$\eta_{q,\text{LLT}}^\lambda(\tilde{C}_w(q)) = \sum_U q^{\text{INV}_P((U_1 \circ \dots \circ U_r)^R)},$$

where the sum is over all row-strict Young tableaux U of shape λ and $(U_1 \circ \dots \circ U_r)^R$ is the reversal of the concatenation of rows in U .

Proof. Omitted. □

This leads to the following generating functions for $\varepsilon_{q,\text{LLT}}^\lambda$ and $\eta_{q,\text{LLT}}^\lambda$.

Theorem 3.5. For $\lambda = (\lambda_1, \dots, \lambda_r) \vdash n$ we have

$$\text{Imm}_{\varepsilon_{q,\text{LLT}}^\lambda}(t) = \sum_{(I_1, \dots, I_r)} (t_{I_1, I_1})^{e,e} \cdots (t_{I_r, I_r})^{e,e}, \quad (3.5)$$

where the sum is over all ordered set partitions (I_1, \dots, I_r) of type λ , and e is the identity permutation in the appropriate subgroup of \mathfrak{S}_n . And

$$\text{Imm}_{\eta_{q,\text{LLT}}^\lambda}(t) = \sum_{(I_1, \dots, I_r)} (t_{I_r, I_r})^{w_0, w_0} \cdots (t_{I_1, I_1})^{w_0, w_0}, \quad (3.6)$$

where the sum is over all ordered set partitions of type λ and w_0 is the longest permutation in the appropriate subgroup of \mathfrak{S}_n .

Proof. Omitted. □

Equivalently, we have the following.

Theorem 3.6. We have

$$\varepsilon_{q,\text{LLT}}^\lambda = (\varepsilon_{q,\text{LLT}}^{\lambda_1} \otimes \cdots \otimes \varepsilon_{q,\text{LLT}}^{\lambda_r}) \uparrow_{H_\lambda(q)}^{H_n(q)}, \quad \text{where } \varepsilon_{q,\text{LLT}}^n(T_w) = \begin{cases} 1 & \text{if } w = e, \\ 0 & \text{otherwise.} \end{cases}$$

And

$$\eta_{q,\text{LLT}}^\lambda = (\eta_{q,\text{LLT}}^{\lambda_1} \otimes \cdots \otimes \eta_{q,\text{LLT}}^{\lambda_r}) \uparrow_{H_\lambda(q)}^{H_n(q)}, \quad \text{where } \eta_{q,\text{LLT}}^n(T_w) = R_{e,w}(q).$$

Proof. By Theorem 3.5 and Proposition 2.2, we have $\text{Imm}_{\varepsilon_{q,\text{LLT}}^n}(t) = t_{1,1} \cdots t_{n,n}$.

Also, we have $t^{w_0, w_0} = \sum_{w \in \mathfrak{S}_n} R_{e,w}(q) q^{-\frac{\ell(w)}{2}} t^{e,w}$, and $\eta_{q,\text{LLT}}^n(\tilde{T}_w) = R_{e,w}(q) q^{-\frac{\ell(w)}{2}}$. □

We can express $\varepsilon_{q,\text{LLT}}^n$ and $\eta_{q,\text{LLT}}^n$ in terms of ordinary $H_n(q)$ -characters and principal specialization of symmetric functions.

Corollary 3.7.

$$\begin{aligned} \frac{\varepsilon_{q,\text{LLT}}^n}{(1-q)^n} &= \sum_{\lambda} \frac{1}{z_\lambda} \prod_i \frac{1}{1-q^{\lambda_i}} \psi_q^\lambda = \sum_{\lambda} \frac{q^{b(\lambda)}}{\prod_{u \in \lambda} (1-q^{h(u)})} \chi_q^\lambda \\ &= \sum_{\lambda} \prod_i \frac{1}{(1-q)(1-q^2) \cdots (1-q^{\lambda_i})} \phi_q^\lambda = \sum_{\lambda} \prod_i \frac{q^{\binom{\lambda_i}{2}}}{(1-q)(1-q^2) \cdots (1-q^{\lambda_i})} \gamma_q^\lambda, \end{aligned}$$

and

$$\begin{aligned} \frac{\eta_{q,\text{LLT}}^n}{(1-q)^n} &= \sum_{\lambda} \frac{\text{sgn}(\lambda)}{z_\lambda} \prod_i \frac{1}{1-q^{\lambda_i}} \psi_q^\lambda = \sum_{\lambda} \frac{q^{b(\lambda')}}{\prod_{u \in \lambda'} (1-q^{h(u)})} \chi_q^\lambda, \\ &= \sum_{\lambda} \prod_i \frac{q^{\binom{\lambda_i}{2}}}{(1-q)(1-q^2) \cdots (1-q^{\lambda_i})} \phi_q^\lambda = \sum_{\lambda} \prod_i \frac{1}{(1-q)(1-q^2) \cdots (1-q^{\lambda_i})} \gamma_q^\lambda, \end{aligned}$$

where $h(u)$ is the hook-length of u and $b(\lambda) = \sum_i (i-1) \cdot \lambda_i = \sum_i \binom{\lambda_i}{2}$.

Proof. Omitted. □

References

- [1] Alex Abreu & Antonio Nigro (2022): *A geometric approach to characters of Hecke algebras*. arXiv:2205.14835.
- [2] Erik Carlsson & Anton Mellit (2018): *A proof of the shuffle conjecture*. *Journal of the American Mathematical Society* 31(3), pp. 661–697, doi:10.1090/jams/893.
- [3] Samuel Clearman, Matthew Hyatt, Brittany Shelton & Mark Skandera (2016): *Evaluations of Hecke algebra traces at Kazhdan-Lusztig basis elements*. *Electron. J. Combin.* 23(2), doi:10.37236/5021. Paper 2.7, 56 pages.
- [4] Meinolf Geck & Götz Pfeiffer (2000): *Characters of finite Coxeter groups and Iwahori-Hecke algebras*. *London Mathematical Society Monographs. New Series* 21, The Clarendon Press Oxford University Press, New York, doi:10.1093/oso/9780198502500.001.0001.
- [5] Ian Grojnowski & Mark Haiman (2006): *Affine Hecke algebras and positivity of LLT and Macdonald polynomials*.
- [6] J. Haglund, M. Haiman, N. Loehr, J. Remmel & A. Ulyanov (2005): *A combinatorial formula for the character of the diagonal coinvariants*. *Duke Mathematical Journal* 126(2), pp. 195–232, doi:10.1215/S0012-7094-04-12621-1.
- [7] M. Haiman (1993): *Hecke algebra characters and immanant conjectures*. *J. Amer. Math. Soc.* 6(3), pp. 569–595, doi:10.1090/S0894-0347-1993-1186961-9.
- [8] John Shareshian & Michelle L. Wachs (2016): *Chromatic quasisymmetric functions*. *Adv. Math.* 295, pp. 497–551, doi:10.1016/j.aim.2015.12.018.
- [9] R. Stanley (1995): *A symmetric function generalization of the chromatic polynomial of a graph*. *Adv. Math.* 111, pp. 166–194, doi:10.1006/aima.1995.1020.
- [10] R. Stanley (1999): *Enumerative Combinatorics*. 2, Cambridge University Press, Cambridge, doi:10.1017/CB09780511609589.

HBS Tilings Extended: State of the Art and Novel Observations

Carole Porrier

Université du Québec à Montréal, Canada

Université Sorbonne Paris Nord, France

Penrose tilings are the most famous aperiodic tilings since Gardner described them [2], and they have been studied extensively [3, 11, 1]. It is thus surprising that one can still discover something new about them, even if it is not completely new. Indeed, patterns composed with hexagons H , boats B and stars S (HBS tiles, Fig. 1a) were soon exhibited and aroused interest among physicists. As serendipity works, we rediscovered HBS shapes but with different decorations (Figure 1b) and forcing rules, while working on a combinatorial optimization problem on graphs defined by kites-and-darts Penrose tilings (type P2), as described in [9]. Additionally, we distinguished three types of P2-stars depending on their surrounding, and as we labeled the HBS-vertices according to these types it appeared that most vertices of a given shape always had the same label, with an exception for the star. We thus called *Star tileset* the five shapes with labeled vertices, before knowing about the existence of HBS tiles – even though they first appeared in Henley’s 1986 paper [5].

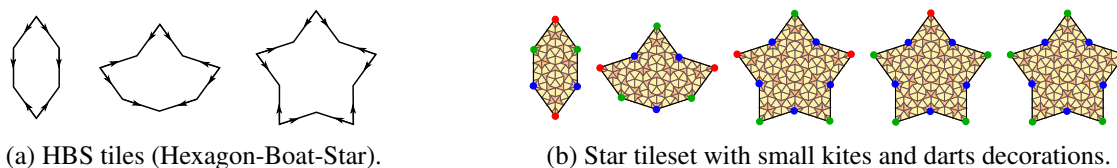


Figure 1: Usual HBS tiles and their new, enriched version.

As we found it difficult to know whether some of our findings had already been published and if so, including which information, it seemed a state of the art would be useful to us and might be for others too. But most of all, we have new findings which we rely on for an article to come [10].

1 Penrose tilings and mutual local derivability

A *tiling* of \mathbb{R}^2 is a countable family of non-empty closed sets $(T_i)_{i \in I}$ called *tiles* such that $\bigcup_{i \in I} T_i = \mathbb{R}^2$ and $\overset{\circ}{T}_i \cap \overset{\circ}{T}_j = \emptyset$ for all $i \neq j$ in I . The *prototiles* of a tiling are the equivalence classes of its tiles up to congruence (in the present context). A set of prototiles is called a *tileset*, and oftentimes many tilings can be composed with copies of the same prototiles. The three types of Penrose tilings, denoted P1, P2 and P3 in [3], were described by Roger Penrose himself [8]. The corresponding tilesets are shown in Figure 2. Uncountably many tilings can be composed with each, but none of them is periodic: they have no translations among their symmetries. The tiles must be arranged according to specific assembly rules, for instance using notches and bumps to assemble the tiles like puzzle pieces (P1 tiles, Fig. 2a) or markings on the tiles which can be of different kinds. For P2, we use two colors which must match on the corners of the tiles (Fig. 2b), forming a full black or blank disk at each vertex of a tiling. Such a marking would not be sufficient for P3 tiles so we use arrows on the edges (Fig. 2c), which must superimpose exactly

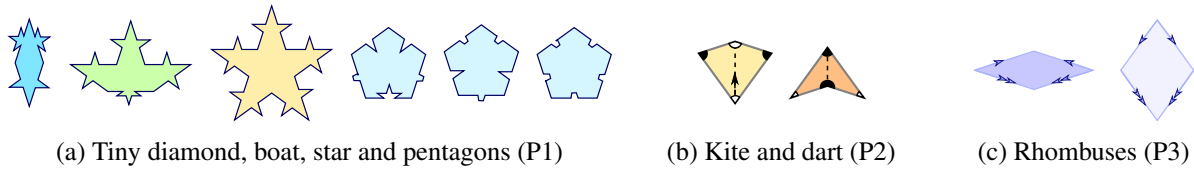


Figure 2: Penrose tilesets.

(same number of arrows, in the same direction). This marking is also the most convenient in this article, in relation with HBS tiles. For the same further purposes, an additional arrow is added on the diagonal of the kite, which corresponds to the single arrow on the edges of rhombuses when a P2 tiling is derived from a P3 tiling. The sides of rhombuses have length 1, which is also the length of the kite's diagonal. Hence for kites and darts the longer side has length 1 and the shorter φ^{-1} where $\varphi = (1 + \sqrt{5})/2$ is the golden ratio. There are only seven vertex configurations, that is seven ways in which kites and darts can be arranged around a vertex of the tiling (Figure 3).

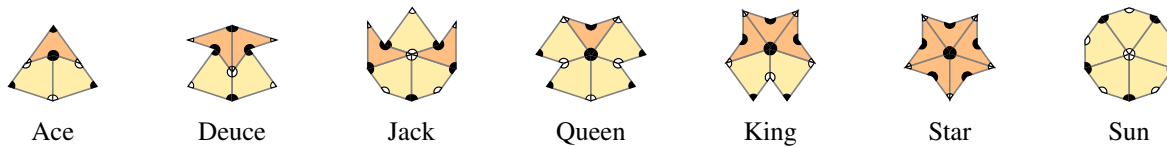


Figure 3: The seven vertex configurations in a Penrose tiling by kites and darts.

Any Penrose tiling (of any type) can be composed or decomposed into another Penrose tiling, of the same or another type, using a local mapping: those tilings are *mutually locally derivable (MLD)*. For instance, Penrose rhombs can be decomposed into kites and darts, which can in turn be decomposed into smaller rhombs. An *inflation* (in the case of Penrose tilings) is a decomposition into tiles of the same type as the original ones, followed with a $\varphi : 1$ scaling, so that the new tiles have the same shapes and sizes as the original ones.

2 HBS tilings summarized

HBS tilings are MLD with Penrose tilings of the three types. We first illustrate the local mappings to and from HBS and Penrose tilings. From HBS to Penrose, simply decorate HBS tiles with (parts of) Penrose tiles of the chosen type, always in the same way. When the Penrose rhombs are marked with arrows as in Figure 2c, each HBS tile is obtained by composing the rhombs as explained in Figure 4a. HBS tiles are a composition of darts and half-kites as described in Figure 4b, with edges of length 1. With the stars and boats, the resemblance between HBS and P1 tilings is obvious, but the local mapping will be presented further, along with a few additional observations.

Henley's paper was cited more than 300 times, but mostly by physicists who experimented different arrangements of atoms based on decorations on HBS shapes – and not always with the same assembly rules. Steurer authored a consistent survey of structure research in quasicrystals [13] comparing such studies, followed with a book on quasicrystals with Deloudi including a lot more theoretical content [15]. In particular, Fig. 1.7 page 24 illustrates the decorations of HBS tiles by Ammann line segments as in Figure 5a and give the relative vertex frequencies of P3 tilings, including the configurations which transform into HBS tiles. The ratio of hexagons to boats to stars is $\sqrt{5}\varphi : \sqrt{5} : 1$, so that the frequencies

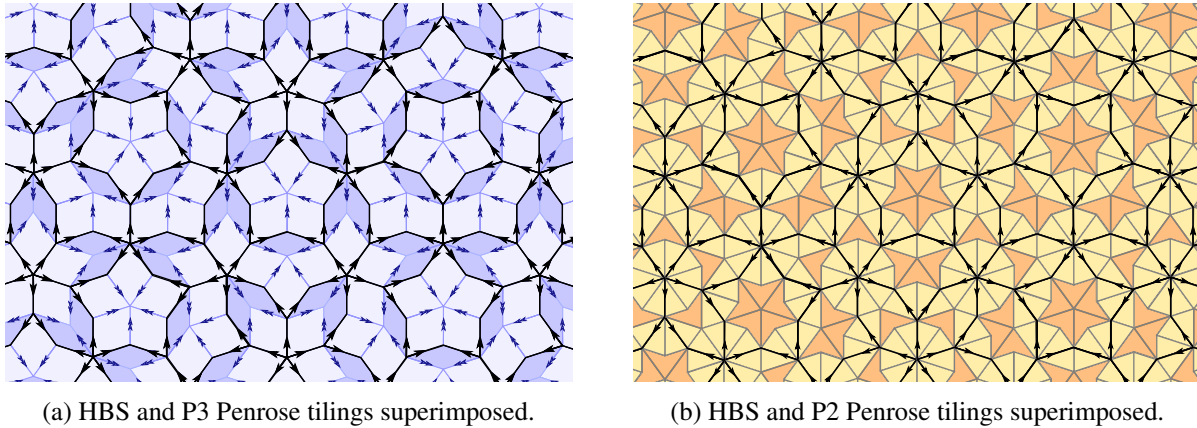


Figure 4: (a) From a P3 tiling, remove all edges with a double arrow and the vertices they point to. (b) From a P2 tiling, trace the axis of symmetry of each kite, with an arrow pointing to the wide angle, then erase all edges of kites and darts.

of the tiles, as computed by Olamy and Kléman [7], are

$$f_H = 7 - 4\varphi = \sqrt{5}\varphi^{-3} \simeq 52,8\% , \quad f_B = 7\varphi - 11 = \sqrt{5}\varphi^{-4} \simeq 32,6\% , \quad f_S = 5 - 3\varphi = \varphi^{-4} \simeq 14,6\% .$$

Gummelt [4] also gives the φ^2 -composition of HBS tilings, drawn in Figure 5b with the arrows. She came up with a decagon covering model, equivalent to the HBS tiling model, as did Lück earlier [6]. Their models were recently compared by Steurer [14].

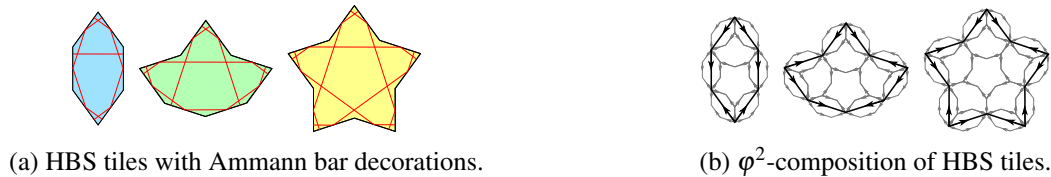


Figure 5: HBS tiles with (a) Ammann segments (arrows are omitted) and (b) their φ^2 -decomposition.

3 The Star tileset

The tileset in Figure 1b yields the same tilings as the HBS tiles once the decorations are removed. It was derived from P2 tilings, but not in the same way as in Figure 4b. The construction is still quite simple and in addition to the φ^2 -composition mentioned above, we have a φ -composition, which is what interested us for the construction in [10]. Vertex colors are strongly related to vertex configurations in P2, hence to vertex configurations in HBS tilings when you look at the compositions.

As can be seen in Figure 4b and considering the local rules of P2 tilings, the shortest distance between the centers of two stars¹ is $3 + 2\varphi^{-1} = \varphi^3$, which occurs when two HBS-stars are incident to the same HBS-edge (or kite of the underlying PT). If you join the centers of the stars, the HBS shapes appear again but larger, composed with many kites (or half-kites) and darts as in Figure 6. The vertices

¹“Star” vertex configurations in a P2 tiling or stars in a HBS tiling.

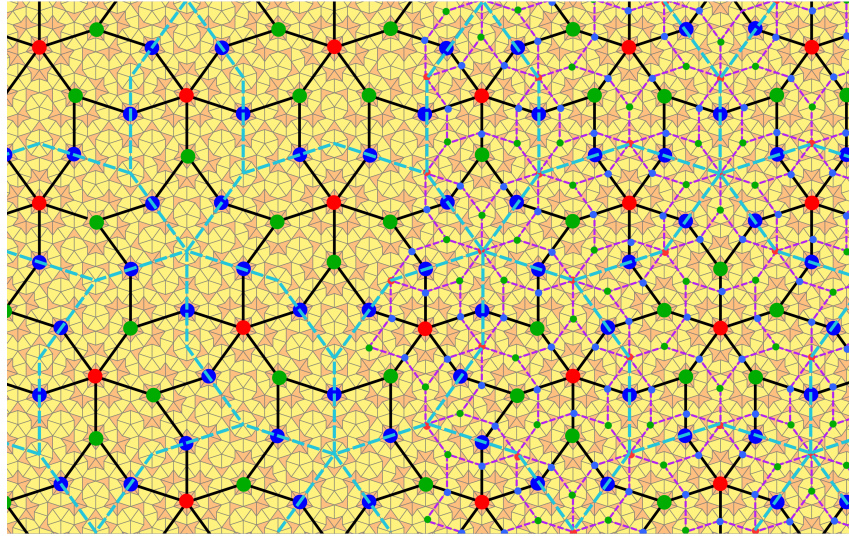


Figure 6: P2 and Star tilings superimposed.

are colored according to their degree, which corresponds to the type of P2-star they lie on: the star can be tangent to either 0 (red), 1 (green) or 2 (blue) suns. If you put on each edge an arrow pointing to the greatest number (among the endpoints), you get exactly the HBS tiles as in Figure 1a. Yet here there is more information encoded in the vertices, hence more local constraints. The hexagons and boats all have exactly the same decorations but the stars come in three versions in what we call the Star tileset.

