

Genomics, Pattern Avoidance, and Statistical Mechanics

Edited by

Michael Albert¹, David Bevan², Miklós Bóna³, and István Miklós⁴

¹ University of Otago, NZ, malbert@cs.otago.ac.nz

² University of Strathclyde – Glasgow, GB, david.bevan@strath.ac.uk

³ University of Florida – Gainesville, US, bona@ufl.edu

⁴ Alfréd Rényi Institute of Mathematics – Budapest, HU,
miklos.istvan.74@gmail.com

Abstract

We summarize key features of the workshop, such as the three main research areas in which the participants are active, the number and types of talks, and the geographic diversity of the attendees. We also provide a sampling of the collaborations started at the workshop, and explain why we believe that the workshop was successful, and why we believe it should take place again in the future.

Seminar November 4–9, 2018 – <http://www.dagstuhl.de/18451>

2012 ACM Subject Classification Mathematics of computing → Approximation algorithms, Applied computing → Bioinformatics, Theory of computation → Data structures design and analysis, Applied computing → Systems biology

Keywords and phrases Genome rearrangements, Matrix, Pattern, Permutation, Statistical Mechanics

Digital Object Identifier 10.4230/DagRep.8.11.1

1 Executive Summary

Miklós Bóna (University of Florida – Gainesville, US)

License © Creative Commons BY 3.0 Unported license
© Miklós Bóna

This report documents the program and the outcomes of Dagstuhl Seminar 18451 “Genomics, Pattern Avoidance, and Statistical Mechanics”.

The workshop took place from November 4, 2018 to November 9, 2018. It had 40 participants, who were researchers in theoretical computer science, combinatorics, statistical mechanics and molecular biology. It was a geographically diverse group, with participants coming from the US, Canada, Brazil, Germany, Iceland, the United Kingdom, Sweden, France, Switzerland, Hungary, Australia, and New Zealand. The workshop featured 21 talks, three of which were hourlong talks, and an open problem session.

Several collaborative projects have been started. For example, Jay Pantone, Michael Albert, Robert Brignall, Seth Pettie, and Vince Vatter started exploring the topic of 1324-avoiding permutations with a bounded number of descents, disproving a 2005 conjecture of Elder, Reznitzer, and Zabrocki related to Davenport-Schinzel sequences. Had the conjecture been affirmed, it would have implied that the generating function for 1324-avoiding permutations is non-D-finite.

At the open problem session, Yann Ponty raised the following question: what is the number of independent sets in restricted families of trees, like caterpillars or complete binary



Except where otherwise noted, content of this report is licensed under a Creative Commons BY 3.0 Unported license

Genomics, Pattern Avoidance, and Statistical Mechanics, *Dagstuhl Reports*, Vol. 8, Issue 11, pp. 1–20

Editors: Michael Albert, David Bevan, Miklós Bóna, and István Miklós



Dagstuhl Reports

Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

plane trees? The main motivation for this question relates to a deep connection between such independent sets and RNA designs. This question led to a new collaborative effort by Mathilde Bouvel, Robert Brignall, Yann Ponty and Andrew Elvey Price.

Sergi Elizalde and Miklós Bóna have started working on Dyck paths that have a unique maximal peak. That collaboration since extended to the area of probabilistic methods, involving a researcher working in that field, Douglas Rizzolo.

Numerous participants expressed their pleasure with the workshop and its sequence of talks. The prevailing view was that while the participants came from three different fields, they were all open to the other two fields, and therefore, they all learned about results that they would not have learned otherwise. Therefore, we have all the reasons to believe that the workshop was a success, and we would like to repeat it some time in the future.