Obviously, the Ammann bar decorations of HBS tiles are still valid for the same shapes in the Star tileset. Each vertex of the star tiling lies inside a small polygon formed by n Ammann bars. The label of the vertex is then $5 - n$ (or the corresponding color). Since the vertices of the HBS tiling are the stars of the P2 tiling, and a star is obtained by inflating a sun, joining the suns which are at distance $2 + \varphi^{-1} = \varphi^2$ (the minimum) as in Figure 6 (magenta dashed lines) yields the same HBS tiling as first inflating once the P2 tiling and then joining the stars. The vertex colors are then given by the number of queen configurations intersecting with the sun. The resulting φ -decomposition of each tile is resumed in Figure 7. Apply it twice to get the φ^2 -decomposition mentioned above. Now if you apply the φ -

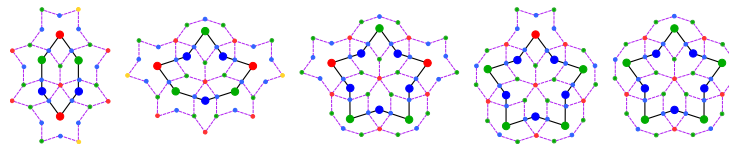


Figure 7: Star tileset decomposition.

decomposition a third time and decorate the smaller shapes as in Figure 4b (usual decoration of HBS tiles with half-kites and darts), the initial Star tiles will have the same decorations as in Figure 1b. Equivalently, from an HBS tiling with the tiles decorated as in Figure 4b you can keep the HBS shapes as is, apply three φ -decompositions to kites and darts, and then appropriately place the vertex colors on hexagons and boats. Actually, you can even place the colors from the start, considering how vertex configurations in P2 tilings substitute. As for the substitution with P1, the label of a pentagon is the number of HBS-arrows pointing to it. Thus each type of pentagon from the original P1 tileset can simply

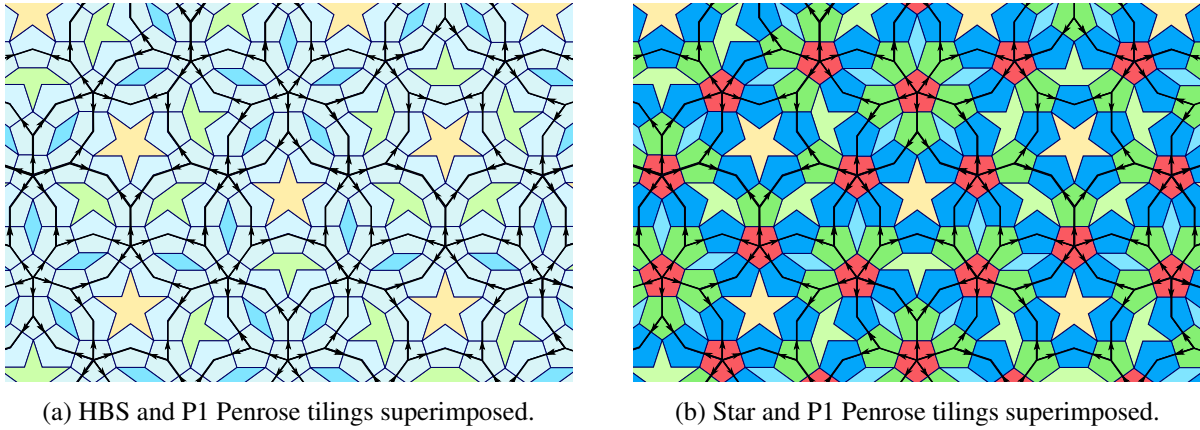


Figure 8: Each P1-star is inside a HBS-star, each P1-boat inside a HBS-boat, and each P1-diamond inside a HBS-hexagon, while all pentagons are decorated with the edges of HBS tiles. In the Star tiling, pentagons are colored according to their type (Figure 9b).

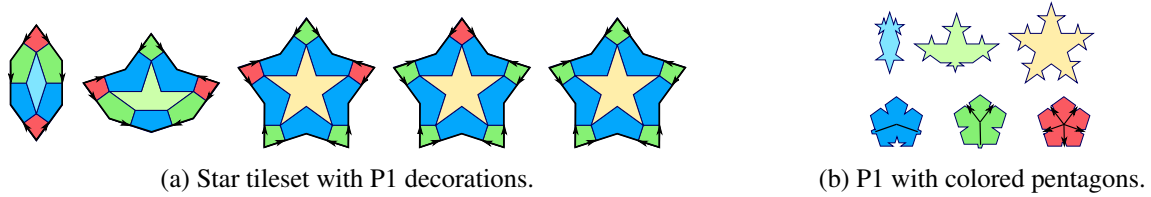


Figure 9: Duality between P1 and Star tilesets.

get the corresponding color, as illustrated in Figures 8b and 9, and if we add the arrows on P1 tiles then the corresponding HBS tiling appears on any P1 tiling.

HBS vertex configurations are given in Figure 10. The main difference between HBS and Star tilings is the amount of information which can be deduced from a local configuration. As you can see in Figure 7, each vertex color in a Star tiling decomposes into an hexagon, a boat or a star according to its color. Hence the ratio of blue to green to red is the same as hexagons to boats to stars, that is $\sqrt{5}\varphi : \sqrt{5} : 1$. Since the frequency of stars in HBS tilings is $f_S = \varphi^{-4}$, we have

$$f_H = \sqrt{5}\varphi^{-3}, \quad f_B = \sqrt{5}\varphi^{-4}, \quad f_2 = \sqrt{5}\varphi^{-7} \simeq 7,7\%, \quad f_1 = \sqrt{5}\varphi^{-8} \simeq 4,76\%, \quad f_0 = \varphi^{-8} \simeq 2,13\%$$

where f_i is the frequency of the star S_i with i red vertices. These frequencies can also be computed using those of vertex configurations in Penrose tilings.

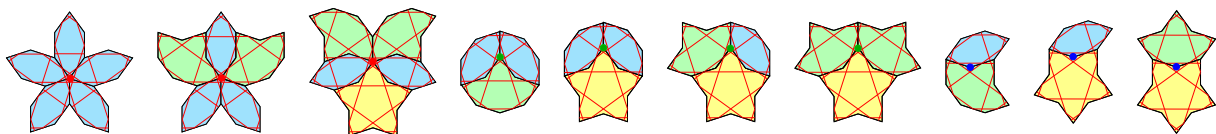


Figure 10: Vertex neighborhoods in HBS and Star tilings, with different possible colors on some vertices of the stars. When stars, boats and hexagons are decomposed, we obtain one of those centered on a red vertex (from left to right), which we call respectively bellflower, orchid and pansy.

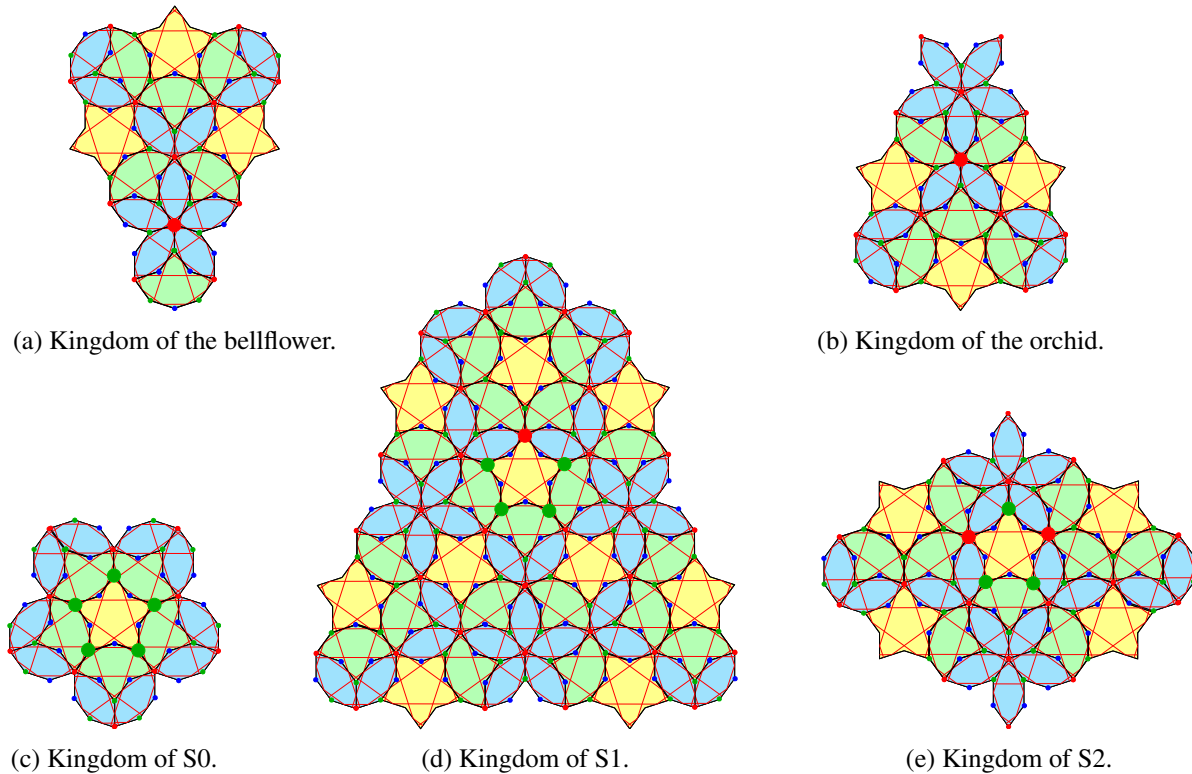


Figure 12: Kingdoms of the bellflower, the orchid and the 3 stars.

As in Penrose tilings, each vertex configuration can force a whole set of tiles (in the tiling) which is called *empire*. This means that anytime a given vertex configuration appears in any tiling of the same type, all the tiles forming the empire are placed exactly in the same way relatively to it. Since several empires are disconnected and can even contain infinitely many tiles, we only represent here the largest connected component, which includes the vertex configuration. We call it *kingdom* (or *local empire*). For the bellflower and the orchid, since they contain no stars, vertex colors could be omitted (as in HBS tilings), but when a vertex configuration includes a star, the colors have a significant impact, as you can see in Figure 12. In particular, the star S1 forces a star S0 above and two stars S2 below. These three kingdoms resemble those of respectively the star, the king and the queen in P2 tilings.

4 Gemstones tileset

The Star tileset can be modified to get convex tiles with only two labels at their vertices, observing that a blue vertex of an hexagon, a boat or a star is actually not a vertex of the Star tiling. Thus we obtain the *Gemstones* tileset in Figure 13. On Figure 15, one can see how HBS tiles are deformed to get gemstones. Substitution from P3 is also quite simple. Trace the long diagonal of each thin rhomb. For each Q configuration (hexagon), compose the fat rhomb with its two adjacent half thin rhombs. Everywhere else just erase all edges of the P3 tiling. Conversely, Gemstones are easily decorated with rhombuses (including arrows or other matching rules). As for P2, the suns and jacks are vertices of Gemstones: trace a (short) segment between those vertices whenever they share a kite, and a (long) segment when they are separated by two kites which share a short edge.

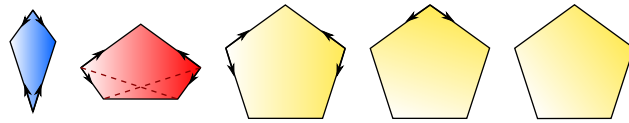


Figure 13: Gemstones tileset: sapphire, ruby and topazes. Edges have length 1 and $2 \sin \frac{2\pi}{5}$. The intersection point of the dashed lines is the “center” of the ruby for the φ -composition in Figure 14.

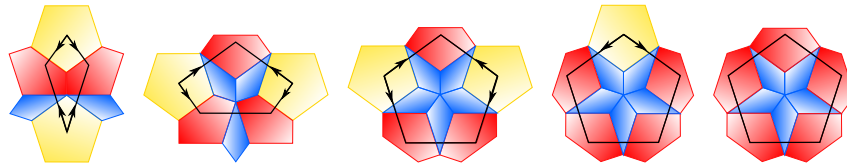


Figure 14: φ -decomposition of Gemstones.

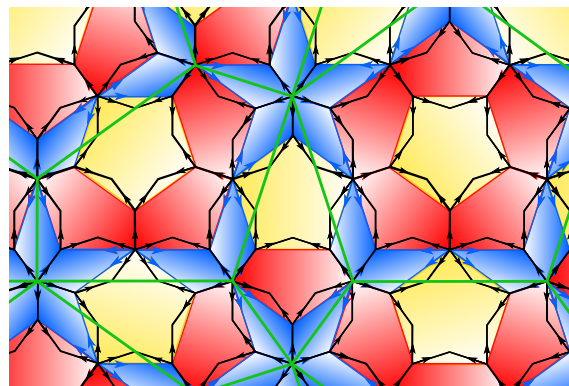


Figure 15: Gemstones and HBS tilings superimposed, along with the φ^2 -composition in green.

References

- [1] Michael Baake & Uwe Grimm (2013): *Aperiodic Order. Vol 1. A Mathematical Invitation. Encyclopedia of Mathematics and its Applications* 149, Cambridge University Press, Cambridge. Available at <http://oro.open.ac.uk/38933/>.
- [2] Martin Gardner (1977): *MATHEMATICAL GAMES*. *Scientific American* 236(1), pp. 110–121, doi:10.1038/scientificamerican0177-110.
- [3] Branko Grünbaum & G. C. Shephard (1987): *Tilings and Patterns*. W. H. Freeman and Company, New York. ISBN 0-7167-1193-1.
- [4] Petra Gummelt (2006): *Decagon covering model and equivalent HBS-tiling model*. *Zeitschrift für Kristallographie - Crystalline Materials* 221(8), pp. 582–588, doi:10.1524/zkri.2006.221.8.582.
- [5] Christopher L. Henley (1986): *Sphere packings and local environments in Penrose tilings*. *Phys. Rev. B* 34, pp. 797–816, doi:10.1103/PhysRevB.34.797. Reprinted in [12].
- [6] Reinhard Lück (1990): *Penrose sublattices*. *Journal of Non-Crystalline Solids* 117-118, pp. 832–835, doi:10.1016/0022-3093(90)90657-8.
- [7] Olamy, Z. & Kléman, M. (1989): *A two dimensional aperiodic dense tiling*. *J. Phys. France* 50(1), pp. 19–33, doi:10.1051/jphys:0198900500101900.
- [8] Roger Penrose (1978): *Pentaplexity*. *Math. Intelligencer* 2(1), pp. 32–37, doi:10.1007/BF03024384.

- [9] Carole Porrier & Alexandre Blondin Massé (2020): *The Leaf Function of Graphs Associated with Penrose Tilings*. *International Journal of Graph Computing* 1, pp. 1–24, doi:10.35708/GC1868-126721.
- [10] Carole Porrier, Alain Goupil & Alexandre Blondin Massé (2023): *The Leaf Function of Penrose P2 Graphs*. arXiv:2312.08262.
- [11] M. Senechal (1995): *Quasicrystals and Geometry*. Cambridge University Press. ISBN 0521372593.
- [12] Paul J Steinhardt & Stellan Ostlund (1987): *The Physics of Quasicrystals*. World Scientific, doi:10.1142/0391. Collection of reprints.
- [13] Walter Steurer (2004): *Twenty years of structure research on quasicrystals. Part I. Pentagonal, octagonal, decagonal and dodecagonal quasicrystals*. *Zeitschrift für Kristallographie - Crystalline Materials* 219(7), pp. 391–446, doi:10.1524/zkri.219.7.391.35643.
- [14] Walter Steurer (2021): *Gummelt versus Lück decagon covering and beyond. Implications for decagonal quasicrystals*. *Acta Crystallographica Section A* 77(1), pp. 36–41, doi:10.1107/S2053273320015181.
- [15] Walter Steurer & Sofia Deloudi (2009): *Crystallography of Quasicrystals: Concepts, Methods and Structures*, 1 edition. Springer Series in Materials Science №126, Springer, doi:10.1007/978-3-642-01899-2.

Counting Colored Tilings on Grids and Graphs

José L. Ramírez

Departamento de Matemáticas
Universidad Nacional de Colombia
Bogotá, Colombia
jlamirezr@unal.edu.co

Diego Villamizar

Escuela de Ciencias Exactas e Ingeniería
Universidad Sergio Arboleda
Bogotá, Colombia
diego.villamizarr@usa.edu.co

In this paper, we explore some generalizations of a counting problem related to tilings in grids of size $2 \times n$, which was originally posed as a question on Mathematics Stack Exchange (Question 3972905). In particular, we consider this problem for the product of two graphs G and P_n , where P_n is the path graph of n vertices. We give explicit bivariate generating functions for some specific cases.

1 Introduction

Question 3972905 in Mathematics Stack Exchange asks for the number of ways to partition a tile $2 \times n$ into s parts. That is the number of different configurations (tilings) in a grid of size $2 \times n$ with exactly s polyominoes using 2 colors. For example, if $n = 4$ we have 12 configurations with exactly 4 polyominoes, see Figure 1.

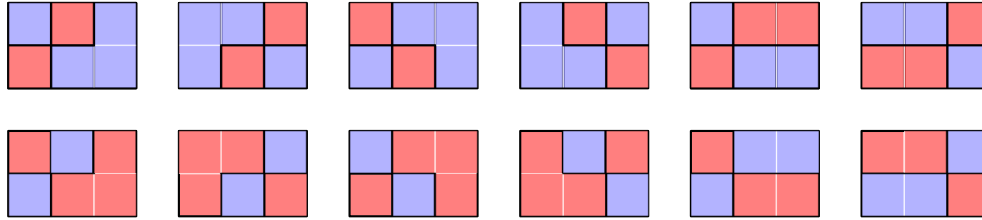


Figure 1: Configurations of a grid 2×3 with exactly 4 polyominoes.

In [4], we study this problem for a general grid of size $m \times n$ and k colors. We employ generating functions to provide a partial solution to this problem for the cases $m = 1, 2, 3$. Specifically, if $c(n, i)$ represents the number of different tilings of a $2 \times n$ grid with exactly i polyominoes and using two colors, then

$$\begin{aligned} \sum_{n,i \geq 1} c(n, i) x^n y^i &= \frac{2xy(1+y-x(1-y)(1-2y))}{1-x(2+y+y^2)+x^2(1-y)(1-5y^2-2y(1-2y))} \\ &= (2y+2y^2)x + (2y+12y^2+2y^4)x^2 + (2y+30y^2+18y^3+12y^4+2y^6)x^3 \\ &\quad + (2y+56y^2+102y^3+56y^4+24y^5+14y^6+2y^8)x^4 + O(x^5). \end{aligned}$$

Figure 1 shows the colored tilings corresponding to the bold coefficient in the above series.

This counting problem was explored by Richey [5] in 2014. Specifically, he showed that $\lim_{n,m \rightarrow \infty} e(m, n)/mn$ exists and is finite, where $e(m, n)$ is the expected number of polyominoes on the $m \times n$ grid. Mansour [3] considers this problem for bicolored tilings ($k = 2$) for $m = 1, 2, 3$ using automata. A related problem was addressed by Bodini during GASCOM 2022, referred to as *rectangular shape partitions* [1].

2 Colored Tilings of Grids

Let $\mathcal{T}_{m,n}^{(k)}$ denote the set of tilings of an $m \times n$ grid with polyominoes colored with one of k colors, such that adjacent polyominoes are colored with different colors. An element of $\mathcal{T}_{m,n}^{(k)}$ is called a k -colored tiling. Given a k -colored tiling T in $\mathcal{T}_{m,n}^{(k)}$, we use $\rho(T)$ to denote the number of polyominoes in T . For fixed positive integers m and k , we define the bivariate generating function

$$C_m^{(k)}(x, y) := \sum_{n \geq 1} x^n \sum_{T \in \mathcal{T}_{m,n}^{(k)}} y^{\rho(T)}. \tag{1}$$

Note that the coefficient of $x^n y^i$ in $C_m^{(k)}(x, y)$ is the number of k -colored tilings of an $m \times n$ grid with exactly i polyominoes. Let $c_{m,k}(n, i)$ denote the coefficient of $x^n y^i$ in the generating function $C_m^{(k)}(x, y)$. In [4], we derive explicit generating functions for the cases $m = 1, 2, 3$. Additionally, we introduce a variation of this problem for hexagonal grids.

The combinatorial problem can be described in terms of graphs. Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be two undirected graphs. The product of G_1 and G_2 is defined as $G_1 \times G_2 = (V_1 \times V_2, E_{G_1 \times G_2})$, where

$$E_{G_1 \times G_2} = \{ \{(v_1, v_2), (w_1, w_2)\} : (v_1 = w_1 \text{ and } \{v_2, w_2\} \in E_2) \text{ or } (v_2 = w_2 \text{ and } \{v_1, w_1\} \in E_1) \}.$$

Let P_n be a *path graph*, that is a simple graph with n vertices arranged in a linear sequence in such a way that two vertices are adjacent if they are consecutive in the sequence, and are non-adjacent otherwise. A *grid graph* of size $m \times n$ is defined as the product $P_m \times P_n$, and it is denoted by $L_{m,n}$.

Let $G = (V, E)$ be an undirected graph. Two non-empty disjoint subsets $V_1, V_2 \subseteq V$ are *neighbors* if there is an edge $(v_1, v_2) \in E$ such that $v_1 \in V_1$ and $v_2 \in V_2$. A k -colored partition of size s of the vertices V of G is a partition of the set $V = \bigcup_{i=1}^s V_i$ such that for each V_i , the induced graph is connected, all vertices in V_i are colored with exactly one of k colors, and any pair V_i and V_j of neighbors are colored with different colors.

For example, Figure 2 (left) shows a 3-colored partition of size 7 of the grid graph $L_{3,8}$, and the Figure 2 (right) shows the corresponding tiling in $\mathcal{T}_{3,8}^{(3)}$.

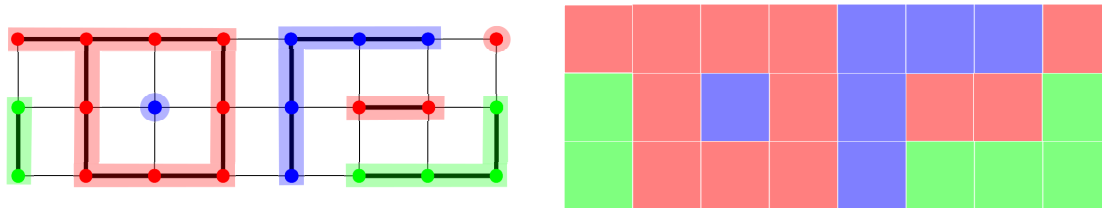


Figure 2: A 3-colored partition of size 7 of $L_{3,8}$.

Let G be an undirected graph. We denote by $\mathcal{T}_n^{(k)}(G)$ the set of k -colored partitions of $G \times P_n$. Given a k -colored partition T in $\mathcal{T}_n^{(k)}(G)$, we use $\rho(T)$ to denote the size of the partition. For fixed positive integers m and k , we define the bivariate generating function

$$C_G^{(k)}(x, y) := \sum_{n \geq 1} x^n \sum_{T \in \mathcal{T}_n^{(k)}(G)} y^{\rho(T)}.$$

It is clear that $C_m^{(k)}(x, y) = C_{P_m}^{(k)}(x, y)$.

3 The complete graph case.

In this section we analyze the case when $G = K_m$, where K_m is the complete graph of size m . For example, Figure 3 shows a 2-colored partition of size 4 of the graph $K_5 \times P_4$.