2 Table of Contents

Executive Summary

<i>Miklós Bóna</i>	1
------------------------------	---

Overview of Talks

The curious behaviour of the total displacement	
<i>David Bevan</i>	5
The solution space of canonical DCJ genome sorting	
<i>Marília Braga</i>	5
Permutations and Permutation Graphs	
<i>Robert Brignall</i>	6
On the Median and Small Parsimony problems in some genome rearrangement Models	
<i>Cedric Chauve</i>	6
Brain Modularity Mediates the Relation between Task Complexity and Performance	
<i>Michael W. Deem</i>	6
A Markov-chain model of chromosomal instability	
<i>Sergi Elizalde</i>	7
Sampling bipartite degree sequence realizations – the Markov chain approach	
<i>Péter L. Erdős</i>	8
Sorting Permutations in the GR+IR model	
<i>Guillaume Fertin</i>	8
The combinatorics of RNA branching	
<i>Christine E. Heitsch</i>	9
Modelling dense polymers by self-avoiding walks	
<i>E. J. Janse van Rensburg</i>	9
Pattern avoidance: algorithmic connections	
<i>László Kozma</i>	10
Combinatorics of the ASEP on a ring and Macdonald polynomials	
<i>Olya Mandelshtam</i>	10
Computational complexity of counting and sampling genome rearrangement scenarios	
<i>István Miklós</i>	10
On some generalizations to trees of problems for permutations	
<i>Alois Panholzer</i>	11
Sorting Permutations with C-Machines	
<i>Jay Pantone, Michael Albert, Cheyne Homberger, Nathaniel Shar, and Vincent Vatter</i>	11
Amortized Analysis of Data Structures via Forbidden 0-1 Matrices	
<i>Seth Pettie</i>	12
Lattice Path Counting: where Enumerative Combinatorics and Statistical Mechanics meet	
<i>Thomas Prellberg</i>	12

Permutations in labellings of trees	
<i>Fiona Skerman</i>	12
Enumerating $1 \times n$ generalised permutation grid classes	
<i>Jakub Sliacan and Robert Brignall</i>	13
Complexity of the Single Cut-or-Join model and Partition Functions	
<i>Heather Smith and István Miklós</i>	13
Two topics on permutation patterns	
<i>Vincent Vatter, Michael Engen, and Jay Pantone</i>	14
On Square Permutations	
<i>Stéphane Vialette</i>	14
Working groups	
Enumerative aspects of multiple RNA design	
<i>Mathilde Bouvel, Robert Brignall, Andrew Elvey Price, and Yann Ponty</i>	15
Explicit enumeration results on constrained dependency graphs	
Union of paths and cycles	17
Caterpillars	17
Complete binary trees	18
Going further	19
Participants	20

3 Overview of Talks

3.1 The curious behaviour of the total displacement

David Bevan (*University of Strathclyde – Glasgow, GB*)

License © Creative Commons BY 3.0 Unported license
© David Bevan

The *total displacement* of a permutation $\sigma = \sigma_1 \dots \sigma_n$ is $\text{td}(\sigma) = \sum_i |\sigma_i - i|$. The ratio of the total displacement to the number of inversions, $R(\sigma) = \text{td}(\sigma)/\text{inv}(\sigma)$, is known to lie in the half-open interval $(1, 2]$ (unless σ is an increasing permutation, with no inversions).

Let $\pi_{n,m}$ denote a permutation chosen uniformly at random from the set of all n -permutations with exactly m inversions. In this talk, we consider the behaviour of the expected asymptotic displacement ratio $R[m] = \lim_{n \rightarrow \infty} \mathbb{E}[R(\pi_{n,m})]$, as $m = m(n)$ increases from 1 to $\binom{n}{2}$.

As long as $m = o(n)$, $R[m]$ takes the constant value of 2. Then, when $m \sim \alpha n$, $R[m]$ decreases as α increases, from 2 down to $2 \log 2 \approx 1.3863$. However, once m becomes superlinear in n , $R[m]$ stalls again, at $2 \log 2$, until $m = \Theta(n^2)$. Finally, when $m \sim \rho \binom{n}{2}$, $R[m]$ decreases from $2 \log 2$, taking the value $\frac{4}{3}$ when $\rho = \frac{1}{2}$ and finally reaching 1 when $\rho = 1$.