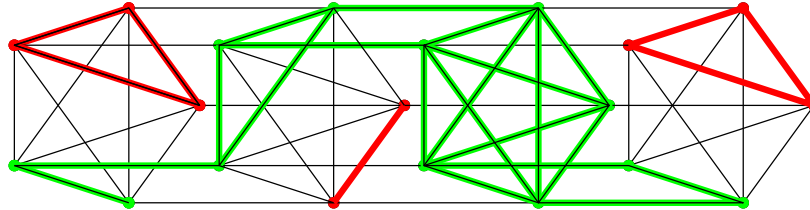


Figure 3: A 2-colored partition of size 4 of $K_5 \times P_4$.

3.1 The case $m = 3$.

In this section we give the explicit bivariate generating function for the 2-colored partitions of $K_3 \times P_n$ for all $n \geq 1$.

Theorem 3.1. *The bivariate generating function $T(x, y) = C_{K_3}^{(2)}(x, y)$ is given by*

$$T(x, y) = \frac{2xy(1 + 3y - x(3 - 7y + 4y^2))}{1 - x(4 + 3y + y^2) + x^2(3 - 7y + 3y^2 + y^3)}.$$

Moreover, $[x^n]T(x, 1) = 8^n$.

Proof. Let \mathcal{A}_n and \mathcal{B}_n denote the sets of colored tilings in $\mathcal{T}_n^{(2)}(K_3)$, such that in the first case the last triangle is colored with only one color, while in \mathcal{B}_n , the last triangle is colored with the two colors.

Now, we define the bivariate generating functions:

$$T_1(x, y) := \sum_{n \geq 1} x^n \sum_{T \in \mathcal{A}_n} y^{\rho(T)} \quad \text{and} \quad T_2(x, y) := \sum_{n \geq 1} x^n \sum_{T \in \mathcal{B}_n} y^{\rho(T)}.$$

It is clear that $T(x, y) = T_1(x, y) + T_2(x, y)$.

Let T be a 2-colored partition in \mathcal{A}_n . If $n = 1$, then $T = K_3$, and its contribution to the generating function is the term $2xy$ because it has to be monochromatic. If $n > 1$, then T may be decomposed as either T_1K_3 or T_2K_3 , where $T_1 \in \mathcal{A}_{n-1}$, and $T_2 \in \mathcal{B}_{n-1}$. Depending on whether the colors of the last two triangles coincide or not, we obtain the cases given in Table 1.

From this decomposition, we obtain the functional equation

$$T_1(x, y) = 2xy + xT_1(x, y) + xyT_1(x, y) + xT_2(x, y) + xT_2(x, y).$$

For the colored tilings in \mathcal{B}_n we obtain the different decompositions given in Table 2. From this decomposition we obtain the functional equation:

$$T_2(x, y) = 6xy^2 + 3xyT_1(x, y) + 3xyT_1(x, y) + 3xT_2(x, y) + xy^2T_2(x, y) + 2xyT_2(x, y).$$

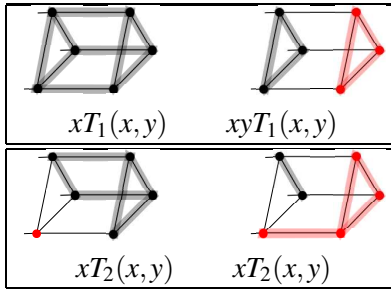


Table 1: Cases for the generating function $T_1(x, y)$.

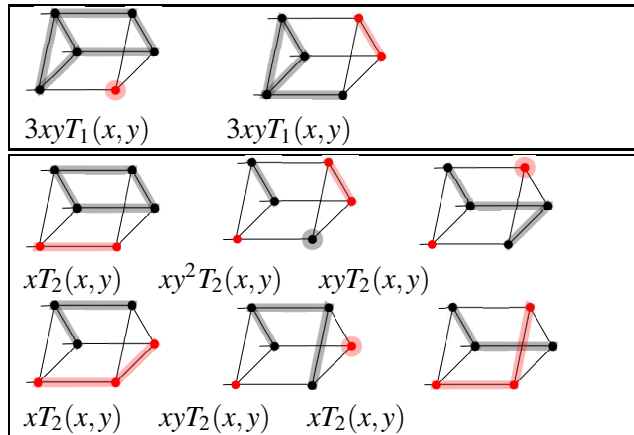


Table 2: Cases for the generating function $T_2(x, y)$.

Since $T(x, y) = T_1(x, y) + T_2(x, y)$, we have a system of three linear equations with three unknowns $T(x, y)$, $T_1(x, y)$, and $T_2(x, y)$. Solving the system for $T(x, y)$ we obtain the desired result. \square

As a series expansion, the generating function $T(x, y)$ begins with

$$T(x, y) = (2y + 6y^2)x + (2y + 44y^2 + 12y^3 + 6y^4)x^2 + (2y + 178y^2 + 218y^3 + 84y^4 + 24y^5 + \mathbf{6y^6})x^3 + (2y + 600y^2 + 1674y^3 + 1100y^4 + 528y^5 + 150y^6 + 36y^7 + 6y^8)x^4 + O(x^5).$$

Figure 4 shows the 2-colored partitions corresponding to the bold coefficient in the above series.

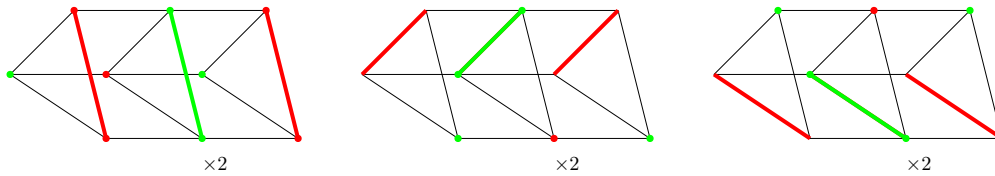


Figure 4: All 2-colored partitions in $\mathcal{T}_3^{(2)}(K_3)$.

Corollary 3.2. *The expected number for the size of the partition when the colors assigned to each vertex are selected uniformly in $\mathcal{T}_3^{(2)}(K_3)$ is given by*

$$\frac{2^{3n-5}(37 + 19n)}{2^{3n}}.$$

References

- [1] O. Bodini (2022): *On the strange kinetic aesthetic of rectangular shape partitions*. *Pure Math. Appl. (P.U.M.A.)* 30, doi:10.2478/puma-2022-0007. Available at <https://sciendo.com/article/10.2478/puma-2022-0007>.
- [2] Mathematics Stack Exchange (2021): *Question 3972905 in Mathematics Stack Exchange*. *Number of ways to partition $2 \times N$. Tile into m parts*. Available at <https://math.stackexchange.com/q/3972905>.
- [3] R. Mansour (2021): *Counting clusters in a coloring grid*. *Discrete Math. Lett.* 5. Available at https://www.dmllett.com/archive/v5/DML21_v5_pp20-23.pdf.
- [4] J. L. Ramírez & D. Villamizar (2024): *Colored random tilings on grids*. To appear in *Journal of Automata, Languages and Combinatorics*.
- [5] J. Richey (2014): *Counting clusters on a grid*. Undergraduate Honors Thesis. Dartmouth College.

A Bijection between Stacked Directed Polyominoes and Motzkin Paths with Alternative Catastrophes

Florian Schager 

TU Graz
Graz, Austria

Institute of Software Technology
florian.schager@tugraz.ac.at

Michael Wallner 

TU Wien
Wien, Austria

Institute of Discrete Mathematics and Geometry
michael.wallner@tuwien.ac.at
dmg.tuwien.ac.at/mwallner

We present a novel bijection between stacked directed polyominoes and Motzkin paths with alternative catastrophes. Further, we show how this new connection can be used in order to obtain a better understanding of certain parameters of stacked directed animals.

Keywords: Bijection, polyominoes, lattice paths, Motzkin paths

1 Stacked directed animals and heaps of dimers

The motivation behind the enumeration of lattice animals or polyominoes can be found in the study of branched polymers [5] and percolation [3]. Even though these combinatorial objects have been studied for more than 40 years, exact enumeration results for general polyominoes are still rare. Thus, one of the main research directions focuses on the investigation of large subclasses of polyominoes that are exactly enumerable. This is also the motivating force behind the class of stacked directed animals that we will define in this section.

Definition 1.1 (Lattice animals). *A polyomino of area n is a connected union of n cells on a lattice. The corresponding lattice animal then lives on the dual lattice obtained by taking the center of each cell.*

The polyominoes we are interested in have square cells, as illustrated in Figure 2. We start now with the definition of a subclass of polyominoes that has already been exactly enumerated by Dhar in [4].

Definition 1.2 (Directed animals). *A directed animal on the square grid is a lattice animal, where one vertex has been designated the source and all other vertices are connected to the source via a directed path consisting only of N - and E -steps, and visiting only vertices belonging to the animal.*

We are interested in so called stacked directed animals, which can be informally described as a sequence of directed animals. However, the easiest description for this class does not build directly upon the above definition. Instead, it defines them indirectly via a one-to-one correspondence to a natural class of heaps of dimers, which were first introduced by Viennot [9]. This approach greatly simplifies the derivation of their generating functions and also serves as an intermediary step for our bijection to Motzkin excursions with alternative catastrophes.

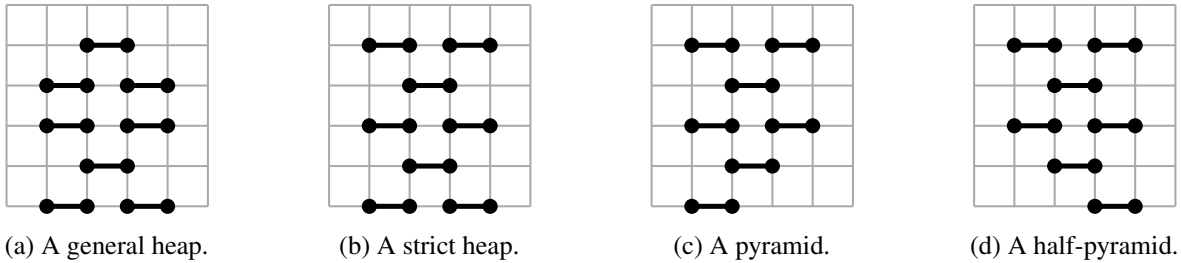


Figure 1: Different types of heaps of dimers.

Definition 1.3 (Heaps of dimers). A dimer consists of two adjacent vertices on a lattice. A heap of dimers is obtained by dropping a finite number of dimers towards a horizontal axis, where each dimer falls until it either touches the horizontal axis or another dimer; see Figure 1. The width of a heap is the number of non-empty columns. The dimers that touch the x -axis are called minimal. A heap is called

- strict, if no dimer has another dimer directly above it;
- connected, if its orthogonal projection on the horizontal axis is connected;
- a pyramid, if it has only one minimal dimer;
- a half-pyramid, if its only minimal dimer lies in the rightmost non-empty column.

The right/left width of a pyramid is the number of non-empty columns to the right/left of the minimal dimer.

Now we will describe a construction from [2, p. 240] that maps directed animals on the square lattice to strict pyramids of dimers.

Definition 1.4 (Mapping from directed animals to heaps). Let \mathcal{D} denote the set of directed lattice animals on the square grid, \mathcal{P} denote the set of strict pyramids and $D \in \mathcal{D}$. We define a mapping $V: \mathcal{D} \rightarrow \mathcal{P}$ as follows:

1. Rotate D by 45° degrees counter-clockwise.
2. Replace each individual cell in D by a dimer.

This results in a pyramid that we call $V(D)$, with the source of the lattice animal being the only minimal dimer.

Remark 1.5. It was observed by Viennot in [9] that this mapping induces a bijection between directed animals on the square lattice and strict pyramids of dimers and we denote the inverse mapping by \bar{V} . This can be easily verified by recalling that any vertex in D lies on a directed path consisting only of \mathbf{N} and \mathbf{E} steps from the source, visiting only other vertices in D . Hence, the corresponding dimer in $V(D)$ lies on a directed path of dimers lying diagonally to the left or the right above each other. As the next definition will show, it only takes a small adaptation to extend this mapping to general lattice animals.

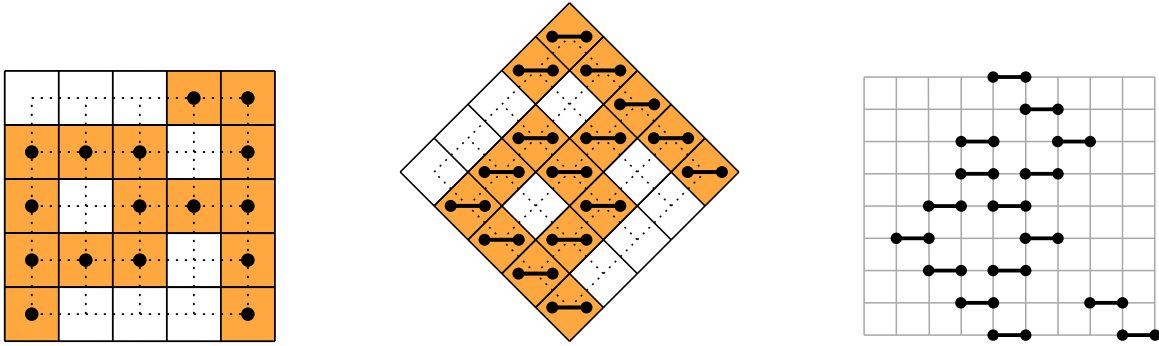


Figure 2: Constructing the connected heap $V(A)$ from an animal A on the square grid.

Definition 1.6 (Mapping from lattice animals to heaps). *Let \mathcal{A} denote the set of lattice animals on the square lattice, \mathcal{H} the set of connected heaps, and $A \in \mathcal{A}$. We define a mapping $V : \mathcal{A} \rightarrow \mathcal{H}$ as follows:*

1. *Rotate A by 45° degrees counter-clockwise.*
2. *Replace each individual cell in A by a dimer.*
3. *Let the dimers fall.*

We call the resulting heap $V(A)$; see Figure 2 for an example of this procedure.

Thus, V maps square lattice animals to connected heaps that are not necessarily strict. However, clearly not every connected heap can be obtained in this way. Hence, we restrict our attention to strict, connected heaps and define a class of lattice animals that stand in one-to-one correspondence with strict, connected heaps via V .

Definition 1.7 (Multi-directed animals). *Let H be a strict, connected heap. We now construct an extension of \bar{V} to connected heaps via induction over the number of minimal dimers k of H :*

- *For $k = 1$, the heap H reduces to a simple pyramid. Thus, by Remark 1.5, $\bar{V}(H)$ is already well-defined.*
- *If instead H has $k > 1$ minimal dimers, we push the $(k - 1)$ leftmost pyramids upwards, producing a connected heap H' with $k - 1$ minimal dimers, placed far above the remaining pyramid P_k . Now, recursively replace H' by $\bar{V}(H')$ and P_k by $\bar{V}(P_k)$.*
- *Finalize the construction by pushing $\bar{V}(H')$ downwards until it connects to $\bar{V}(P_k)$.*

We define $\bar{V}(H)$ as the resulting animal and call the class of square lattice animals obtainable in this way square multi-directed animals.

Now we are ready to define stacked directed animals as a subclass of multi-directed animals.

Definition 1.8 (Stacked directed animals). *Take a connected heap H with k minimal dimers. Let us denote by P_1, P_2, \dots, P_k , from left to right, the corresponding pyramidal factors of H from the construction in Definition 1.7. Let us call stacked pyramids the connected heaps such that for $2 \leq i \leq k$, the horizontal projection of P_i intersects the horizontal projection of P_{i-1} . Then, stacked directed animals are defined as the image of the set of stacked pyramids under \bar{V} . The right width of a stacked pyramid is the right width of its rightmost pyramidal factor.*

These lattice animals are easier to enumerate due to their recursive description, visualized in Figure 3. This description yields algebraic equations for their generating functions. It will also prove crucial in constructing our correspondence to Motzkin excursions with alternative catastrophes that we introduce in the next section.

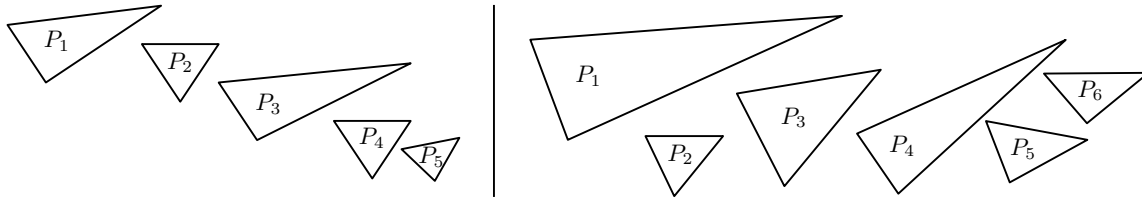


Figure 3: The schematic structure of stacked directed animals (left) and multi-directed animals (right) from Definitions 1.8 and 1.7, respectively [2, Figure 9]. Each triangle represents a directed animal from Definition 1.2.

2 Lattice paths with alternative catastrophes

Definition 2.1 (Lattice paths). Let $\mathcal{S} \subseteq \mathbb{Z}$ be a finite set of integers called steps. A lattice path is a sequence $(s_1, s_2, \dots, s_n) \in \mathcal{S}^n$ of steps with a fixed starting point y_0 .¹ We set $y_0 = 0$ and define $y_k = \sum_{i=1}^k s_i$ as the altitude of the path after k steps. Furthermore, we distinguish different classes of paths:

- An unconstrained path is called a walk.
- A walk ending on the x -axis (i.e., $y_n = 0$) is called a bridge.
- A walk that may never cross the x -axis (i.e., $y_k \geq 0$) is called a meander.
- A walk that is at the same time a bridge and a meander is called an excursion.

Dyck paths are probably the most ubiquitous class of paths, which are excursions associated with the steps $\mathcal{S} = \{-1, 1\}$ and famously enumerated by the Catalan numbers. In our bijections we will encounter the nearly equally famous *Motzkin paths*, which are excursions associated with $\mathcal{S} = \{-1, 0, 1\}$. We will call them Motzkin excursions (resp. Motzkin meanders), when they use the step set of Motzkin paths and are excursions (resp. meanders). Moreover, we will need *2-Motzkin paths*, which are associated with the colored step set $\mathcal{S} = \{-1, 0_1, 0_2, 1\}$, which means that the flat step 0 comes in two different colors. We will now enrich these models by a new type of step that takes a path immediately down to the x -axis.

Definition 2.2 (Catastrophe). For $s > 0$ a catastrophe is a step $-s$, with $-s \notin \mathcal{S}$ allowed only at altitude s .

Dyck meanders with catastrophes were first introduced in 2005 by Krinik et al. [6] as a model for the classical single server queueing system M/M/1/H with a finite capacity, with a constant catastrophe rate γ . In addition, catastrophe queues also arise as simple, natural models of the evolution of stock markets [8], or under the name of *random walks with resetting* in the field of probability theory and statistical mechanics [7]. Later, Banderier and Wallner [1] studied enumerative properties and derived limit laws for several parameters, such as the total number of catastrophes.

Note that catastrophes never coincide with regular jumps. As we will see, it also makes sense to allow such catastrophes as well as catastrophes at height zero. This conveniently leads to a model that is easier to handle and simplifies some of the more tedious calculations. To distinguish these two models, we will refer to catastrophes of the second kind as *alternative catastrophes*.

Definition 2.3 (Alternative catastrophe). For $s \geq 0$, an alternative catastrophe is a step of the form $-s$, allowed only at altitude s , that takes the path immediately down to the x -axis; see Figure 4.

¹In the literature this model is called simple and directed.

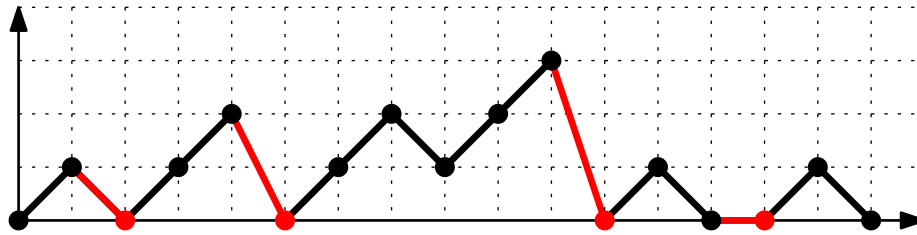


Figure 4: Example of a Dyck excursion with alternative catastrophes marked in red; see Definition 2.3. The first and last catastrophe are not classical catastrophes in the sense of Definition 2.2. Note that a step from altitude 1 to 0 can either be an alternative catastrophe (red) or a step -1 from the step set \mathcal{S} (black).

3 Bijection to Motzkin excursions with alternative catastrophes

We are now ready to link Motzkin excursions with alternative catastrophes to stacked directed animals, respectively, strict heaps. The following bijection is introduced in three steps, always linking more and more complicated objects. We start with a bijection between strict half-pyramids and classical Motzkin paths, both of which are enumerated by A001006 in the OEIS².

Lemma 3.1. *The set of strict half-pyramids of size $n + 1$ is in bijection with the set of Motzkin excursions of length n .*

Proof (Sketch). Observe that both classes satisfy a structurally equivalent decomposition as shown in Figure 5. Thus, it is straightforward to build an explicit bijection. \square

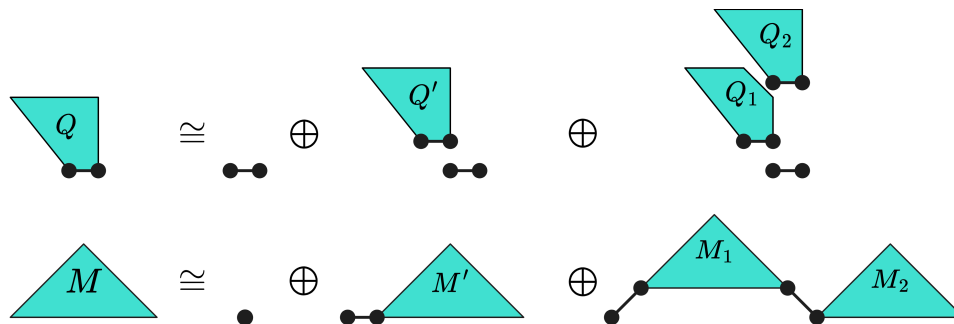


Figure 5: The factorizations of half-pyramids and Motzkin excursions.

Building on this result, we present a bijection from strict pyramids to a subclass of 2-Motzkin paths.

Lemma 3.2. *The set of strict pyramids of size $n + 1$ is in bijection with the set of 2-Motzkin excursions of length n (with black and red E-steps), such that no red E-step occurs at positive height $h > 0$. Equivalently, we could describe it as the set of Motzkin excursions of length n with catastrophes only occurring at height $h = 0$.*

Proof (Sketch). As shown in Figure 6, both structures satisfy structurally equivalent decompositions. \square

Finally, we are now able to present our main result, again, building on the previous results. The corresponding integer sequence is A059712.

²The On-Line Encyclopedia of Integer Sequences: <http://oeis.org/>

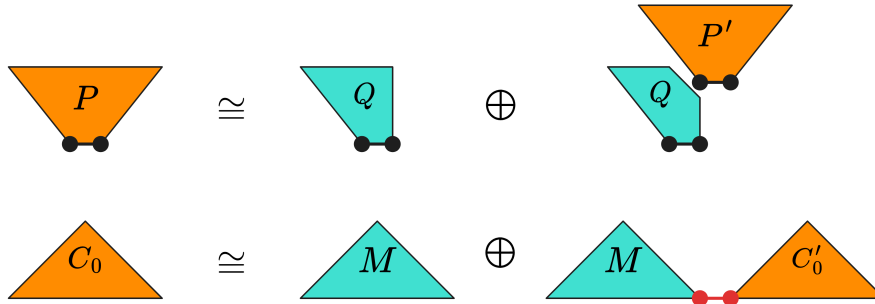


Figure 6: The factorizations of strict pyramids and Motzkin excursions with only horizontal catastrophes.

Theorem 3.3. *The set of Motzkin excursions with alternative catastrophes of length n is in bijection with the set of stacked directed animals of size $n + 1$ on the square grid.*

Proof (Sketch). Let H be the connected heap of dimers representing a stacked directed animal on the square grid and denote by P_1, P_2, \dots, P_k the corresponding pyramidal factors of H . Using Figure 7 as a visual aid, it is easy to see that the rightmost pyramid P_k corresponds to the excursion $C_{0,1}$ in the lattice path. The horizontal distance between the minimal dimer of P_{k-1} and the leftmost dimer of P_k then determines the height ℓ of the catastrophe at the end of E_1 . To offset the height lost with the catastrophe, we mark the start of P_{k-1} with a **NE**-step and, in addition, for the first $\ell - 1$ pyramids in the factorisation of P_{k-1} , we replace the horizontal catastrophe used to mark the beginning of P' by a **NE**-step.

For the reverse direction, let M be a Motzkin excursion with alternative catastrophes. Firstly, we split M into a sequence of excursions E_1, \dots, E_k , each one ending with their first non-horizontal catastrophe. For each excursion, we point out ℓ **NE**-step, with ℓ being the respective height of the final catastrophe by a last passage decomposition: At every height $i = 0, \dots, \ell - 1$ we point out the last step to leave this particular altitude. These **NE**-steps then mark the start of the half-pyramids $Q_{k-1,i}$ each. \square

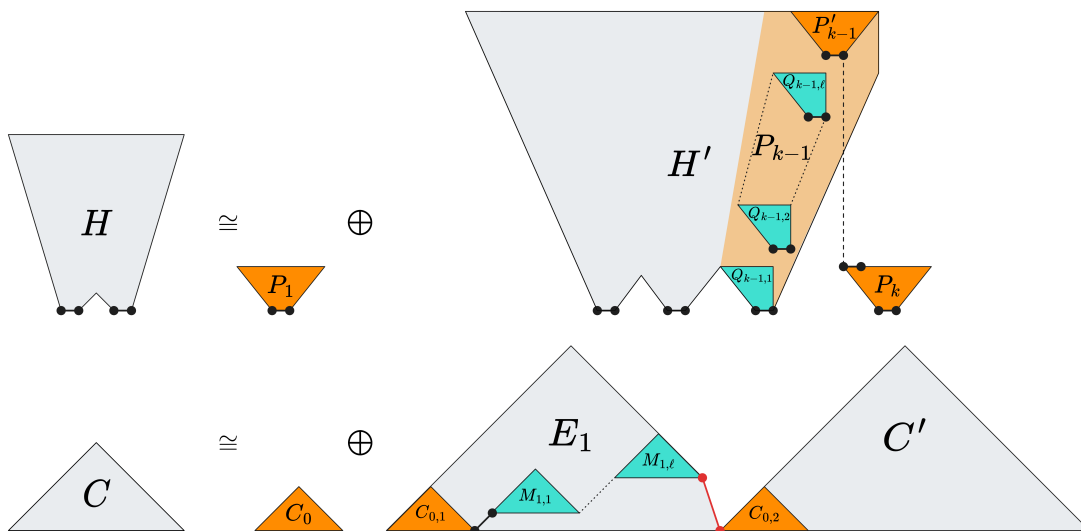


Figure 7: The recursive construction of stacked pyramids and Motzkin excursions with alternative catastrophes.

Figure 8 shows how the bijection transforms an explicit stacked directed animal into a Motzkin path with alternative catastrophes.

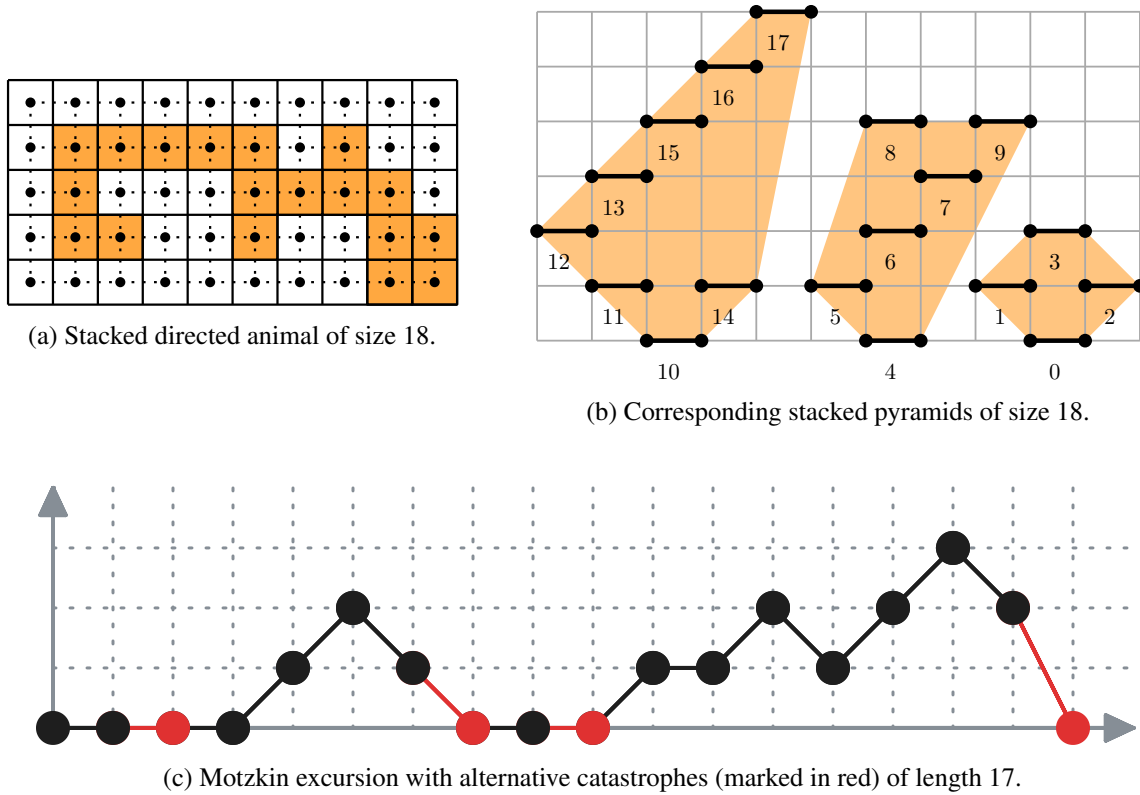


Figure 8: A stacked directed animal and its corresponding Motzkin excursion with alternative catastrophes. The dimers are numbered according to the order of their corresponding steps in the lattice path.

4 Conclusion and outlook

One advantage that is gained by viewing stacked directed animals as lattice paths comes with the reinterpretation of parameters in the language of lattice paths. In particular, we can use the cumulative size of all catastrophes to obtain the improved asymptotic lower bound $\frac{9}{28}n$ on the width of the connected heap associated with the animal compared to $\frac{3}{28}n$ from [2, Proposition 2].

Theorem 4.1. *Let M be a Motzkin excursion with alternative catastrophes of length n , let c_1, \dots, c_k denote the heights of all non-horizontal catastrophes and let H be the associated strict, connected heap obtained by Theorem 3.3. Then we have the following:*

1. *The number of minimal dimers in H is asymptotically equal to $\frac{3}{28}n$.*
2. *The cumulative size of all catastrophes plus the number of minimal dimers is asymptotically equal to $\frac{9}{28}n$.*

Proof (Sketch).

1. The number of minimal dimers in H is equal to $k + 1$. Using [1, Theorem 4.12] we get that this number is asymptotically equivalent to $\frac{3}{28}n$, where we recover the asymptotics from [2, Proposition 2].
2. Since the height of a non-horizontal catastrophe models the distance between a minimal dimer and the left-most dimer of an adjacent pyramidal factor, it serves as a lower bound on the distance between two minimal dimers. Hence, $1 + \sum_{\ell=1}^k (c_\ell + 1)$ yields a lower bound on the width of H . Using [1, Theorem 4.18] we get that the cumulative size of all catastrophes is asymptotically equivalent to $\frac{3}{14}n$. Therefore, $\frac{9}{28}n$ gives a lower bound on the width of stacked directed animals. This result improves the previous best asymptotic lower bound on the width by a factor of three. \square

In the long version of this paper, we will study lattice paths with alternative catastrophes in more detail, as well as more statistics between them and stacked directed animals.

Acknowledgements

Florian Schager was supported by the Austrian Science Fund (FWF): P 36280.

Michael Wallner was supported by the Austrian Science Fund (FWF): P 34142.

References

- [1] Cyril Banderier & Michael Wallner (2017): *Lattice paths with catastrophes*. *Discrete Math. Theor. Comput. Sci.* 19(1), doi:10.23638/DMTCS-19-1-23.
- [2] Mireille Bousquet-Mélou & Andrew Rechnitzer (2002): *Lattice animals and heaps of dimers*. *Discrete Math.* 258(1-3), pp. 235–274, doi:10.1016/S0012-365X(02)00352-7.
- [3] Andrew R. Conway & Anthony J. Guttmann (1995): *On two-dimensional percolation*. *J. Phys. A: Math. and Gen.* 28(4), pp. 891–904, doi:10.1088/0305-4470/28/4/015.
- [4] Deepak Dhar (1982): *Equivalence of the two-dimensional directed-site animal problem to Baxter's hard-square lattice-gas model*. *Phys. Rev. Lett.* 49(14), pp. 959–962, doi:10.1103/PhysRevLett.49.959.
- [5] S. Flesia, D.S. Gaunt, C.E. Soteros & S.G. Whittington (1994): *Statistics of collapsing lattice animals*. *J. Phys. A: Math. and Gen.* 27(17), pp. 5831–5846, doi:10.1088/0305-4470/27/17/016.
- [6] Alan Krinik, Gerardo Rubino, Daniel Marcus, Randall J Swift, Hassan Kasfy & Holly Lam (2005): *Dual processes to solve single server systems*. *J. Statist. Plann. Inference* 135(1), pp. 121–147, doi:10.1016/j.jspi.2005.02.010.
- [7] Lukasz Kusmierz, Satya N. Majumdar, Sanjib Sabhapandit & Grégory Schehr (2014): *First order transition for the optimal search time of Lévy flights with resetting*. *Phys. Rev. Lett.* 113, p. 220602, doi:10.1103/PhysRevLett.113.220602.
- [8] Wim Schoutens (2003): *Lévy Processes in Finance: Pricing Financial Derivatives*. John Wiley & Sons, doi:10.1002/0470870230.
- [9] Gérard Viennot (1985): *Problèmes combinatoires posés par la physique statistique*. *Astérisque* (121-122), pp. 225–246. Available at http://www.numdam.org/article/SB_1983-1984__26__225_0.pdf. Seminar Bourbaki, Vol. 1983/84.

Natural Measures on Polyominoes Induced by the Abelian Sandpile Model

Andrea Sportiello

CNRS, and LIPN, Université Sorbonne Paris Nord*
Villetaneuse, France

andrea.sportiello@univ-paris13.fr

We introduce a natural Boltzmann measure over polyominoes induced by boundary avalanches in the Abelian Sandpile Model. Through the study of a suitable associated process, we give an argument suggesting that the probability distribution of the avalanche sizes has a power-law decay with exponent $\frac{3}{2}$, in contrast with the present understanding of bulk avalanches in the model (which has some exponent between 1 and $\frac{5}{4}$), and to the ordinary generating function of polyominoes (which is conjectured to have a logarithmic singularity, i.e. exponent 1). We provide some numerical evidence for our claims, and evaluate some other statistical observables on our process, most notably the density of triple points.

1 Non-uniform measures on polyominoes from Statistical Mechanics

Given a periodic tiling of the plane, a (general) *polyomino* is a finite connected geometric structure formed by joining one or more cells of the tiling edge to edge. The name polyomino is typically associated to the square grid, while for the triangular and hexagonal grids the names polyiamonds and polyhexes (respectively) are sometimes used [30, 37, 38].

The history in the study of polyominoes started within recreational mathematics more than one century ago [15, 16, 18]. In a modern vision, they form a challenging problem in Combinatorics and Statistical Mechanics (see e.g. [19, sec. 10.8] or [17]), somewhat in analogy with the study of Self-Avoiding Walks: despite allowing for an elementary and natural definition, very little is known rigorously from a mathematical perspective, although mathematicians and physicists have provided numerous conjectures that are believed to be true and are strongly supported by numerical simulations. A reason for this difficulty is that, within the field of exactly-solvable models in Statistical Mechanics, we know more about locally-homogeneous random systems, than about finite compact random structures embedded in Euclidean space.

Given a polyomino P , define $n(P)$, the *size* of P , as the number of faces contained in P . The exhaustive generation of polyominoes, or their enumeration, at any finite size n , is a finite problem, and the problem of decreasing the computational cost of the associated algorithms has been studied by several authors [4, 5, 9, 14, 22, 23, 39]. That is, calling $A_n^{\mathcal{L}}$ the number of polyominoes of size n on a given lattice \mathcal{L} , the problem of determining the first N values $\{A_1^{\mathcal{L}}, \dots, A_N^{\mathcal{L}}\}$, with the smallest possible asymptotic growth of the complexity as a function of N (and the largest possible value of N given the present technology), is an interesting problem in the theory of algorithms, and also a topic appropriated for GASCom, but it is not our subject today.

*The work of A. Sportiello is supported by the French ANR projects DIMERS (ANR-18-CE40-0033) and COMBINE (ANR-19-CE48-0011).

Determining the asymptotic of $A_n^{\mathcal{L}}$ as a function of n is a very interesting subject. Padé approximants can be used on the list of the first few values, so that the previous question is important also for this goal, but also other insights can give access to this information, mostly coming from Statistical Physics. It is believed that $A_n^{\mathcal{L}} \sim c_{\mathcal{L}} \lambda_{\mathcal{L}}^n / n$, [22], where the overall constant $c_{\mathcal{L}}$ and the *growth rate* $\lambda_{\mathcal{L}}$ are expected to depend on the lattice (for the square lattice it is known that $4.00253 \leq \lambda_{\square} \leq 4.5252$ and the best estimates are $c_{\square} \simeq 0.3169$ and $\lambda_{\square} \simeq 4.0626$ [23]), while, crucially, the exponent -1 of the algebraic correction is an exact rational, and it is expected to be *universal* (in the sense of universality for Critical Phenomena [41]), and is a *critical exponent*, i.e., among its various properties of robustness, it shall be the same for all two-dimensional lattices.

Finally, it is of interest to determine the asymptotics for large n of statistical observables of large random polyominoes, taken with the uniform measure. Some examples of interesting observables are the perimeter, that is, the number of edges on the boundary, and the gyration radius, that is, the radius of the smallest disk that contains the polyomino. The average of both these quantities is expected to scale algebraically with n , again with some *critical exponents* expected to be the same for all lattices.

An interesting subclass of polyominoes consists of *simply-connected* polyominoes, that is, polyominoes such that the boundary consists of a single cycle (or, in simple words, “polyominoes with no holes”), see for example [17]. The same questions as above (determination of the A_n ’s, asymptotics, critical exponents for observables like the perimeter and the radius of gyration, . . .) apply to this subfamily, and involve in principle a different set of critical exponents.

The point of this paper is that one can consider some measure of interest $\mu_n(P)$ over polyominoes P of size n , instead that the uniform one. Of course, for such a measure to be interesting, it shall relate to some relevant probabilistic process. Again, this connects to the notion of universality of critical phenomena, where modifying the measure in such a way would correspond to “couple” the first model to a second one, and tune again the parameters such that the system becomes critical (of which a signal would be the fact that the natural “Boltzmann” series, i.e., the grand-canonical partition function, has an algebraic singularity at $z = 1$). An example of such a philosophy comes from random planar maps. On one side, there is an overwhelming evidence that critical exponents associated to maps (asymptotics in the enumeration, scaling of distances, etc.) are universal, that is do not depend on the precise local structure of the map (for example, are the same for random triangulations, or for quadrangulations, or for all maps altogether). Furthermore, if one consider maps “with matter” (that is, coupled with a critical Statistical Mechanics model, such as the Ising Model, the Potts Model, the $O(n)$ Loop Model, etc.), the critical exponents change, again in a universal way, that depends only on the type of matter introduced, and not on the local structure of the map. In some rare cases, the introduction of matter may even simplify the problem (for example, the enumeration of maps is much simpler if they are equipped with a spanning tree, which is the limit $q \rightarrow 0$ of the q -colour Potts Model).

In the case of polyominoes, a simple example in this direction is the measure induced by critical site percolation (that is, the q -state Potts model in the limit $q \rightarrow 1$), e.g. on the triangular lattice (which is the simplest case, as, by simple symmetry arguments, it is known that the critical parameter is $q_c = \frac{1}{2}$). Interestingly, this measure is much simpler to study than the original problem, and is quite explicit: calling $b(P)$ the number of faces not in P , and adjacent to P , we have $\mu_n^{\text{perc}}(P) = 2^{-n-b(P)+1}/n$ for polyominoes P of size n . Also, in this case, exact sampling in polynomial time can be performed quite easily: one should just explore the percolation cluster containing the origin, repeating the algorithm up to have the desired size, and perform anticipated rejection on small clusters. The peculiar factor $1/n$ has a trivial explanation in this case: when the underlying lattice is face-transitive (as is the case for the square, hexagonal and triangular lattices, for example), without loss of generality we can consider polyominoes rooted at one face, as there is a 1-to- n correspondence between unrooted and rooted objects. In particular

the corresponding enumeration series is just $nA_n^{\mathcal{L}}$, and all statistical averages remain the same.

The measure induced by percolation is a simple illustration of how modifications of the uniform measure induced by a Statistical Mechanics model on the whole plane, although apparently more complicated, may be more accessible than the uniform measure. Exploring one certain class of examples within this framework, namely the ones induced by the Abelian Sandpile Model (ASM) of Statistical Mechanics [12] (which is related to Uniform Spanning Trees, that is, the q -state Potts model in the limit $q \rightarrow 0$), is the topic of this paper. Contrarily to the model of percolation (and, more generally, of critical q -colour Potts Model), this model induces measures on polyominoes supported on the simply-connected ones, that is, our (grand canonical) measures $\mu(P)$ will be non-zero if and only if the polyomino P has no holes.

Other natural measures on lattice animals, with a large literature, that we do not mention at length in this paper are for example the *Diffusion-Limited Aggregation* model (DLA) or the *Eden Model* [3, 13, 29, 40]. These models are, yet again, simpler versions of the uniform measure over polyominoes, but, contrarily to the point stressed here, the simplification does not come from the fact that the measure is defined in terms of a Statistical Mechanics model, but rather from the fact that the configurations can be generated by iteratively adding the unit elements one by one, with some growth rule.

2 Avalanches in the Abelian Sandpile Model and polyominoes

The *Abelian Sandpile Model* [1] is a lattice automaton in the class of out-of-equilibrium models in Statistical Mechanics. Pictorially, it is a model in which some “sand” arrives in the system, according to some protocol, and then the local instabilities are relaxed through some “sand avalanches”, which are possibly large, so that the sand can ultimately leave the system through its boundary. When a single grain of sand is added, provided that an avalanche occurs, every site has performed either a positive number of topplings, or none, and the set of sites which have performed at least one toppling is connected, and thus constitutes a non-empty polyomino. Here we shall give a short introduction to the formalism, following in part the notations of [6, 12].