We investigate how this curious behaviour can be explained in terms of the different effects that local and global constraints have on $\pi_{n,m}$.

3.2 The solution space of canonical DCJ genome sorting

Marília Braga (*Universität Bielefeld, DE*)

License © Creative Commons BY 3.0 Unported license
© Marília Braga

Main reference Marília D. V. Braga, Jens Stoye: “The Solution Space of Sorting by DCJ”, *Journal of Computational Biology*, Vol. 17(9), pp. 1145–1165, 2010.

URL <https://doi.org/10.1089/cmb.2010.0109>

In genome rearrangements, the double cut and join (DCJ) operation, introduced by Yancopoulos et al. in 2005, allows one to represent most rearrangement events that could happen in multichromosomal genomes, such as inversions, translocations, fusions, and fissions. No restriction on the genome structure considering linear and circular chromosomes is imposed. An advantage of this general model is that it leads to considerable algorithmic simplifications compared to other genome rearrangement models. Several studies about the DCJ operation have been published, and in particular, an algorithm was proposed to find an optimal DCJ sequence for sorting one genome into another one. Here we analyze the solution space of this problem and give an easy-to-compute formula that corresponds to the exact number of optimal DCJ sorting sequences for a particular subset of instances of the problem. We also give an algorithm to count the number of optimal sorting sequences for any instance of the problem. An additional interesting result is the demonstration that, by properly replacing any pair of consecutive operations in any optimal sorting sequence, one always obtains another optimal sorting sequence. As a consequence, any optimal sorting sequence can be obtained from one other by applying such replacements successively.

This work was published in 2010 by Braga and Stoye in the *Journal of Computational Biology* (DOI: 10.1089/cmb.2010.0109).

3.3 Permutations and Permutation Graphs

Robert Brignall (The Open University – Milton Keynes, GB)

License © Creative Commons BY 3.0 Unported license
© Robert Brignall

Main reference Nicholas Korpelainen, Vadim V. Lozin, Igor Razgon: “Boundary Properties of Well-Quasi-Ordered Sets of Graphs”, *Order*, Vol. 30(3), pp. 723–735, 2013.

URL <https://doi.org/10.1007/s11083-012-9272-2>

The inversion graph of a permutation provides a convenient tool for translating results and theory between the study of permutations and the study of graphs. In particular, the pattern containment ordering corresponds, in a fairly direct way, to the induced subgraph ordering.

In this talk, I will present two distinct-but-conjecturally-connected topics where permutations have helped: (1) in the study of the graph parameter clique-width, permutation classes provide us with a rich source of hereditary graph properties which are minimal with unbounded clique-width; (2) in the study of well-quasi-ordering for graphs, we exhibit a counterexample to a conjecture made by Korpelainen, Lozin and Razgon, which was found using intuition obtained from the study of well-quasi-ordering in permutations.

3.4 On the Median and Small Parsimony problems in some genome rearrangement Models

Cedric Chauve (Simon Fraser University – Burnaby, CA)

License © Creative Commons BY 3.0 Unported license
© Cedric Chauve

The main goal of genome rearrangement problems is to compute evolutionary scenarios that can explain the order of genes observed in extant genomes. This naturally leads to questions about the order of genes in ancestral genomes, often of extinct species. If a species phylogeny is given, this problem is known as the Small Parsimony Problem, and in its simplest form, where a single ancestral genome is considered, the Median Problem. In this talk, I will first review several algorithmic results on the Median and Small Parsimony Problems, from initial intractability results to surprising tractability results, and then present some more recent results on the same problems in the context where duplicated genes are considered.

3.5 Brain Modularity Mediates the Relation between Task Complexity and Performance

Michael W. Deem (Rice University – Houston, US)

License © Creative Commons BY 3.0 Unported license
© Michael W. Deem

Recent work in cognitive neuroscience has focused on analyzing the brain as a network, rather than as a collection of independent regions. Prior studies taking this approach have found that individual differences in the degree of modularity of the brain network relate to performance on cognitive tasks. However, inconsistent results concerning the direction of this relationship have been obtained, with some tasks showing better performance as modularity increases and other tasks showing worse performance. Our recent theoretical model suggests

that these inconsistencies may be explained on the grounds that high-modularity networks favor performance on simple tasks whereas low-modularity networks favor performance on more complex tasks. I will review experiments being carried out by collaborators showing a negative correlation between individuals' modularity and their performance on a composite measure combining scores from the complex tasks and a positive correlation with performance on a composite measure combining scores from the simple tasks. I will further present theory showing that a dynamic measure of brain connectivity termed flexibility is predicted to correlate in the opposite way with performance. I will review experiments confirming these predictions and also showing that flexibility plays a greater role in predicting performance on complex tasks requiring cognitive control and executive functioning. The theory and results presented here provide a framework for linking measures of whole-brain organization from network neuroscience to cognitive processing.

References

- 1 A.I. Ramos-Nuez, S. Fischer-Baum, R. Martin, Q.-H. Yue, F.-D. Ye, and M.W. Deem, "Static and Dynamic Measures of Human Brain Connectivity Predict Complementary Aspects of Human Cognitive Performance," *Front. Hum. Neurosci.* (2017) doi: 10.3389/fn-hum.2017.00420.
- 2 Q.-H. Yue, R. Martin, S. Fischer-Baum, A.I. Ramos-Nuez, F.-D. Ye, and M.W. Deem, "Brain Modularity Mediates the Relation of Cognitive Performance to Task Complexity," *J. Cog. Neurosci.* **29** (2017) 1532–1546.
- 3 J.-M. Park, M. Chen, D. Wang, and M.W. Deem, "Modularity Enhances the Rate of Evolution in a Rugged Fitness Landscape," *Phys. Biol.* **12** (2015) 025001.
- 4 J.-M. Park, L.R. Niestemski, and M.W. Deem, "Quasispecies Theory for Evolution of Modularity," *Phys. Rev. E* **91** (2015) 012714.

3.6 A Markov-chain model of chromosomal instability

Sergi Elizalde (Dartmouth College – Hanover, US)

License  Creative Commons BY 3.0 Unported license
© Sergi Elizalde

Joint work of Sergi Elizalde, Sam Bakhoun, Ashley Laughney

Genomic instability allows cancer cells to rapidly vary the number of copies of each chromosome (karyotype) through chromosome missegregation events during mitosis, enabling genetic heterogeneity that leads to tumor metastasis and drug resistance. We construct a Markov chain that describes the evolution of the karyotypes of cancer cells. The Markov chain is based on a stochastic model of chromosome missegregation which incorporates the observed fact that individual chromosomes contain proliferative and anti-proliferative genes, leading to cells with varying fitness levels and allowing for Darwinian selection to occur. We analyze the Markov chain mathematically, and we use it to predict the long-term distribution of karyotypes of cancer cells. We then adapt it to study the behavior of tumors under targeted therapy and to model drug resistance.

3.7 Sampling bipartite degree sequence realizations – the Markov chain approach

Péter L. Erdős (Alfréd Rényi Institute of Mathematics – Budapest, HU)

License © Creative Commons BY 3.0 Unported license

© Péter L. Erdős

Joint work of Péter L. Erdős, Miklós, István

Main reference Péter L. Erdős, Tamás Róbert Mezei, István Miklós, Dániel Soltész: “Efficiently sampling the realizations of bounded, irregular degree sequences of bipartite and directed graphs”. PLoS ONE 13(8): e0201995, 2018.

URL <https://doi.org/10.1371/journal.pone.0201995>

How to analyze real life networks? There are myriads of them and usually experiments cannot be performed directly on them. Instead, scientists define models, fix parameters and imagine the dynamics of evolution.

Then, they build synthetic networks on this basis (one, several, all) and they want to sample them. However, there are far too many such networks. Therefore, typically, some probabilistic method is used for sampling.