By the celebrated work of Dhar and collaborators [11, 12, 26, 27], it is known that, under the protocol in which the sand is added randomly and uniformly, the steady-state probability distribution of the sand configurations is supported on the so-called “recurrent configurations”, and is uniform. Also, the uniform measure is stable under addition of any given configuration, followed by relaxation. These configurations are characterised by the avoidance of an infinite list of “forbidden subconfigurations” (FSC), and are in bijection with the spanning trees of the lattice, rooted at the boundary, through a (slightly non-canonical¹) algorithm called “burning test”. The relation between configurations and spanning trees is valid if we consider the boundary as a single site. If instead we prefer to keep a visual notation induced by the lattice, and do not connect the boundary edges among themselves, it is more precise to say that the relation is with rooted spanning forests, where each component of the forest is rooted at a boundary edge. Yet another characterisation of recurrent configurations is that, by adding a “frame identity” to the configuration and performing the resulting avalanches, the system goes back to the original configuration, and the avalanche consists in exactly one toppling per site (the frame identity Id_f is the configuration such that $\text{Id}_f(v)$ is the number of boundary edges incident on v).

It is useful to recall the main ideas of the Propp and Wilson LERW algorithm [33] for the exact sampling of rooted spanning trees, or more generally rooted spanning forests. The algorithm, for a

¹The bijection is described in terms of an auxiliary data structure: for each site, one shall choose a total ordering of the incident edges.

generic graph with boundary edges, goes as follows. Choose any ordering of the sites of the domain (excluding the boundary). Initialise the *absorbing set* to the boundary. Then, for every site, if it is not already in the absorbing set, start a random walk from the site (with rates associated to the Laplacian matrix of the graph), up to reaching the absorbing set, and add to the absorbing set the loop-erasure of this walk (performed in the time ordering of the walk). At the end of the algorithm we have a rooted spanning forest, with roots on the initial absorbing set, uniformly sampled, and in bijection with recurrent configurations through the burning test.

From the point of view of Statistical Mechanics, the most natural measure on sand configurations is the uniform measure on recurrent configurations. From this point onward, our constructions will be tacitly assumed to be performed over sand configurations sampled with this measure.

Some reflection shows that, for an avalanche to produce a non-simply-connected polyomino, it shall surround a FSC, thus the measure on polyominoes induced by avalanches on uniform random recurrent configurations is supported on the simply-connected subfamily. This remark is implicit in the work of Dhar, and appears explicitly for example in [34].

In general, avalanches may involve more than one toppling on certain sites, a well-known fact which has led to the definition of “waves of avalanches”, in [21]. The characterisation of recurrent configurations has an implication on the wave decomposition. Indeed, for any recurrent configuration z , the relaxation of $z + \text{Id}_f$ gives again z , through an avalanche that makes each site topple exactly once. As a result, for every portion of the frame identity, $0 \prec u \prec \text{Id}_f$, the relaxation of $z + u$ must produce an avalanche that makes each site topple either one or zero times, and the support of sites which have not toppled must remain accessible from the boundary, as they will be toppling if we now add $\text{Id}_f - u$ to the configuration and relax. In other words, if we add the amount of sand described by $0 \prec u \prec \text{Id}_f$, the resulting avalanche will contain no more than a single wave. We shall call *boundary avalanche* an avalanche induced by a u of this form.

The study of the probability distribution of avalanches, and possibly of the single waves, has been performed since the early days of the model, but has proven difficult and controversial, and also complicated to analyse on numerical experiments, because of strong finite-size corrections [2, 21, 24, 28, 31, 32]. Part of the complicity is due to the interplay among the different waves (cf. in particular [31]). It is thus conceivable that the study of boundary avalanches does not suffer of the same pathologies as for generic avalanches.

For definiteness, let us describe a process consisting of single-site boundary avalanches, that we shall call the *permutation boundary avalanche process*. Let us call V the number of sites in the domain (i.e. its “volume”), $\mathcal{B} \subset E$ the set of boundary edges, and $B = |\mathcal{B}| = |\text{Id}_f|$ the number of boundary edges (which is also the number of sand grains in the frame identity). Let $\sigma \in \mathfrak{S}_B$ be a random permutation of the boundary edges. We can add the grains of sand constituting Id_f one by one, in the order given by σ , and register the B (possibly empty) avalanches. By the abelianity properties of the ASM, the collection of all the B supports of the avalanches (i.e., the B polyominoes) coincides with the avalanche due to the addition of the whole frame identity, and thus constitutes a partition of the domain. By the stability of the uniform measure on recurrent configuration under addition of deterministic configurations, for every $1 \leq k \leq B$, the probability distribution over the polyomino associated to b_k , the k -th boundary edge in the order of σ , is only a function of the boundary edge itself, and not of the position it occupies in the ordering σ . In particular, if v_b is the average size of the polyomino associated to a boundary avalanche due to the boundary edge b , we must have $\sum_b v_b = V$ (again, regardless of the choice of σ). In particular, on a lattice in which the boundary edges are all equivalent (i.e., on “boundary-edge-transitive graphs”²),

²That is, graphs G with an outer boundary \mathcal{B} s.t., for all $b_1, b_2 \in \mathcal{B}$, there exists $g \in \text{Aut}(G)$ s.t. $g(b_1) = b_2$.

we must have $v_b = V/B$ for all b .

A strongly related process, that we shall call the *BT boundary avalanche process*, is more directly related to the burning test, and the Propp and Wilson algorithm for generating uniform rooted spanning trees [33]. In this case, for each site v we shall choose, once and for all, a total ordering \mathcal{O} of the set of incident edges. We shall now add the whole Id_f , and perform the relaxation in parallel. The sand grains of Id_f are “coloured” in B different colours. Each site v will become unstable at some moment of the avalanche, that is, it will have a height $h + c$, where k is the maximal allowed stable height, and $1 \leq c \leq d(v)$. The colour of the site v is inherited from the colour of the site u which has donated the c -th grain of sand among those which have been donated to v at the present stage of the avalanche, where “the c -th” neighbour is defined according to the given ordering \mathcal{O} . We can visualise the process of colour inheritance by drawing an oriented edge (uv) in this case. The overall set of oriented edges added in this way describes the rooted spanning forest which, through the burning test, is in bijection with the given recurrent configuration. And, as we have mentioned, the partition of the domain into polyominoes can be studied in terms of the components of the forest obtained through the Propp and Wilson algorithm.

We must have some values v'_b (in principle different from the v_b 's) for the average size of the polyomino associated to the tree rooted on the boundary edge b , with $\sum_b v'_b = V$. The independence of the set of spanning forests from the choice of ordering \mathcal{O} implies that the v_b 's do not depend on \mathcal{O} , and thus, in particular, on a boundary-edge-transitive graph, we have $v'_b = V/B$ for all b .

A crucial non-trivial fact is that the permutation boundary avalanche process and the BT boundary avalanche process are in fact *the same probabilistic process*. A way of seeing this is to realise that in the permutation boundary avalanche process, for any given σ , we can construct some trees on the various avalanches, following the rules of the burning test. Conditioning the sand configuration z to have some avalanche support $P = P_{b_1}$ for the boundary avalanche associated to the boundary edge b_1 corresponds to say that $z|_P$ is recurrent for an ASM model defined on a suitable restriction of the domain to $G \setminus P$, with appropriate boundary conditions, and that the heights in the sites adjacent to P are such that, after the topplings on P have been performed, no site has reached its critical height value (this condition can be rephrased by a shift of both the height values and the critical height values at these sites). We can use this argument repeatedly, for all b in \mathcal{B} in the order given by σ , to deduce that the spanning forests constructed from the permutation boundary avalanche process for the given σ , applied to the list of all recurrent configurations, produce the list of all spanning forests on the domain, with no repetitions. In particular, $v'_b = v_b$ for all b , and more generally we can calculate any observable for one process using the defining properties of the other process (we will use this argument several times in the following sections). See Figure 1 for an illustration.

A typical example of boundary-edge-transitive domain is a $L_x \times L_y$ cylinder, in which (say) L_x is the periodicity and L_y is the distance between the two portions of the boundary. In this case $V/B = L_y/2$, and thus is a divergent quantity if we perform the thermodynamic limit $V \rightarrow \infty$ by keeping the aspect ratio fixed. We shall call this case the *cylinder geometry*. A variant of this geometry is again a $L_x \times L_y$ cylinder, but now, instead of having two open boundaries, we have an open boundary and a “folded” boundary, that is, a toppling at (i, j) on this boundary leaves one particle at (i, j) , and gives out three particles, in the directions W,S,E. We shall call this case the *folded cylinder geometry*. Note that the folded geometry can be interpreted as an ordinary geometry $L_x \times 2L_y$, where we restrict to configurations which are symmetric under horizontal reflection (and add the sand to the system accordingly).

Some examples of realisations of this process are given in Figure 4 at the end of this paper.

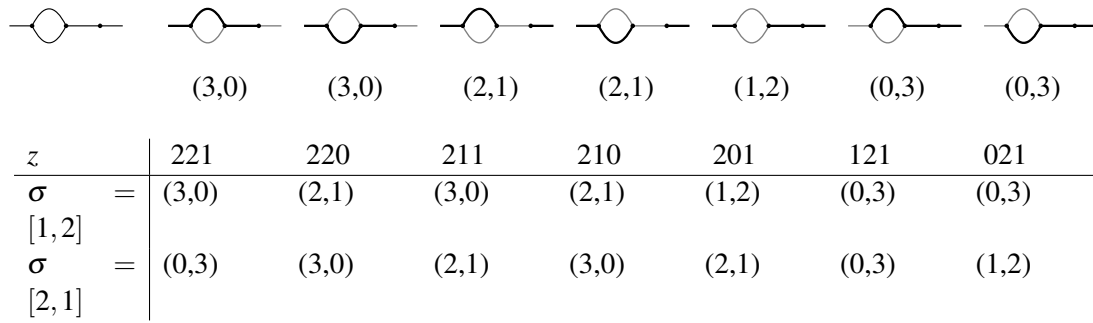


Figure 1: An example of the correspondence between the BT boundary avalanche process and the permutation boundary avalanche processes for the possible choices of σ , for the small graph with $V = 3$ and $B = 2$ depicted on the top-left corner. Top: the list of the 7 spanning forests, and the corresponding list of $(|T_1|, |T_2|)$. Bottom: the 7 recurrent configurations, and the associated lists of $(|P_1|, |P_2|)$ for the 2 permutations of the boundary edges. The three unordered lists are the same (namely, $(3,0)$, $(2,1)$, $(1,2)$ and $(0,3)$ are repeated 2,2,1,2 times, respectively), this being the consequence, for this graph, of the statement that the permutation boundary avalanche process and the BT boundary avalanche process on the uniform measure over recurrent configurations are the same probabilistic process.

3 Some accessible observables in the Boundary Avalanche Process

In this section we want to evaluate some statistical observables in the Boundary Avalanche Process. The key idea is that we can use the bijection between the implementation of the burning test and the construction of spanning forests rooted at the boundary edges. Then, we can use either the implications of the Propp and Wilson LERW algorithm [33], or also, more directly, the Kirchhoff Matrix-Tree Theorem, by evaluating determinants of suitable Laplacian matrices. Not surprisingly, these probabilities will turn out to be ratios of determinants of very similar matrices, so that in fact, by the Jacobi's theorem on complementary minors, through "small" determinants involving the inverse of the Laplacian matrix (that is, the Green's function).

Note however that not all the potentially useful observables can be calculated directly by this method. For reasons reminiscent of the Lindström–Gessel–Viennot lemma, or the Kasteleyn solution of the Dimer Model on bipartite planar graphs, probabilities of events are accessible only if some topological property of the event guarantees that the signs appearing in the determinant are controlled.

A useful formalism goes through Grassmann calculus, that is, a representation of determinants (and determinants of minors) as formal Gaussian integrals over complex scalar non-commuting variables, as described in detail in [8, 20]. In this case, the roots $R = \{r_i\}$ of the forests are described by factors $\bar{\psi}_{r_i} \psi_{r_i}$ in the integrand, while the factor $\bar{\psi}_{u_1} \psi_{v_1} \cdots \bar{\psi}_{u_k} \psi_{v_k}$ implements the fact that the vertices in some ordered list $U = (u_1, \dots, u_k)$ are connected pairwise to the vertices in the list $V = (v_1, \dots, v_k)$ (according to some permutation $\sigma \in \mathfrak{S}_k$, that is, u_j is in the same component than $v_{\sigma(j)}$, and is not in the same component of any other u_i , or v_i , or r_i). However, such an event comes with a sign equal to the signature of σ . That is, for the three lists $R = \{r_1, \dots, r_h\}$, $U = (u_1, \dots, u_k)$ and $V = (v_1, \dots, v_k)$, we will consider Grassmann integrals of the form

$$Z_{R,U,V} = \int \mathcal{D}(\psi, \bar{\psi}) \left(\prod_{r \in R} \bar{\psi}_r \psi_r \right) \bar{\psi}_{u_1} \psi_{v_1} \cdots \bar{\psi}_{u_k} \psi_{v_k} e^{\bar{\psi} L \psi}. \quad (1)$$

The consequence of the Kirchhoff Theorem is that these expressions count (with signs) certain $h + k$ -component spanning forests of the graph,

$$Z_{R,U,V} = \sum_{\substack{F=\{T_1, T_2, \dots, T_{h+k}\} \subseteq G \\ r_i \in T_i \text{ for } 1 \leq i \leq k \\ u_i, v_{\sigma(i)} \in T_{k+i} \text{ for } 1 \leq i \leq h}} \varepsilon(\sigma). \quad (2)$$

The explicit calculations on a generic weighted digraph \mathcal{G} (with a boundary), such that the sum of the weights of the outgoing edges of a vertex is the same for all vertices, involve the graph Green's functions $G(u, v)$, identified by the defining equation $L_u G(u, v) = \delta_{u,v}$, where $L_u f(u) = \sum_{(uu')} w_{(uu')} (f(u') - f(u))$ is the graph (weighted) Laplacian (w.r.t. position u). The collection of the “boundary Green's functions” $G(u, v)$, for v on the boundary of \mathcal{G} , corresponds to the probabilities that a random walk, starting at u , diffusing with the weights w_e and absorbed at the boundary, terminates in v . Thus, in particular,

$$\sum_{v \in \partial \mathcal{G}} G(u, v) = 1 \quad \forall u. \quad (3)$$

These remarks are of interest here because, as we will see, most of the interesting choices of (R, U, V) in (2) are such that (say) $U \cup R = \partial \mathcal{G}$, so that the relevant Green's functions in the evaluation of $Z_{R,U,V}$ are indeed boundary Green's functions in the sense above.

The calculations are more explicit on portions of regular lattices, and involve lattice sums on certain lattice Green functions on the domain, which, when the domain allows for the use of the “method of images”, can be constructed in terms of the lattice Green function of the infinite lattice under investigation (most notably, the square, triangular or hexagonal lattice). The theoretical investigation of lattice Green function has a long history, of which a breakthrough result is due to Lüscher and Weisz [25] (where an important ingredient is an observation of Vohwinkel unpublished elsewhere), which, for the square and triangular lattice, has been implemented in [35, 36] and in [10], respectively (recall that, as polyominoes are defined on the faces of the lattice, the Green function of the triangular lattice in fact relates to polyhexes). See also [7] for further details.

In order to calculate the algebraic asymptotic decay of probabilities of events, however, it is enough to use the asymptotic Green function, which for all lattices, once that the lattice spacing is rescaled in order to have unit density, is universally $G(\vec{x}_1, \vec{x}_2) = \frac{1}{4\pi} \ln |\vec{x}_1 - \vec{x}_2|^2$. However, in the special case of a straight boundary, the method of images implies that (say, for the square lattice) we have to consider the combination $G(\vec{x}_1, \vec{x}_2) - G(\vec{x}_1, \vec{x}_2 - 2\hat{e}_y)$, that scales as $G_{\text{bd}}(x, y) = \frac{1}{\pi} \frac{y}{x^2 + y^2}$ for $x^2 + y^2 \gg 1$ (for the triangular lattice with unit density, we have a correction factor $\alpha = 2^{\frac{1}{2}} 3^{-\frac{1}{4}}$).

A first warm-up example of observable can be the explicit check of the simple fact that any site must be in some tree of the forest. So we must have the identity

$$Z_{\mathcal{B}, \emptyset, \emptyset} = \sum_{b \in \mathcal{B}} Z_{\mathcal{B} \setminus b, (b), (s)} \quad \forall s. \quad (4)$$

On a generic graph \mathcal{G} , and using the Kirchhoff Matrix-Tree Theorem and Jacobi minor formula, this is rephrased into the statement (3) above (and indeed the random walk defining the boundary Green's function can be interpreted as the support for the first LERW in Propp and Wilson's algorithm, when s is chosen to be the first vertex in the ordering).

It is instructive to check that, for the specific case of the square lattice and in a limit of $y \gg 1$,

$L_x, L_y \gg y$,³ the combinatorial statement above is in agreement with the identity

$$\sum_{x \in \mathbb{Z}} \frac{1}{\pi} \frac{y}{x^2 + y^2} \simeq \int_{-\infty}^{\infty} dx \frac{1}{\pi} \frac{y}{x^2 + y^2} = 1 \quad \forall y, \quad (5)$$

(for the triangular lattice, a factor α^{-1} for the density of sites along a row cancels out with the scaling factor α in the Green's function).

A more interesting calculation consists (for example, in the case of hexagonal cells) in determining the probability that the vertex in (x, y) is a triple point of the process, that is, its three adjacent hexagonal faces are in three different polyominoes. The fact that 3 is an odd number, that the set U is on the outer boundary and the set V consists of adjacent faces implies that the annoying signs are in fact protected, that is, of the six possible permutations, only the three connectivity patterns with equal signature are allowed. For $y \gg 1$ the probability that (x, y) is a triple point, and the three adjacent polyominoes are rooted on the points (x_1, x_2, x_3) , is given (up to a simple scaling factor for the lattice spacings, and in a limit $L_x, L_y \gg x, y, x_1, x_2, x_3$) by the determinant of the matrix

$$M[x, y, (x_1, x_2, x_3)] = \left(\frac{y}{(x_i - x)^2 + y^2}, \frac{\partial}{\partial x} \frac{y}{(x_i - x)^2 + y^2}, \frac{\partial}{\partial y} \frac{y}{(x_i - x)^2 + y^2} \right)_{i=1,2,3} \quad (6)$$

Integrating over x and y gives the overall probability that the polyominoes rooted on the real axis at coordinates (x_1, x_2, x_3) share a triple point. A calculation shows that this probability is proportional to the inverse of the Vandermonde factor,

$$\int_{-\infty}^{\infty} dx \int_0^{\infty} dy \det M[x, y, (x_1, x_2, x_3)] \propto \frac{1}{(x_3 - x_2)(x_3 - x_1)(x_2 - x_1)}. \quad (7)$$

Integrating over the x_i 's, at $x = 0$, gives the overall probability that $(0, y)$ is a triple point, which is, for y large enough,

$$\int_{-\infty \leq x_1 \leq x_2 \leq x_3 \leq \infty} dx_1 dx_2 dx_3 \det M[0, y, (x_1, x_2, x_3)] = \frac{1}{2\pi y^2}. \quad (8)$$

Indeed, the algebraic decay $1/y^2$ is integrable at infinity, a fact in agreement with the deterministic information that there are exactly $L_x - 2$ triple points in a configuration on a folded cylinder, that is, asymptotically on average one triple point per column.

Now we calculate an observable in which the role of the signs is more subtle. Consider a realisation of the boundary avalanche process, in a limit $L_x, L_y \rightarrow \infty$, so that the boundary vertices can be totally ordered along \mathbb{Z} . For $i < j$, if the polyominoes P_i and P_j share a boundary, then they have exactly two triple points, with some polyominoes $P_{k_{\text{int}}(i,j)}$ and $P_{k_{\text{ext}}(i,j)}$. A peculiar fact is that, of these two vertices, only one will be in the range $\{i + 1, \dots, j - 1\}$ (we will set it to be $k_{\text{int}}(i, j)$). So we can define unambiguously the vector $\vec{v}_{ij} = t_{i,j,k_{\text{ext}}(i,j)} - t_{i,j,k_{\text{int}}(i,j)}$, where $t_{i,j,l}$ is the triple point between the polyominoes P_i , P_j and P_l , and set $\vec{v}_{ij} = 0$ if P_i and P_j do not share a boundary. Now, given two adjacent faces v_1, v_2 , consider $Z_{\mathcal{B} \setminus \{u_i, u_j\}, \{u_i, u_j\}, \{v_1, v_2\}}$. This quantity gives the probability that $v_1 \in P_i$ and $v_2 \in P_j$, minus the probability that $v_1 \in P_j$ and $v_2 \in P_i$. Call $e'_{v_1, v_2} = (v'_1, v'_2)$ the oriented dual edge associated to the oriented edge (v_1, v_2) . Remark that

$$\sum_{(v_1, v_2)} e'_{v_1, v_2} Z_{\mathcal{B} \setminus \{u_i, u_j\}, \{u_i, u_j\}, \{v_1, v_2\}} = \mathbb{E} \vec{v}_{ij}. \quad (9)$$

³In this limit we can use the asymptotic form of the boundary Green's function given above, and, as it will be useful only later on, trade lattice derivatives with ordinary derivatives.

Indeed, the boundary between P_i and P_j is a polygonal curve resulting from the concatenation of dual edges (in either orientation), going from $t_{i,j,k_{\text{int}}(i,j)}$ to $t_{i,j,k_{\text{ext}}(i,j)}$.

Similar arguments give, for a region Ω ,

$$\sum_{(v_1, v_2): v'_1 \in \Omega, v'_2 \notin \Omega} Z_{\mathcal{B} \setminus \{u_i, u_j\}, \{u_i, u_j\}, \{v_1, v_2\}} = \mathbb{P}(t_{i,j,k_{\text{int}}(i,j)} \in \Omega, t_{i,j,k_{\text{ext}}(i,j)} \notin \Omega) - \mathbb{P}(t_{i,j,k_{\text{ext}}(i,j)} \in \Omega, t_{i,j,k_{\text{int}}(i,j)} \notin \Omega). \quad (10)$$

Calculations of the asymptotic behaviour of observables of this form rely on the evaluation of the quantities $Z_{\mathcal{B} \setminus \{u_i, u_j\}, \{u_i, u_j\}, \{v_1, v_2\}}$, which are related to the evaluation of the determinant of a matrix of the form

$$M'[x, y, (x_1, x_2)] = \left(\frac{y}{(x_i - x)^2 + y^2}, \frac{\partial}{\partial x} \frac{y}{(x_i - x)^2 + y^2} \right)_{i=1,2}. \quad (11)$$

In particular, taking as Ω the half-plane above height y , and summing over all pairs $i < j$, we have

$$\begin{aligned} \frac{1}{L_x} \sum_{i < j} (\mathbb{P}(t_{i,j,k_{\text{int}}(i,j)} \in \Omega, t_{i,j,k_{\text{ext}}(i,j)} \notin \Omega) - \mathbb{P}(t_{i,j,k_{\text{ext}}(i,j)} \in \Omega, t_{i,j,k_{\text{int}}(i,j)} \notin \Omega)) \\ = \int_{-\infty \leq x_1 \leq x_2 \leq \infty} dx_1 dx_2 dx_3 \det M'[0, y, (x_1, x_2)] = \frac{1}{\pi y}. \end{aligned} \quad (12)$$

Now the algebraic decay $1/y$ is not integrable at infinity, and gives a sensible information on the fractal properties of the process. We discuss the implications of this calculation in the next section.

4 A scaling argument

We shall try to give a prediction for the asymptotic behaviour of the tail of the probability distribution for the boundary avalanches. Say that we are in a cylinder with aspect ratio of order 1. Let us suppose that, on some length scales much larger than the lattice spacing, and much smaller than the size of the domain, the process of boundary avalanches is approximatively scale invariant. Then, the distribution of the sizes of the avalanches must be a power law for the range $1 \ll n \ll L^2$, and then must be truncated by the finiteness of the domain, i.e.

$$p_L(n) \sim \begin{cases} n^{-\gamma} & n \ll L^2 \\ 0 & n \gg L^2 \end{cases} \quad (13)$$

The value of γ , unknown up to this point, can now be determined: indeed we know that $\sum_n n p_L(n) \sim L$, and such a behaviour is compatible with a single value of γ , namely $\gamma = \frac{3}{2}$. In other words, we expect that $\mathbb{P}_L(n(P) > n) \sim n^{-\frac{1}{2}}$ for $1 \ll n \ll L^2$.

This sketchy prediction seems numerically verified, but somewhat “for the wrong reasons”. A more detailed description of the truly scale-invariant process of boundary avalanches should be given in a regime in which the geometry of the cylinder does not introduce a new finite parameter in the model, that is, in a regime $L_y \gg L \gg 1$ (in this section it is convenient to adopt the notation $L_x = L$). In this case we do not see anymore the effect of the top boundary of the cylinder, or a difference between the cylinder and the folded-cylinder geometries, and we shall expect that there exists almost surely one “giant avalanche”, occupying a fraction $1 - \mathcal{O}(L/L_y)$ of the volume, so that the probability distribution may take the form

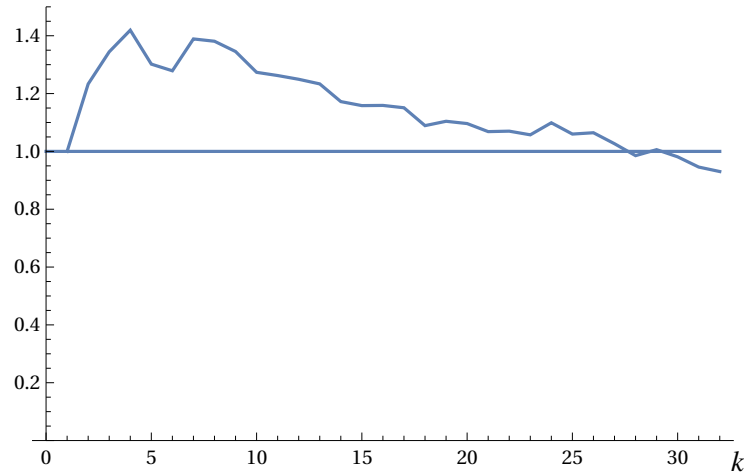


Figure 2: Averages of the k -th largest polyomino in a process (except the giant one), multiplied by k^2 , and rescaled so that the first value is 1, on the data presented in Figure 3. A far-fetching conjecture based on the formula (12) would suggest that this function is 1, up to values of $k \ll L_x$.

(calling $V = LL_y$, the volume)

$$p_L(n) \sim \begin{cases} n^{-\gamma} & n \ll L^2 \\ o(n^{-\gamma}) & L^2 \ll n \ll V \\ p'_L(n) & n = V - \mathcal{O}(L^2) \\ 0 & n > V \end{cases} \quad (14)$$

where $\sum_n p'_L(n) = 1/L$, and again we must have $\gamma > 1$. This implies for the average

$$\frac{V}{L} = \mathbb{E}(|P|) = \frac{V - \Theta(L^2)}{L} + \int^{L^2} dx x^{1-\gamma} = \frac{V}{L} - \Theta(L) + \Theta((L^2)^{2-\gamma}) \quad (15)$$

which requires

$$1 = 2(2 - \gamma) \quad (16)$$

that is, again $\gamma = 3/2$.

It is not completely evident that, except for the trivial giant avalanche, the process of boundary avalanches occupies a height of order L of the domain, and that the second largest avalanche is on a scale $\sim L^2$. However, this can be established through the calculation, performed in (12), of the average number of interfaces between pairs of polyominoes that reach height y , which scales as $L/(\pi y)$. So, this average goes from $\gg 1$ to $\ll 1$ when y goes from much smaller than L/π to much larger than L/π . As avalanches have possibly fractal boundaries, but their interior has Hausdorff dimension 2, we deduce that the second largest avalanche must have a volume on the scale $\sim L^2$. Then, as the appearance of each further avalanche approximately adds one to the number of interfaces, from the behaviour in L/y of our observable we may deduce that the average sizes of avalanches listed in decreasing order (and excluding the giant one) may form a sequence not too far from the series CL^2/k^2 , for some constant C , up to values of k so that the avalanches have macroscopic sizes. It is remarkable that such a far-fetching prediction is vaguely in accordance with the numerics, even at relatively small sizes (cf. Figure 2).

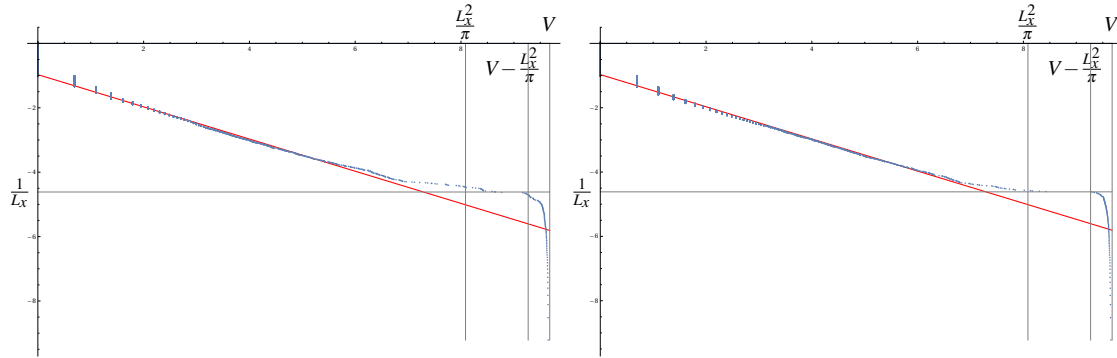


Figure 3: Left: plot of the ordered list of 10^4 avalanche sizes, for a folded-cylinder geometry on the square lattice, of size 101×158 . We adopt a log-log plot, with a superposed red line of slope $-1/2$, which highlights the validity of the ansatz in equation (14) in this case. Right: plot of the ordered list of $L_x \times 10^2 = 10100$ avalanche sizes, for 100 realisations of the permutation process.

Note that, as yet another consequence of the properties of the ASM, the probability distribution for the avalanche process, shown in the bottom of Figure 3, and for the single-site boundary avalanches, shown in the top of the same figure, are essentially coincident (except for the fact that the fraction of giant avalanches in the first case is exactly equal to $1/L$, while in the second case it is only approximately equal to this value, with Gaussian fluctuations on a scale compatible with an approximation of independent events). Indeed, as explained above, the coincidence of these two distributions is implied by the principles of the Abelian Sandpile Model, while the scaling ansatz only concerns the determination of the qualitative properties of this function.

Arguments of this type have been a *leitmotif* of this paper: the relation between apparently different boundary avalanche processes has allowed us to deduce fine statistical properties for each of them (and in particular for the most basic procedure, of a single-site boundary avalanche), by using each time the most convenient formulation. Without using this multiplicity of definitions, we wouldn't have been able to perform most of our calculations.

References

- [1] Per Bak, Chao Tang & Kurt Wiesenfeld (1987): *Self-organized criticality: An explanation of the $1/f$ noise*. *Phys. Rev. Lett.* 59, pp. 381–384, doi:10.1103/PhysRevLett.59.381.
- [2] Per Bak, Chao Tang & Kurt Wiesenfeld (1988): *Self-organized criticality*. *Phys. Rev. A* 38, pp. 364–374, doi:10.1103/PhysRevA.38.364.
- [3] R. Ball, M. Nauenberg & T. A. Witten (1984): *Diffusion-controlled aggregation in the continuum approximation*. *Phys. Rev. A* 29, pp. 2017–2020, doi:10.1103/PhysRevA.29.2017.
- [4] Gunnar Brinkmann, Gilles Caporossi & Pierre Hansen (2003): *A Survey and New Results on Computer Enumeration of Polyhex and Fusene Hydrocarbons*. *Journal of Chemical Information and Computer Sciences* 43(3), pp. 842–851, doi:10.1021/ci025526c.
- [5] Jon Brunvoll, Björg N. Cyvin & Sven J. Cyvin (1992): *Benzenoid chemical isomers and their enumeration*, pp. 181–221. Springer Berlin Heidelberg, Berlin, Heidelberg, doi:10.1007/BFb0018564.

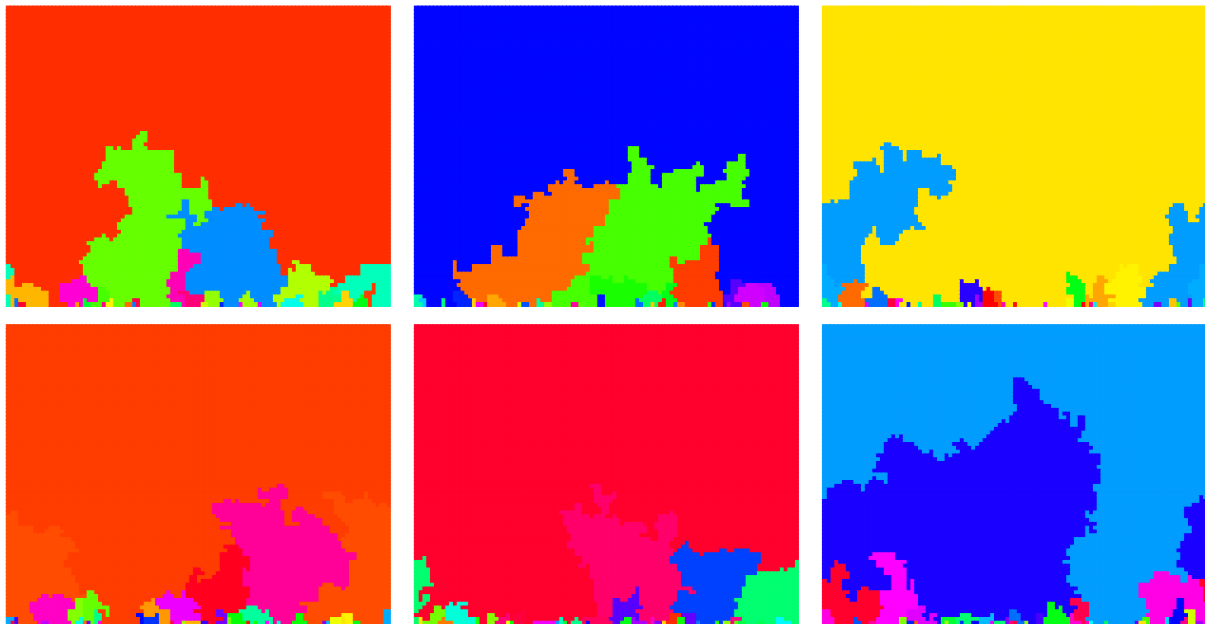


Figure 4: Some examples of realisations of the permutation boundary avalanche process on the square lattice in a folded-cylinder geometry with $L_x = 101$ (only the most relevant part of the cylinder is shown). The hue value of the colour describes the position of the corresponding boundary edge in the permutation σ .

- [6] S. Caracciolo, G. Paoletti & A. Sportiello (2012): *Multiple and inverse topplings in the Abelian Sandpile Model*. *The European Physical Journal Special Topics* 212(1), pp. 23–44, doi:10.1140/epjst/e2012-01652-9.
- [7] Sergio Caracciolo & Andrea Pelissetto (1998): *Corrections to finite-size scaling in the lattice N -vector model for $N = \infty$* . *Phys. Rev. D* 58, p. 105007, doi:10.1103/PhysRevD.58.105007.
- [8] Sergio Caracciolo, Alan D Sokal & Andrea Sportiello (2007): *Grassmann integral representation for spanning hyperforests*. *Journal of Physics A: Mathematical and Theoretical* 40(46), p. 13799, doi:10.1088/1751-8113/40/46/001.
- [9] Björg N. Cyvin, Jon Brunvoll & Sven J. Cyvin (1992): *Enumeration of benzenoid systems and other polyhexes*, pp. 65–180. Springer Berlin Heidelberg, Berlin, Heidelberg, doi:10.1007/BFb0018563.
- [10] C. De Grandi (2005): *Fermionic field theory for trees and forests on the triangular lattice*. Laurea thesis, Univ. degli Studi di Milano. Available at <https://pcteserver.mi.infn.it/~caraccio/Lauree/DeGrandi.pdf>. Note.
- [11] D Dhar, P Ruelle, S Sen & D N Verma (1995): *Algebraic aspects of Abelian sandpile models*. *Journal of Physics A: Mathematical and General* 28(4), p. 805, doi:10.1088/0305-4470/28/4/009.
- [12] Deepak Dhar (1999): *The Abelian sandpile and related models*. *Physica A: Statistical Mechanics and its Applications* 263(1), pp. 4–25, doi:10.1016/S0378-4371(98)00493-2. Proceedings of the 20th IUPAP International Conference on Statistical Physics.
- [13] M. Eden (1961): *A Two-Dimensional Growth Process*. In F. Neyman, editor: *Proceeding of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, IV*, University of California, p. 223.
- [14] Ian G. Enting & Iwan Jensen (2009): *Exact Enumerations*, pp. 143–179. Springer Netherlands, Dordrecht, doi:10.1007/978-1-4020-9927-4.

- [15] Martin Gardner (1960): *MATHEMATICAL GAMES*. *Scientific American* 203(5), pp. 186–201, doi:10.1038/scientificamerican1160-186. Available at <http://www.jstor.org/stable/24940703>.
- [16] Solomon W. Golomb (1994): *Puzzles, Patterns, Problems, and Packings*. Princeton University Press, Princeton, doi:10.1515/9780691215051.
- [17] Branko Grünbaum & Geoffrey Colin Shephard (1989): *Tilings and patterns. An introduction*. New York: W. H. Freeman and Company.
- [18] Anthony J. Guttmann (2009): *History and Introduction to Polygon Models and Polyominoes*, pp. 1–21. Springer Netherlands, Dordrecht, doi:10.1007/978-1-4020-9927-4.
- [19] Frank Harary & Edgar M. Palmer (1973): *Graphical Enumeration*. Academic Press, doi:10.1016/B978-0-12-324245-7.50008. Available at <https://www.sciencedirect.com/science/article/pii/B978012324245750008>.
- [20] Claude Itzykson & Jean-Michel Drouffe (1989): *Statistical Field Theory. Volume 1: From Brownian Motion to Renormalization and Lattice Gauge Theory*. Cambridge Monographs on Mathematical Physics, Cambridge University Press, doi:10.1017/CBO9780511622786.
- [21] E.V. Ivashkevich, D.V. Ktitarov & V.B. Priezzhev (1994): *Waves of topplings in an Abelian sandpile*. *Physica A: Statistical Mechanics and its Applications* 209(3), pp. 347–360, doi:10.1016/0378-4371(94)90188-0.
- [22] Iwan Jensen (2001): *Enumerations of Lattice Animals and Trees*. *Journal of Statistical Physics* 102(3), pp. 865–881, doi:10.1023/A:1004855020556.
- [23] Iwan Jensen & Anthony J Guttmann (2000): *Statistics of lattice animals (polyominoes) and polygons*. *Journal of Physics A: Mathematical and General* 33(29), pp. L257–L263, doi:10.1088/0305-4470/33/29/102.
- [24] D. V. Ktitarov & V. B. Priezzhev (1998): *Expansion and contraction of avalanches in the two-dimensional Abelian sandpile*. *Phys. Rev. E* 58, pp. 2883–2888, doi:10.1103/PhysRevE.58.2883.
- [25] Martin Lüscher & Peter Weisz (1995): *Coordinate space methods for the evaluation of Feynman diagrams in lattice field theories*. *Nuclear Physics B* 445(2), pp. 429–450, doi:10.1016/0550-3213(95)00185-U.
- [26] S N Majumdar & D Dhar (1991): *Height correlations in the Abelian sandpile model*. *Journal of Physics A: Mathematical and General* 24(7), p. L357, doi:10.1088/0305-4470/24/7/008.
- [27] S.N. Majumdar & Deepak Dhar (1992): *Equivalence between the Abelian sandpile model and the $q \rightarrow 0$ limit of the Potts model*. *Physica A: Statistical Mechanics and its Applications* 185(1), pp. 129–145, doi:10.1016/0378-4371(92)90447-X.
- [28] S.S. Manna (1991): *Critical exponents of the sand pile models in two dimensions*. *Physica A: Statistical Mechanics and its Applications* 179(2), pp. 249–268, doi:10.1016/0378-4371(91)90063-L.
- [29] P Meakin (1988): *Models for Colloidal Aggregation*. *Annual Review of Physical Chemistry* 39(Volume 39), pp. 237–267, doi:10.1146/annurev.pc.39.100188.001321.
- [30] Wolfgang R. Müller, Klaus Szymanski, Jan von Knop & Nenad Trinajstić (1993): *On the number of square-cell configurations*. *Theoretica chimica acta* 86, pp. 269–278, doi:10.1007/BF01130823.
- [31] Maya Paczuski & Stefan Boettcher (1997): *Avalanches and waves in the Abelian sandpile model*. *Phys. Rev. E* 56, pp. R3745–R3748, doi:10.1103/PhysRevE.56.R3745.
- [32] V. B. Priezzhev, D. V. Ktitarov & E. V. Ivashkevich (1996): *Formation of Avalanches and Critical Exponents in an Abelian Sandpile Model*. *Phys. Rev. Lett.* 76, pp. 2093–2096, doi:10.1103/PhysRevLett.76.2093.
- [33] James Gary Propp & David Bruce Wilson (1998): *How to Get a Perfectly Random Sample from a Generic Markov Chain and Generate a Random Spanning Tree of a Directed Graph*. *Journal of Algorithms* 27(2), pp. 170–217, doi:10.1006/jagm.1997.0917.
- [34] Frank Redig (2006): *Mathematical Aspects of the Abelian Sandpile Model*. In Anton Bovier, François Dunlop, Aernout van Enter, Frank den Hollander & Jean Dalibard, editors: *Mathematical Statistical Physics, Les Houches* 83, Elsevier, pp. 657–729, doi:10.1016/S0924-8099(06)80051-X.
- [35] Dong-Shin Shin (1998): *Application of a coordinate-space method for the evaluation of lattice Feynman diagrams in two dimensions*. *Nuclear Physics B* 525(1), pp. 457–482, doi:10.1016/S0550-3213(98)00232-6.

- [36] Dong-Shin Shin (1999): *Correction to four-loop RG functions in the two-dimensional lattice $O(n)$ σ -model*. *Nuclear Physics B* 546(3), pp. 669–690, doi:10.1016/S0550-3213(99)00020-6.
- [37] Nenad Trinajstić (1992): *On the classification of polyhexes*. *Journal of Mathematical Chemistry* 9, pp. 373–380, doi:10.1007/BF01166101.
- [38] Nenad Trinajstić, Jan von Knop, Wolfgang R. Müller, Konrad Syzmański & Sonja Nikolić (1991): *Computational Chemical Graph Theory: Characterization, Enumeration and Generation of Chemical Structures by Computer Methods*. Ellis Horwood.
- [39] S.G. Whittington & C.E. Soteros (1990): *Lattice Animals: Rigorous Results and Wild Guesses*. In Geoffrey Grimmett & Dominic Welsh, editors: *Disorder in Physical Systems: A Volume in Honour of John M. Hammersley*, Oxford University Press, Oxford, pp. 323–335. Available at <http://www.statslab.cam.ac.uk/~grg/books/jmh.html>.
- [40] T. A. Witten & L. M. Sander (1981): *Diffusion-Limited Aggregation, a Kinetic Critical Phenomenon*. *Phys. Rev. Lett.* 47, pp. 1400–1403, doi:10.1103/PhysRevLett.47.1400.
- [41] Jean Zinn-Justin (2002): *Quantum Field Theory and Critical Phenomena; 4th ed.* International series of monographs on physics, Clarendon Press, Oxford, doi:10.1093/acprof:oso/9780198509233.001.0001.