We will survey one such approach, the Markov Chain Monte Carlo method, to sample realizations of given degree sequences. Some new results will be discussed. The majority of the talk is published in [1].

References

- 1 P.L. Erdős, T.R. Mezei, I. Miklós, D. Soltész: Efficiently sampling the realizations of bounded, irregular degree sequences of bipartite and directed graphs, *PLOS One* 2018 (2018), # e0201995, 1–19.

3.8 Sorting Permutations in the GR+IR model

Guillaume Fertin (University of Nantes, FR)

License © Creative Commons BY 3.0 Unported license

© Guillaume Fertin

Joint work of Guillaume Fertin, Géraldine Jean, Eric Tannier

Main reference Guillaume Fertin, Géraldine Jean, Eric Tannier: “Genome Rearrangements on Both Gene Order and Intergenic Regions”, in Proc. of the Algorithms in Bioinformatics – 16th International Workshop, WABI 2016, Aarhus, Denmark, August 22-24, 2016. Proceedings, Lecture Notes in Computer Science, Vol. 9838, pp. 162–173, Springer, 2016.

URL https://doi.org/10.1007/978-3-319-43681-4_13

A genome can be, in its simplest form, modeled as a permutation π of length n , where each π_i , $1 \leq i \leq n$, represents a gene. A genome rearrangement (or GR) is a large scale evolutionary event that modifies a genome. For instance, a *reversal* consists in taking a contiguous subsequence from π , reversing it, and reincorporating it at the same location:

$$\pi = 3 \ 5 \ 1 \ 2 \ 7 \ 4 \ 6 \rightarrow \pi' = 3 \ 2 \ 1 \ 5 \ 7 \ 4 \ 6$$

Sorting by rearrangements then consists, given a permutation π and a set \mathcal{S} of allowed GRs, in determining a shortest sequence of GRs from \mathcal{S} that transforms π in the identity permutation Id_n . The algorithmic study of sorting by rearrangements has led to an abundant literature in the last 20 years or so, and given rise to many fascinating results.

It is however possible to enrich the model as follows: since consecutive genes in a genome are actually separated by an *intergenic region* (or IR) – i.e. by a certain number of DNA bases –, we can model a genome by a pair consisting of (a) a permutation π , together with (b)

an ordered multiset $S = \{r_1, r_2 \cdots r_{n-1}\}$ of positive integers, where each r_j , $1 \leq j \leq n-1$, represents the size of the IR between genes π_j and π_{j+1} . A GR acts between genes, thus inside IRs – in the above example, the shown reversal cuts π between genes 3 and 5, and between genes 2 and 7. Hence, any GR can simultaneously modify the sizes of the affected IRs *and* the order of the genes.

In this setting, which we call the GR+IR model, the sorting problem becomes the following: given a pair (π, S) representing a genome together with its IRs, find a shortest sequence of GRs that leads to the pair (Id_n, S') , where S' encodes the IRs of the target permutation.

In this talk, I will introduce the GR+IR model in more details, and give some algorithmic results related to the corresponding sorting problem, with a specific focus on the following two types of GR: DCJ (Double Cut and Join) and reversals.

3.9 The combinatorics of RNA branching


Christine E. Heitsch (Georgia Institute of Technology – Atlanta, US)

License  Creative Commons BY 3.0 Unported license
© Christine E. Heitsch

Understanding the folding of RNA sequences into three-dimensional structures is one of the fundamental challenges in molecular biology. For example, the branching of an RNA secondary structure is an important molecular characteristic yet difficult to predict correctly. However, results from enumerative, probabilistic, and geometric combinatorics can characterize different types of branching landscapes, yielding insights into RNA structure formation.