Construction of Minkowski Sums by Cellular Automata

Pierre-Adrien Tahay

Université de Lorraine, Loria, UMR 7503, F-54506 Vandœuvre-lès-Nancy, France

pierre-adrien.tahay@univ-lorraine.fr

We give a construction in a column of a one-dimensional cellular automaton of the Minkowski sum of two sets which can themselves occur in columns of cellular automata. It enables us to obtain another construction of the set of integers that are sums of three squares, answering a question by the same author in [8].

1 Introduction

A one-dimensional *cellular automaton* (CA) is a dynamical system $(\mathcal{A}^{\mathbb{Z}}, F)$, where \mathcal{A} is a finite set, and where the map $F : \mathcal{A}^{\mathbb{Z}} \rightarrow \mathcal{A}^{\mathbb{Z}}$ is defined by a local rule acting uniformly and synchronously on the configuration space. More precisely, there exists an integer $r \geq 0$ called the *radius* of the CA, and a *local rule* $f : \mathcal{A}^{2r+1} \rightarrow \mathcal{A}$ such that

$$F(x)_k = f((x_{k+i})_{-r \leq i \leq r}), \text{ for all } x \in \mathcal{A}^{\mathbb{Z}}, \text{ and for all } k \in \mathbb{Z}.$$

By the Curtis-Hedlund-Lyndon theorem, a map $F : \mathcal{A}^{\mathbb{Z}} \rightarrow \mathcal{A}^{\mathbb{Z}}$ is a CA if and only if it is continuous with respect to the product topology, and it commutes with the shift map σ defined by

$$\sigma(x)_k = x_{k-1}, \text{ for all } x \in \mathcal{A}^{\mathbb{Z}}, \text{ and for all } k \in \mathbb{Z}.$$

A cellular automaton can be visualized by using a spacetime diagram consisting of a 2-dimensional grid where each cell contains an element of \mathcal{A} and is represented by a space and time coordinate.

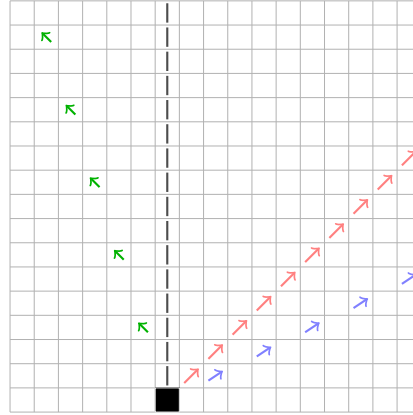
The problem of representing a word (a sequence over a finite alphabet) in a column of the spacetime diagram of a cellular automaton is an interesting one but relatively unexplored. One of the oldest results on the subject is the construction of the characteristic function of prime numbers by Fischer in 1965, using a cellular automaton with more than 30,000 states [3]. The number of states was considerably reduced by Korec in 1997 who provided another construction with only 11 states [4]. Following on from Fischer's work, Mazoyer and Terrier have established various results on words that can be realized as a column of a CA, which they call *Fisher's constructible function* [6] (I think Fischer's name is misspelled as Fisher throughout their article, including when they cite [3] in the bibliography.)

Afterwards, several widely known words have been obtained as column of a CA. In 2015, Rowland and Yassawi, established an effective construction for all p -automatic sequences, for any prime number p , as column of a CA by using generating series and the theory of finite fields [7]. In 2022, Dolce and Tahay [2] obtained a construction for all Sturmian words having quadratic slope using the directive sequences and their ultimate periodicity. Other constructions have been obtained by Marcovici, Stoll and Tahay in 2018 [5], such as the characteristic function of any polynomial $P \in \mathbb{Q}[X]$ of degree $d \geq 1$ with $P(\mathbb{N}) \subset \mathbb{N}$.

In this paper we establish the constructibility as a column of a CA of the Minkowski sum of two constructible sets. This generalizes the method used by the author for constructing the characteristic function of the set of integers that are sums of two squares from the construction of the characteristic function of the squares [8].

2 Signals in cellular automata

In their paper [6], Mazoyer and Terrier give some constructions such as the sum of two constructible functions, the linear combination of constructible functions, or constant-recursive sequences. They use *signals* to obtain their various constructions.



- vertical signal - slope 1 - slope -3 - slope 1/2

Figure 1: some instances of signals

In the spacetime diagram of a CA, signals are a way to transmit information between two cells, by connecting two cells (m, n) and $(m', n + t)$. The *slope* of the signal is the number $\frac{t}{m' - m}$.

3 Minkowski sums

Let A and B be two sets. Recall that the *Minkowski sum* of A and B is

$$A + B = \{a + b \mid a \in A, b \in B.\}$$

Definition 1. A set A is called *constructible* by a CA if the characteristic sequence $\mathbf{1}_A$ of A is obtained as a column of a CA.

Theorem 2. Let A and B be two sets constructible by some cellular automata. Then the set $A + B$ is also constructible by a cellular automaton.

Proof. Let us call G the cellular automaton that will build $\mathbf{1}_{A+B}$. Let us call F the cellular automaton that constructs A in the left column of the CA. We compute the cellular automaton $\sigma \circ F$ and a signal of slope 1 in the diagonal of the spacetime diagram of G , so that we can mark the elements of A on the diagonal. From these marked cells, we send vertical signals. Now we compute the set B in the left column of G . From each element b of B in this left column we send a signal of slope 1/2. When this signal meets a vertical signal in a column of rank a for any $a \in A$ we define a signal of slope $-1/2$. When this signal meets the left column we are on the line of rank $b + a$. Since

$$A + B = \bigcup_{a \in A} \{a + b, b \in B\},$$

the final set obtained in the left column is therefore $A + B$.

Note that signals with slope $1/2$ can meet columns of rank a , with $a \in A$ below the vertical signal, in which case some elements $a + b$ could be missing. If there is only a finite number of these, then we define the first lines of the cellular automaton as initial conditions. If there is an infinite number of these, we change the signals of slopes $1/2$ and $-1/2$ by signals of slopes 1 and -1 . Thus, we build an element of the form $b + 2a$ with $b \in B$ and $a \in A$, but we can recover the elements of the form $a + b$ with $a \in A$ and $b \in B$ by using the cellular automaton G^2 . \square

4 Examples

Let S be the set of squares, i.e. $S = \{n^2, n \in \mathbb{N}\}$. So $S + S$ is the set of integers that are sums of two squares and $S + (S + S)$ the set of the integers that are sums of three squares. We recall below the constructions of S and $S + S$ obtained in [8].

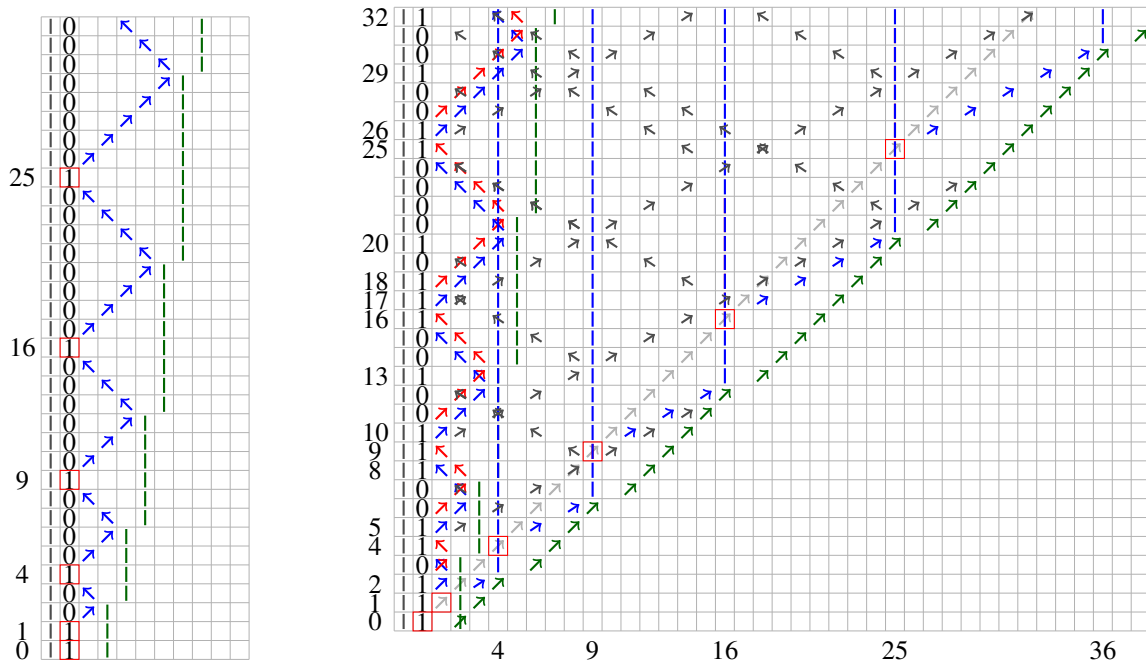


Figure 2: CA for S (left) and $S + S$ (right)

The construction of S (see Figure 2) is due to Delacourt, Poupet, Sablik and Theyssier [1]. From a cell containing a 1 in the left column, we send a signal of slope 1. When this signal meets a wall (vertical signal in green) we send a signal of slope -1 and the wall is shifted one cell to the right and continues to spread vertically. The signal of slope -1 marks the next square when it meets the left column.

For $S + S$ (see Figure 2), we recall the construction previously obtained by the author in [8]. The principle is to start by constructing integers which are sums of two squares of the form $n^2 + 0^2$ and $n^2 + 1^2$ by using the same construction as for the squares. For the other elements of $S + S$ we mark the columns whose horizontal coordinate is a perfect square by using a method developed by Marcovici,

Stoll and Tahay [5] for the polynomial sequences. From each marked column we define a blue wall and from each perfect square in the left column we send a signal of slope 1/2. When these signals meet a blue wall, we define a new signal a slope $-1/2$ which marks an element of $S + S$ when he meets the left column.

From these two constructions and Theorem 2 we give a new construction of $S + (S + S)$ in Figure 3 using signals which answers the second open question in [8]. Note that in the figure, the first five lines are initial conditions.

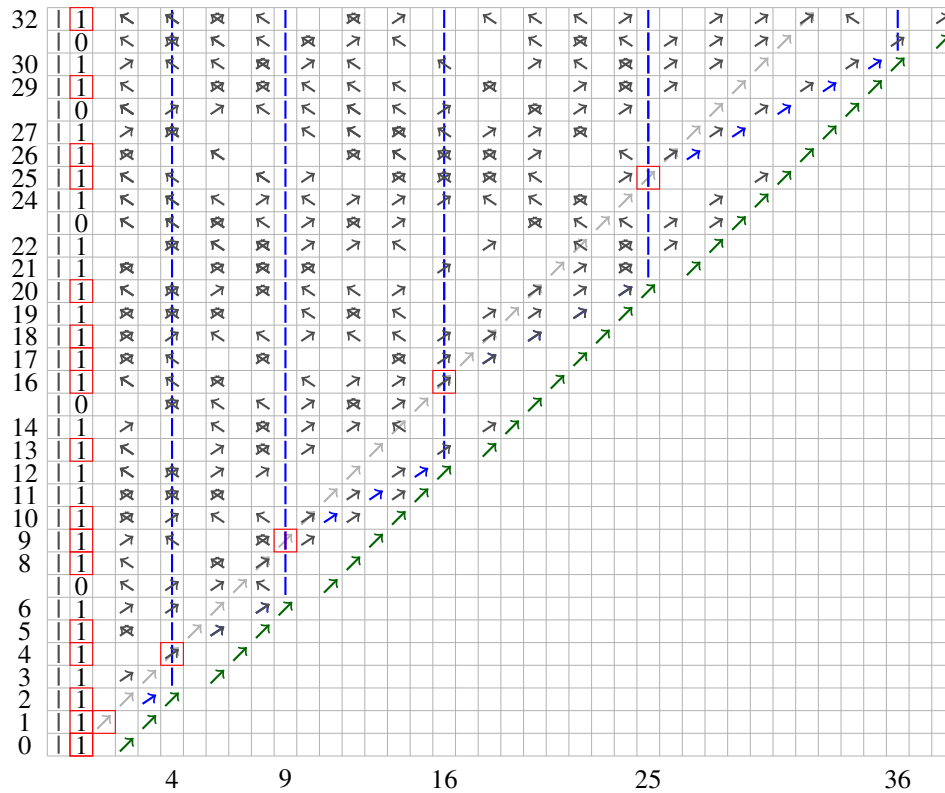


Figure 3: Construction of $S + (S + S)$ by a cellular automaton

Acknowledgment The author thanks Guillaume Theyssier for asking him about the constructibility of Minkowski sums from those of the set of integers that are sums of two squares, that the author presented during a conference at CIRM on 29th February 2024.

References

[1] M. Delacourt, V. Poupet, M. sablik & G. Theyssier (2011): *Directional Dynamics along Arbitrary Curves in Cellular Automata*. *Theoretical Computer Science* 412(30), doi:10.1016/j.tcs.2011.02.019.

[2] F. Dolce & P.-A. Tahay (2022): *Column Representation of Sturmian Words in Cellular Automata*. In V. Diekert & M. Volkov, editors: *Developments in Language Theory*, 13257, Springer International Publish-

- ing, Cham, pp. 127–138, doi:10.1007/978-3-031-05578-2_10. Available at <https://hal.science/hal-04066361v1/document>. Series Title: Lecture Notes in Computer Science.
- [3] P. C. Fischer (1965): *Generation of Primes by a One-Dimensional Real-Time Iterative Array*. *Journal of the ACM* 12(3), doi:10.1145/321281.321290.
- [4] I. Korec (1997): *Real-Time Generation of Primes by a One-Dimensional Cellular Automaton with 11 States*. In G. Goos, J. Hartmanis, J. van Leeuwen, I. Prívvara & P. Ružička, editors: *Mathematical Foundations of Computer Science 1997*, 1295, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 358–367, doi:10.1007/BFb0029979. Available at <https://link.springer.com/chapter/10.1007/BFb0029979>. Series Title: Lecture Notes in Computer Science.
- [5] I. Marcovici, T. Stoll & P.-A. Tahay (2018): *Construction of Some Nonautomatic Sequences by Cellular Automata*. In J. M. Baetens & M. Kutrib, editors: *Cellular Automata and Discrete Complex Systems*, 10875, Springer International Publishing, Cham, pp. 113–126, doi:10.1007/978-3-319-92675-9_9. Available at <https://inria.hal.science/hal-01824876/document>. Series Title: Lecture Notes in Computer Science.
- [6] J. Mazoyer & V. Terrier (1999): *Signals in One-Dimensional Cellular Automata*. *Theoretical Computer Science* 217(1), doi:10.1016/S0304-3975(98)00150-9.
- [7] E. Rowland & R. Yassawi (2015): *A Characterization of p -Automatic Sequences as Columns of Linear Cellular Automata*. *Advances in Applied Mathematics* 63, doi:10.1016/j.aam.2014.10.002.
- [8] P.-A. Tahay (2023): *Characteristic Sequences of the Sets of Sums of Squares as Columns of Cellular Automata*. In A. Frid & R. Mercas, editors: *Combinatorics on Words*, 13899, Springer Nature Switzerland, pp. 288–300, doi:10.1007/978-3-031-33180-0_22. Available at <https://hal.science/hal-04504166v1/document>. Series Title: Lecture Notes in Computer Science.

Local Limit Theorems for q -Multinomial and Multiple Heine Distributions

Malvina Vamvakari

Department of Informatics and Telematics
Harokopio University of Athens
Greece

mvamv@hua.gr

In this work we establish local limit theorems for q -multinomial and multiple Heine distributions. Specifically, the pointwise convergence of the q -multinomial distribution of the first kind, as well as for its discrete limit, the multiple Heine distribution, to a multivariate Stieltjes-Wigert type distribution, are provided.

1 Brief Introduction

Recently, Vamvakari [8] introduced multivariate discrete q -distributions. Specifically, she derived a multivariate absorption distribution as a conditional distribution of a Heine process at a finite sequence of q -points in a time interval which had been defined by Kyriakoussis and Vamvakari [6]. Also, she deduced a multivariate q -hypergeometric distribution, as a conditional distribution of the multivariate absorption distribution.

Afterwards, Charalambides [2, 3] introduced in detail q -multinomial, negative q -multinomial, multivariate q -Pólya and inverse q -Pólya distributions and also examined their limiting discrete distributions. Analytically, he considered a stochastic model of a sequence of independent Bernoulli trials with chain-composite successes (or failures), where the odds of success of a certain kind at a trial is assumed to vary geometrically, with rate q , with the number of previous trials and introduced the q -multinomial and negative q -multinomial distributions of the first kind as well as their discrete limit, multivariate Heine distribution. Also, he considered a stochastic model of a sequence of independent Bernoulli trials with chain-composite successes (or failures), where the probability of success of a certain kind at a trial varies geometrically, with rate q , with the number of previous successes and introduced the q -multinomial and negative q -multinomial distributions of the second kind as well as their discrete limit, multivariate Euler distribution.

Kyriakoussis and Vamvakari [4, 5, 7] studied the continuous limiting behaviour of the univariate discrete q -distributions. Analytically, they established the pointwise convergence of the q -binomial and the negative q -binomial distributions of the first kind, as well as of the Heine distribution, to a deformed Stieltjes-Wigert continuous one. Moreover, they proved the pointwise convergence of the q -binomial and the negative q -binomial distributions of the second kind, as well as of the Euler distribution, to a deformed Gaussian one.

Vamvakari [9] initiated the study of continuous limiting behaviour of multivariate discrete q -distributions inspired by the limiting behaviour of the univariate ones. Specifically, she studied the asymptotic behavior of the univariate, bivariate and multivariate absorption discrete q -distributions. The pointwise convergence of the univariate absorption distribution to a deformed Gaussian one and that of the bivariate and multivariate absorptions to a bivariate and multivariate deformed Gaussian ones, have been provided.

The aim of this work is to study further the continuous limiting behaviour of multivariate discrete q -distributions. Local limit theorems for the q -multinomial and multiple Heine distributions are established. Specifically, the pointwise convergence of the q -multinomial and its discrete limit, the multiple Heine distribution, to a multivariate Stieltjes-Wigert type distribution are provided.

2 Preliminaries

Charalambides [1] had studied in details the q -binomial distribution of the first kind and its discrete limit, the Heine Distribution with probability functions (p.f.) given by

$$f_X^B(x) = \binom{n}{x}_q q^{\binom{x}{2}} \theta^x \prod_{j=1}^n (1 + \theta q^{j-1})^{-1}, \quad x = 0, 1, \dots, n, \tag{1}$$

where $\theta > 0$ and $0 < q < 1$ and

$$f_X^H(x) = e_q(-\lambda) \frac{q^{\binom{x}{2}} \lambda^x}{[x]_q!}, \quad x = 0, 1, 2, \dots, \quad 0 < q < 1, \quad 0 < \lambda < \infty, \tag{2}$$

where

$$e_q(z) := \sum_{n=0}^{\infty} \frac{(1-q)^n z^n}{(q; q)_n} = \sum_{n=0}^{\infty} \frac{z^n}{[n]_q!} = \frac{1}{((1-q)z; q)_{\infty}}, \quad |z| < 1, \tag{3}$$

and

$$[n]_q! = [1]_q [2]_q \cdots [n]_q = \prod_{k=1}^n \frac{1-q^k}{(1-q)^n} = \frac{(q; q)_n}{(1-q)^n}, \quad 0 < q < 1, \quad [t]_q = \frac{1-q^t}{1-q}. \tag{4}$$

Kyriakoussis and Vamvakari [4, 5] proved limit theorems among others for the q -Binomial distribution of the first kind and Heine distribution for constant q , by using pointwise convergence in a “ q -analogous sense” of the classical de Moivre–Laplace limit theorem. Specifically for the needs of their study they established a q -Stirling formula for $n \rightarrow \infty$ of the q -factorial of order n , defined by relation (4). Analytically, for $0 < q < 1$ constant, it was proved that,

$$[n]_q! = \frac{q^{-1/8} (2\pi(1-q))^{1/2}}{(q \log q^{-1})^{1/2}} \frac{q^{\binom{n}{2}} q^{-n/2} [n]_{1/q}^{n+1/2}}{\prod_{j=1}^{\infty} (1 + (q^{-n} - 1)q^{j-1})} (1 + O(n^{-1})). \tag{5}$$

Next, the pointwise convergence of the q -Binomial distribution of the first kind to a deformed continuous Stieltjes–Wigert distribution was established. The continuous Stieltjes–Wigert distribution has probability density function

$$v_W^{SW}(w) = \frac{q^{1/8}}{\sqrt{2\pi \log q^{-1} w}} e^{\frac{(\log w)^2}{2 \log q}}, \quad w > 0, \tag{6}$$

with mean value $\mu^{SW} = q^{-1}$ and standard deviation $\sigma^{SW} = q^{-3/2}(1-q)^{1/2}$.

Transferred from the random variable X of the q -Binomial distribution (1) to the equal-distributed deformed random variable $Y = [X]_{1/q}$ and for $n \rightarrow \infty$, the q -Binomial distribution of the first kind was approximated by a deformed standardized continuous Stieltjes–Wigert distribution as follows:

$$f_X^B(x) \cong \frac{q^{-7/8}}{\sigma_q (2\pi)^{1/2}} \left(\frac{\log q^{-1}}{q^{-1} - 1} \right)^{1/2} \left(q^{-3/2} (1-q)^{1/2} \frac{[x]_{1/q} - \mu_q}{\sigma_q} + q^{-1} \right)^{-1/2} q^{-x} \cdot \exp \left(\frac{1}{2 \log q} \log^2 \left(q^{-3/2} (1-q)^{1/2} \frac{[x]_{1/q} - \mu_q}{\sigma_q} + q^{-1} \right) \right), \quad x \geq 0, \tag{7}$$

where $\theta = \theta_n$, for $n = 0, 1, 2, \dots$, such that $\theta_n = q^{-\alpha n}$ with $0 < a < 1$ constant and μ_q and σ_q^2 the mean value and variance of the random variable Y , respectively. A similar asymptotic result has been provided for the Heine distribution when $\lambda \rightarrow \infty$.

3 Main Results

Let X_j be the number of successes of a j th kind in a sequence of n independent Bernoulli trials with chain composite failures, where the probability of success of the j th kind at the i th trial is given by

$$p_{j,i} = \frac{\theta_j q^{i-1}}{1 + \theta_j q^{i-1}}, \quad 0 < \theta_j < \infty, \quad j = 1, 2, \dots, i = 1, 2, \dots, 0 < q < 1 \text{ or } 1 < q < \infty.$$

Then the joint probability function of the random vector $\mathcal{X} = (X_1, X_2, \dots, X_k)$ is given by

$$f_{\mathcal{X}}^B(x_1, x_2, \dots, x_k) = P(X_1 = x_1, X_2 = x_2, \dots, X_k = x_k) = \binom{n}{x_1, x_2, \dots, x_k}_q \prod_{j=1}^k \frac{\theta_j^{x_j} q^{\binom{x_j}{2}}}{\prod_{i=1}^{n-s_{j-1}} (1 + \theta_j q^{i-1})} \quad (8)$$

$x_j = 0, 1, 2, \dots, n, \sum_{j=1}^k x_j \leq n, s_j = \sum_{i=1}^j x_i, 0 < \theta_j < 1, j = 1, 2, \dots, k$, and $0 < q < 1$ or $1 < q < \infty$. This discrete q -distribution is known as a q -multinomial distribution (see Charalambides [2]).

The discrete limit of the joint p.f. of the q -multinomial distribution of the 1st kind, as $n \rightarrow \infty$, is the joint p.f. of the multiple Heine distribution,

$$\lim_{n \rightarrow \infty} \binom{n}{x_1, x_2, \dots, x_k}_q \prod_{j=1}^k \frac{\theta_j^{x_j} q^{\binom{x_j}{2}}}{\prod_{i=1}^{n-s_{j-1}} (1 + \theta_j q^{i-1})} = \prod_{j=1}^k \frac{q^{\binom{x_j}{2}} \lambda_j^{x_j}}{[x_j]_q!} \prod_{i=1}^{\infty} (1 + \lambda_j (1 - q) q^{i-1})^{-1} \quad (9)$$

$x_j = 0, 1, 2, \dots, \lambda_j > 0, 0 < q < 1, \lambda_j = \theta_j / (1 - q), j = 1, 2, \dots, k$ (see Charalambides [2]).

Next we will study the continuous limiting behaviour of the q -trinomial distribution. Let (X_1, X_2) be the discrete bivariate random variable with joint probability function

$$f_{X_1, X_2}^B(x_1, x_2) = P(X_1, X_2) = \binom{n}{x_1, x_2}_q \frac{\theta_1^{x_1} \theta_2^{x_2} q^{\binom{x_1}{2} + \binom{x_2}{2}}}{\prod_{i=1}^n (1 + \theta_1 q^{i-1}) \prod_{i=1}^{n-x_1} (1 + \theta_2 q^{i-1})} \quad (10)$$

$x_j = 0, 1, 2, \dots, n, j = 1, 2, x_1 + x_2 \leq n, 0 < \theta_1, \theta_2 < 1$ and $0 < q < 1$ or $1 < q < \infty$. The distribution of the bivariate random variable (X_1, X_2) is known as a q -trinomial distribution.

The marginal probability function of the random variable X_1 , is distributed according to the q -binomial of the 1st kind with probability function

$$f_{X_1}^B(x_1) = \binom{n}{x_1}_q \frac{\theta_1^{x_1} q^{\binom{x_1}{2}}}{\prod_{i=1}^n (1 + \theta_1 q^{i-1})}, \quad x_1 = 0, 1, 2, \dots, n.$$

The mean and the variance of the deformed variable $[X_1]_{1/q}$ are given by

$$\begin{aligned} \mu_{[X_1]_{1/q}} &= E([X_1]_{1/q}) = [n]_q \frac{\theta_1}{1 + \theta_1 q^{n-1}} \\ &\text{and} \\ (\sigma_{[X_1]_{1/q}})^2 &= V([X_1]_{1/q}) = \frac{1-q}{q} [n]_q^2 \frac{\theta_1^2}{(1 + \theta_1 q^{n-1})^2 (1 + \theta_1 q^{n-2})} + [n]_q \frac{\theta_1}{(1 + \theta_1 q^{n-1})(1 + \theta_1 q^{n-2})} \end{aligned} \quad (11)$$

respectively.

The conditional random variable $X_2|X_1$, is distributed according to the univariate q -binomial of the 1st kind with probability function

$$f_{X_2|X_1}^B(x_2|x_1) = \binom{n-x_1}{x_2} \frac{\theta_2^{x_2} q^{\binom{x_2}{2}}}{q \prod_{i=1}^{n-x_1} (1 + \theta_2 q^{i-1})}, \quad x_2 = 0, 1, 2, \dots, n-x_1.$$

The conditional mean and conditional variance of the deformed variable $[X_2]_{1/q}$ given $X_1 = x_1$, are given by

$$\begin{aligned} \mu_{[X_2]_{1/q}|X_1} &= E([X_2]_{1/q}|X_1) = [n-x_1]_q \frac{\theta_2}{1 + \theta_2 q^{n-x_1-1}}, \\ \text{and} & \\ (\sigma_{[X_2]_{1/q}|X_1})^2 &= V([X_2]_{1/q}|X_1) = \frac{1-q}{q} [n-x_1]_q^2 \frac{\theta_2^2}{(1 + \theta_2 q^{n-x_1-1})^2 (1 + \theta_2 q^{n-x_1-2})} \\ &\quad + [n-x_1]_q \frac{\theta_2}{(1 + \theta_2 q^{n-x_1-1})(1 + \theta_2 q^{n-x_1-2})}, \end{aligned} \tag{12}$$

respectively.

Note 1. The conditional q -mean, $\mu_{[X_2]_{1/q}|X_1}$, can be interpreted as a q -regression curve.

Let us now consider the deformed random variables $[X_1]_{1/q}$ and $[X_2]_{1/q}$ as well as the q -standardized random variables $Z = \frac{[X_1]_{1/q} - \mu_{[X_1]_{1/q}}}{\sigma_{[X_1]_{1/q}}}$ and $W = \frac{[X_2]_{1/q} - \mu_{[X_2]_{1/q}|X_1}}{\sigma_{[X_2]_{1/q}|X_1}}$ with $\mu_{[X_1]_{1/q}}, \sigma_{[X_1]_{1/q}}$ and $\mu_{[X_2]_{1/q}|X_1}, \sigma_{[X_2]_{1/q}|X_1}$ given by (11) and (12), respectively. Then, we apply pointwise convergence techniques to the joint probability function (10), by using suitably the q -Stirling type (5), and we obtain the following theorem concerning the asymptotic behaviour of the q -trinomial distribution.

Theorem 2. Let $\theta_1 = \theta_{1,n} = q^{-\alpha_1 n}$ and $\theta_2 = \theta_{2,n} = q^{-\alpha_2 n}$ with $0 < \alpha_1, \alpha_2 < 1$ constants and $0 < q < 1$. Then, for $n \rightarrow \infty$, the q -trinomial distribution of the first kind is approximated by a deformed standardized bivariate continuous Stieltjes-Wigert distribution as follows:

$$\begin{aligned} f_{X_1, X_2}^B(x_1, x_2) &\cong \frac{q^{-7/4} \log q^{-1}}{2\pi(q^{-1} - 1) \sigma_{[X_1]_{1/q}} \sigma_{[X_2]_{1/q}|X_1}} q^{-(x_1+x_2)} \left(q^{-3/2} (1-q)^{1/2} \frac{[x_1]_{1/q} - \mu_{[X_1]_{1/q}}}{\sigma_{[X_1]_{1/q}}} + q^{-1} \right)^{-1/2} \\ &\cdot \left(q^{-3/2} (1-q)^{1/2} \frac{[x_2]_q - \mu_{[X_2]_{1/q}|X_1}}{\sigma_{[X_2]_{1/q}|X_1}} + q^{-1} \right)^{-1/2} \\ &\cdot \exp \left(\frac{1}{2 \log q} \left(\log^2 \left(q^{-3/2} (1-q)^{1/2} \frac{[x_1]_{1/q} - \mu_{[X_1]_{1/q}}}{\sigma_{[X_1]_{1/q}}} + q^{-1} \right) \right) \right) \\ &\cdot \exp \left(\frac{1}{2 \log q} \log^2 \left(q^{-3/2} (1-q)^{1/2} \frac{[x_2]_q - \mu_{[X_2]_{1/q}|X_1}}{\sigma_{[X_2]_{1/q}|X_1}} + q^{-1} \right) \right), \quad x_1, x_2 \geq 0, \end{aligned} \tag{13}$$

where $\mu_{[X_1]_{1/q}}$ and $\sigma_{[X_1]_{1/q}}^2$, given in (11), are the mean value and the variance of the random variable $[X_1]_{1/q}$ while $\mu_{[X_2]_{1/q}|X_1}$ and $\sigma_{[X_2]_{1/q}|X_1}^2$, given in (12), are the conditional mean value and the conditional variance of the random variable $[X_2]_{1/q}$ given $X_1 = x_1$.

Next we expand our study on the asymptotic behaviour of the q -multinomial distribution with joint p.f. (8).

The marginal probability function of the random variable X_1 , is distributed according to the q -binomial of the 1st kind with probability function

$$f_{X_1}^B(x_1) = \binom{n}{x_1}_q \frac{\theta_1^{x_1} q^{\binom{x_1}{2}}}{\prod_{i=1}^n (1 + \theta_1 q^{i-1})}, \quad x_1 = 0, 1, 2, \dots, n.$$

The mean and the variance of the deformed variable $[X_1]_{1/q}$ are given by

$$\begin{aligned} \mu_{[X_1]_{1/q}} &= E([X_1]_{1/q}) = [n]_q \frac{\theta_1}{1 + \theta_1 q^{n-1}} \\ &\text{and} \\ (\sigma_{[X_1]_{1/q}})^2 &= V([X_1]_{1/q}) = \frac{1-q}{q} [n]_q^2 \frac{\theta_1^2}{(1 + \theta_1 q^{n-1})^2 (1 + \theta_1 q^{n-2})} + [n]_q \frac{\theta_1}{(1 + \theta_1 q^{n-1})(1 + \theta_1 q^{n-2})} \end{aligned} \quad (14)$$

respectively.

The conditional random variables $X_2|X_1, X_3|(X_1, X_2), \dots, X_k|(X_1, \dots, X_{k-1})$ are distributed according to univariate q -binomial distributions of the 1st kind with probability functions

$$\begin{aligned} f_{X_k|(X_1, \dots, X_{k-1})}^B(x_k | x_1, x_2, \dots, x_{k-1}) &= \binom{n - \sum_{j=1}^{k-1} x_j}{x_k}_q \frac{\theta_k^{x_k} q^{\binom{x_k}{2}}}{\prod_{i=1}^{n - \sum_{j=1}^{k-1} x_j} (1 + \theta_k q^{i-1})}, \\ &x_k = 0, 1, \dots, \sum_{j=1}^{k-1} x_j, \quad k \geq 2. \end{aligned}$$

The conditional mean and conditional variance of the deformed variables $[X_j]_{1/q}$ given $X_1 = x_1, \dots, X_{j-1} = x_{j-1}$, $j = 2, \dots, k$, $k \geq 2$, are given respectively by

$$\begin{aligned} \mu_{[X_j]_{1/q}(X_1, \dots, X_{j-1})} &= E([X_j]_{1/q} | (X_1, \dots, X_{j-1})) = [n - s_{j-1}]_q \frac{\theta_j}{1 + \theta_j q^{n-s_{j-1}-1}} \\ &\text{and} \\ \sigma_{[X_j]_{1/q}(X_1, \dots, X_{j-1})}^2 &= V([X_j]_{1/q} | (X_1, \dots, X_{j-1})) \\ &= \frac{1-q}{q} [n - s_{j-1}]_q^2 \frac{\theta_j^2}{(1 + \theta_j q^{n-s_{j-1}-1})^2 (1 + \theta_j q^{n-s_{j-1}-2})} \\ &\quad + [n - s_{j-1}]_q \frac{\theta_j}{(1 + \theta_j q^{n-s_{j-1}-1})(1 + \theta_j q^{n-s_{j-1}-2})}, \end{aligned} \quad (15)$$

where $s_{j-1} = \sum_{i=1}^{j-1} x_i$, $j = 2, \dots, k$, $k \geq 2$.

Note 3. It should be noted that the conditional q -means, $\mu_{[X_j]_{1/q}(X_1, \dots, X_{j-1})}$, $3 \leq j \leq k$, $k \geq 3$, can be interpreted as q -regression hyperplanes.

Let us now consider the deformed random variables

$$[X_j]_{1/q}, \quad j = 1, \dots, k, \quad k \geq 1$$

and the q -standardized random variables

$$Z_1 = \frac{[X_1]_{1/q} - \mu_{[X_1]_{1/q}}}{\sigma_{[X_1]_{1/q}}}, Z_j = \frac{[X_j]_{1/q} - \mu_{[X_j]_{1/q}|(X_1, \dots, X_{j-1})}}{\sigma_{[X_j]_{1/q}|(X_1, \dots, X_{j-1})}}, j = 2, \dots, k, k \geq 3,$$

with $\mu_{[X_1]_{1/q}}, \sigma_{[X_1]_{1/q}}$ and $\mu_{[X_j]_{1/q}|(X_1, \dots, X_{j-1})}, \sigma_{[X_j]_{1/q}|(X_1, \dots, X_{j-1})}$ given by (14) and (15), respectively. Then, we apply pointwise convergence techniques to the joint probability function (8), by using suitably the q -Stirling type (5), and we obtain the following theorem concerning the asymptotic behaviour of the q -multinomial distribution.

Theorem 4. *Let $\theta_j = \theta_{j,n} = q^{-\alpha_j n}$ with $0 < a_j < 1, j = 1, 2, \dots, k$ constants and $0 < q < 1$. Then, for $n \rightarrow \infty$, the q -multinomial distribution is approximated by a deformed multivariate standardized continuous Stieltjes-Wigert distribution distribution as follows:*

$$\begin{aligned} f_{\mathcal{X}}^B(x_1, x_2, \dots, x_k) &\cong \left(\frac{q^{-7/8} (\log q^{-1})^{1/2}}{(2\pi)^{1/2} (q^{-1} - 1)^{1/2}} \right)^k \frac{q^{-\sum_{j=1}^k x_j}}{\sigma_{[X_1]_{1/q}} \prod_{j=2}^k \sigma_{[X_j]_{1/q}|(X_1, \dots, X_{j-1})}} \\ &\cdot \left(q^{-3/2} (1-q)^{1/2} \frac{[x_1]_{1/q} - \mu_{[X_1]_{1/q}}}{\sigma_{[X_1]_{1/q}}} + q^{-1} \right)^{-1/2} \\ &\cdot \prod_{j=2}^k \left(q^{-3/2} (1-q)^{1/2} \frac{[x_j]_{1/q} - \mu_{[X_j]_{1/q}|(X_1, \dots, X_{j-1})}}{\sigma_{[X_j]_{1/q}|(X_1, \dots, X_{j-1})}} + q^{-1} \right)^{-1/2} \\ &\cdot \exp \left(\frac{1}{2 \log q} \left(\log^2 \left(\frac{(1-q)^{1/2} [x_1]_{1/q} - \mu_{[X_1]_{1/q}}}{q^{3/2} \sigma_{[X_1]_{1/q}}} + q^{-1} \right) \right) \right) \\ &\cdot \exp \left(\frac{1}{2 \log q} \sum_{j=2}^k \log^2 \left(\frac{(1-q)^{1/2} [x_j]_{1/q} - \mu_{[X_j]_{1/q}|(X_1, \dots, X_{j-1})}}{q^{3/2} \sigma_{[X_j]_{1/q}|(X_1, \dots, X_{j-1})}} + q^{-1} \right) \right), \\ &x_j \geq 0, j = 1, 2, \dots, k, k \geq 2, \end{aligned} \tag{16}$$

where $\mu_{[X_1]_{1/q}}$ and $\sigma_{[X_1]_{1/q}}^2$, given in (14), are the mean value and the variance of the random variable $[X_1]_{1/q}$, while $\mu_{[X_j]_{1/q}|(X_1, \dots, X_{j-1})}$ and $\sigma_{[X_j]_{1/q}|(X_1, \dots, X_{j-1})}^2$, given in (15), are the conditional mean values and the conditional variances of the random variables $[X_j]_{1/q}$ given $X_1 = x_1, \dots, X_{j-1} = x_{j-1}, j = 2, \dots, k, k \geq 2$.

Remark 5. *Let $\mathcal{X} = (X_1, X_2, \dots, X_k)$ be a random vector that follows the multiple Heine distribution, defined in (9). Then the joint p.f. the multiple Heine distribution is given by*

$$f_{\mathcal{X}}^H(x_1, x_2, \dots, x_k) = \prod_{j=1}^k \frac{q^{\binom{x_j}{2}} \lambda_j^{x_j}}{[x_j]_q!} \prod_{i=1}^{\infty} (1 + \lambda_j (1-q) q^{i-1})^{-1},$$

where $x_j = 0, 1, 2, \dots, \lambda_j > 0, 0 < q < 1, \lambda_j = \theta_j / (1-q), j = 1, 2, \dots, k, k \geq 2$. Since the random variables $X_j, j = 1, 2, \dots, k, k \geq 2$, are independent, we easily derive that, for $\lambda_j \rightarrow \infty, j = 1, 2, \dots, k$, the multiple Heine distribution is approximated by a deformed multivariate standardized

continuous Stieltjes-Wigert distribution as follows:

$$\begin{aligned}
 f_{\mathcal{D}}^H(x_1, x_2, \dots, x_k) &\cong \left(\frac{q^{-7/8} (\log q^{-1})^{1/2}}{(2\pi)^{1/2} (q^{-1} - 1)^{1/2}} \right)^k \frac{q^{-\sum_{j=1}^k x_j}}{\prod_{j=1}^k \sigma_{[X_j]_{1/q}}} \\
 &\cdot \prod_{j=1}^k \left(q^{-3/2} (1-q)^{1/2} \frac{[x_j]_{1/q} - \mu_{[X_j]_{1/q}}}{\sigma_{[X_j]_{1/q}}} + q^{-1} \right)^{-1/2} \\
 &\cdot \exp \left(\frac{1}{2 \log q} \left(\sum_{j=1}^k \log^2 \frac{(1-q)^{1/2} [x_j]_{1/q} - \mu_{[X_j]_{1/q}}}{q^{3/2} \sigma_{[X_j]_{1/q}}} \right) \right), \\
 &x_j \geq 0, j = 1, \dots, k, k \geq 2,
 \end{aligned} \tag{17}$$

where $\mu_{[X_j]_{1/q}} = \lambda_j$ and $\sigma_{[X_j]_{1/q}}^2 = \lambda_j q^{-1} (1-q) + \lambda_j$, $j = 1, 2, \dots, k$, are respectively the mean values and the variances of the random variables $[X_j]_{1/q}$, $j = 1, 2, \dots, k$.

References

- [1] Charalambos A. Charalambides (2016): *Discrete q -Distributions*. John Wiley & Sons, Hoboken, NJ, doi:10.1002/9781119119128.
- [2] Charalambos A. Charalambides (2021): *q -Multinomial and Negative q -Multinomial Distributions*. *Communications in Statistics - Theory and Methods* 50, pp. 5673–5898, doi:10.1080/03610926.2020.1737711.
- [3] Charalambos A. Charalambides (2022): *Multivariate q -Pólya and Inverse q -Pólya Distributions*. *Communications in Statistics - Theory and Methods* 51, pp. 4854–4876, doi:10.1080/03610926.2020.1825740.
- [4] Andreas Kyriakoussis & Malvina Vamvakari (2013): *On a q -analogue of the Stirling formula and a continuous limiting behaviour of the q -Binomial distribution-Numerical calculations*. *Methodology and Computing in Applied Probability* 15, pp. 187–213, doi:10.1007/s11009-011-9231-1.
- [5] Andreas Kyriakoussis & Malvina Vamvakari (2014): *Continuous Stieltjes–Wigert Limiting Behaviour of a Family of Confluent q -Chu-Vandermonde Distributions*. *Axioms* 3, pp. 140–152, doi:10.3390/axioms3020140.
- [6] Andreas Kyriakoussis & Malvina Vamvakari (2017): *Heine Process as a q -Analog of the Poisson Process: Waiting and Interarrival Times*. *Communications in Statistics - Theory and Methods* 46, pp. 4088–4102, doi:10.1080/03610926.2015.1078476.
- [7] Andreas Kyriakoussis & Malvina Vamvakari (2019): *Asymptotic Behaviour of Certain q -Poisson, q -Binomial and Negative q -Binomial Distributions*. *Lattice Path Combinatorics and Applications - Edited by George E. Andrews, Christian Krattenthaler and Alan Krinik, Developments in Mathematics* 58, pp. 283–306, doi:10.1007/978-3-030-11102-1_13.
- [8] Malvina Vamvakari (2020): *On Multivariate Discrete q -Distributions: A Multivariate q -Cauchy's Formula*. *Communications in Statistics - Theory and Methods* 49, pp. 6080–6095, doi:10.1080/03610926.2019.1626427.
- [9] Malvina Vamvakari (2024): *Asymptotic Behaviour of Univariate and Multivariate Absorption Distributions*. *Randomness and Combinatorics - Edited by Luca Ferrari and Paolo Massazza. RAIRO - Theoretical Informatics and Applications* 58, doi:10.1051/ita/2024006.