3.10 Modelling dense polymers by self-avoiding walks


E. J. Janse van Rensburg (York University — Toronto, CA)

License  Creative Commons BY 3.0 Unported license
© E. J. Janse van Rensburg

A dense polymer (for example in a polymer melt, or polymers in confined spaces in living cells) can be modelled by a lattice self-avoiding walk in a confined space. For example, in the square lattice a self-avoiding walk can be confined to a square. If the walk is very short compared to size of the square, then it is in an expanded phase, but when it is long, then it will start to fill the area of the square and is a compressed walk. In this talk I give a summary about modelling the free energy of a compressed walk by using Flory-Huggins theory (a mean field phenomenological theory of the free energy of dense polymer systems). We estimate numerically the Flory Interaction Parameter for square lattice self-avoiding walks, and also give an extrapolated estimate of the connective constant of Hamiltonian walks of a square. I will also produce tentative results on compressed and knotted lattice polygons in 3 dimensions.

3.11 Pattern avoidance: algorithmic connections

László Kozma (*FU Berlin, DE*)


License  Creative Commons BY 3.0 Unported license
© László Kozma

Permutation-patterns and pattern-avoiding permutations have long been the subject of algorithmic study. Classical problems include pattern matching, enumeration of pattern-occurrences and of pattern-avoiding sequences, and others. More recently, pattern-avoidance was also studied in the property-testing framework.

Perhaps less-studied – apart from special cases – is the question whether the avoidance of patterns makes classical algorithmic problems easier. (Analogous questions in graphs are well-studied.) In my talk I discuss sorting and searching in basic data structures, when the input is pattern-avoiding.

3.12 Combinatorics of the ASEP on a ring and Macdonald polynomials

Olya Mandelshtam (*Brown University – Providence, US*)

License  Creative Commons BY 3.0 Unported license
© Olya Mandelshtam

Joint work of Olya Mandelshtam, Sylvie Corteel, Lauren Williams

Main reference Sylvie Corteel, Olya Mandelshtam, Lauren Williams: “Combinatorics of the two-species ASEP and Koornwinder moments”. *Advances in Mathematics*, Vol. 321, pp.160–204, 2017

URL <https://doi.org/10.1016/j.aim.2017.09.034>

The multispecies asymmetric simple exclusion process (ASEP) is a model of hopping particles of M different types hopping on a one-dimensional lattice of N sites. In this talk, we consider the ASEP on a ring with the following dynamics: particles at adjacent sites can swap places with either rate 1 or t depending on their relative types. Recently, James Martin gave a combinatorial formula for the stationary probabilities of the ASEP with generalized *multiline queues*. We will begin by describing the combinatorial methods we use to study the ASEP on a ring.

Furthermore, it turns out that by introducing additional statistics on the multiline queues, we get a new formula for both symmetric Macdonald polynomials P_λ and nonsymmetric Macdonald polynomials E_λ , where λ is a partition. For the second part of the talk, we will discuss the recent results and remarkable connection with Macdonald polynomials.

3.13 Computational complexity of counting and sampling genome rearrangement scenarios

István Miklós (*Alfréd Rényi Institute of Mathematics – Budapest, HU*)

License  Creative Commons BY 3.0 Unported license
© István Miklós

Joint work of István Miklós, Heather C. Smith, Eric Tannier

Main reference István Miklós, Heather Smith: “Sampling and counting genome rearrangement scenarios”, *BMC Bioinformatics*, Vol. 16(Suppl 14), pp. S6, 2015.

URL <https://doi.org/10.1186/1471-2105-16-S14-S6>

Most of the counting problems fall into one of the following 3 categories:

1. In FP, that is, exactly solvable in polynomial time

2. In the intersection of $\#P$ -complete and FPRAS, that is, exact polynomial solution does not exist (assuming that $P \neq NP$) but efficient random approximation exists
3. In $\#P$ -complete and outside of FPRAS, that is, cannot be well approximated even in a stochastic manner (assuming that $RP \neq NP$).

The sampling counterparts usually follow the counting complexity and thus there is a perfect uniform sampler, an approximate uniform sampler or any sampler that runs in polynomial time is far from the uniform distribution.

There are several genome rearrangement models (sorting by reversals, SCJ, DCJ, etc.) and several tasks (counting the most parsimonious scenarios between two genomes, computing the number of median genomes, etc.), and for each combination of models and tasks, we are interested in the computational complexity of the so-emerging computational problem. The talk will focus on the recent results and open problems. Connections to enumerative combinatorics, statistical physics and network analysis will also be discussed.

3.14 On some generalizations to trees of problems for permutations

Alois Panholzer (TU Wien, AT)

License © Creative Commons BY 3.0 Unported license
© Alois Panholzer

Main reference Marie-Louise Lackner, Alois Panholzer: “Parking functions for mappings”, J. Comb. Theory, Ser. A, Vol. 142, pp. 1–28, 2016.

URL <https://doi.org/10.1016/j.jcta.2016.03.001>

Various enumeration problems and statistics for permutations have been generalized to other combinatorial structures. In this talk we focus on some of such generalizations to labelled tree structures. In particular, some old and new results for random sequential adsorption, records, and local label-patterns in trees are discussed.

3.15 Sorting Permutations with C-Machines

Jay Pantone (Marquette University, US), Michael Albert (University of Otago, NZ), Cheyne Homberger, Nathaniel Shar, and Vincent Vatter (University of Florida – Gainesville, US)

License © Creative Commons BY 3.0 Unported license
© Jay Pantone, Michael Albert, Cheyne Homberger, Nathaniel Shar, and Vincent Vatter

Joint work of Michael H. Albert, Cheyne Homberger, Jay Pantone, Nathaniel Shar, Vincent Vatter

Main reference Michael H. Albert, Cheyne Homberger, Jay Pantone, Nathaniel Shar, Vincent Vatter: “Generating permutations with restricted containers”, J. Comb. Theory, Ser. A, Vol. 157, pp. 205–232, 2018.

URL <https://doi.org/10.1016/j.jcta.2018.02.006>

A C-machine is a type of sorting device that naturally generalizes stacks and queues. A C-machine is a container that is allowed to hold permutations from the permutation class C into which entries can be pushed and out of which entries may be popped. With this notation, a traditional stack is the $Av(12)$ -machine. This structural description allows us to find many terms in the counting sequences of several permutation classes of interest, but despite these numerous initial terms we are unable to find the exact or asymptotic behavior of their generating functions. I’ll discuss what we do know, what we don’t know, and what experimental methods tell us we might one day know.

3.16 Amortized Analysis of Data Structures via Forbidden 0-1 Matrices

Seth Pettie (University of Michigan – Ann Arbor, US)

License © Creative Commons BY 3.0 Unported license
© Seth Pettie

Main reference Seth Pettie: “On Nonlinear Forbidden 0-1 Matrices: A Refutation of a Füredi-Hajnal Conjecture”, in Proc. of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2010, Austin, Texas, USA, January 17-19, 2010, pp. 875–885, SIAM, 2010.

URL <https://doi.org/10.1137/1.9781611973075.71>

The amortized performance of a data structure is usually proved by designing and analyzing a “potential function”, which is an accounting mechanism for letting faster-than-average operations pay for slower-than-average operations. In this talk I will survey an alternative method for analyzing amortized data structures that models executions by 0-1 matrices and bounds their weight using theorems on the density of such matrices avoiding 0-1 patterns.

3.17 Lattice Path Counting: where Enumerative Combinatorics and Statistical Mechanics meet

Thomas Prellberg (Queen Mary University of London, GB)

License © Creative Commons BY 3.0 Unported license
© Thomas Prellberg

A topic common to the two disciplines in the title of this talk is the wish to count truly large ensembles of structures. This talk will examine different ways of how the problem of lattice path counting is approached using methods from both of these areas. While enumerative combinatorics strives to ideally provide exact counting numbers, statistical mechanics rather deals with the thermodynamic limit of large system sizes, where concepts like entropy and energy are related to asymptotic growth. I will endeavour to close the gap between exact counting formulas from enumerative combinatorics and the approximate counting underlying the statistical mechanical approach, and to clearly define jargon particular to either discipline.

3.18 Permutations in labellings of trees

Fiona Skerman (Uppsala University, SE)

License © Creative Commons BY 3.0 Unported license
© Fiona Skerman

Joint work of Fiona Skerman, Michael Albert, Cecilia Holmgren, Tony Johansson

We investigate the number of permutations that occur in random node labellings of trees. This is a generalisation of the number of sub-permutations occurring in a random permutation. It also generalises some recent results on the number of inversions in randomly labelled trees by Cai, Holmgren, Janson, Johansson and Skerman. We consider complete binary trees as well as random split trees a large class of random trees of logarithmic height introduced by Devroye. Split trees consist of nodes (bags) which contain balls and are generated by a random trickle down process of balls through the nodes.

In the case of the complete binary trees that asymptotically the cumulants of the number of occurrences of a fixed permutation in the random node labelling have explicit formulas. For a random split tree with high probability we show the cumulants of the number of

occurrences are asymptotically an explicit parameter of the split tree. I will describe some results on the number of embeddings of digraphs into split trees, used in the proof of the second result, which may be of independent interest.

This is joint work with Michael Albert, Cecilia Holmgren, Tony Johansson.

3.19 Enumerating $1 \times n$ generalised permutation grid classes

Jakub Sliacan (University of Umeå, SE) and Robert Brignall (The Open University – Milton Keynes, GB)

License © Creative Commons BY 3.0 Unported license
© Jakub Sliacan and Robert Brignall

This talk is concerned with the study of $1 \times n$ almost-monotone permutation grid classes. In particular, we show that $1 \times n$ grid classes with $n - 1$ monotone cells and one context-free cell admit algebraic generating functions. Our approach is algorithmic and, in principle, allows us to enumerate any such class in particular (assuming sufficient computational resources). We give examples to illustrate the method on familiar objects.

Our methods, which leverage the inductive/recursive structure of these $1 \times n$ classes, will be the focus of the talk. We rely on combinatorial specifications of context-free classes and define operators on them which do the job of “appending a monotone class to the right of a context-free cell”. With appropriate pre- and post-processing, this constitutes the method in its entirety.

3.20 Complexity of the Single Cut-or-Join model and Partition Functions

Heather Smith (Davidson College, US) and István Miklós (Alfréd Rényi Institute of Mathematics – Budapest, HU)

License © Creative Commons BY 3.0 Unported license
© Heather Smith and István Miklós


Main reference István Miklós, Heather Smith: “The computational complexity of calculating partition functions of optimal medians with Hamming distance”, *Advances in Applied Mathematics*, Vol. 102, pp.18–82, 2019

URL <https://doi.org/10.1016/j.aam.2018.09.002>

We survey computational complexity results for the Single Cut-or-Join model for genome rearrangement, a common mode of molecular evolution. Our main result, enumerating the most parsimonious median scenarios is $\#P$ -complete, follows from a more general result for partition functions. In particular, calculating the partition function of optimal medians of binary strings with Hamming distance is $\#P$ -complete for several weight functions. This is joint work with István Miklós.

3.21 Two topics on permutation patterns

Vincent Vatter (*University of Florida – Gainesville, US*), Michael Engen, and Jay Pantone (*Marquette University, US*)

License  Creative Commons BY 3.0 Unported license

© Vincent Vatter, Michael Engen, and Jay Pantone

Main reference Michael Engen, Vincent Vatter: “Containing all permutations”, CoRR, Vol. abs/1810.08252, 2018.

URL <https://arxiv.org/abs/1810.08252>

Main reference Jay Pantone, Vincent Vatter: “Growth rates of permutation classes: categorization up to the uncountability threshold”, CoRR, Vol. abs/1605.04289, 2016

URL <https://arxiv.org/abs/1605.04289>

Main reference Vincent Vatter: “Growth rates of permutation classes: from countable to uncountable”, CoRR, Vol. abs/1605.04297, 2016


URL <https://arxiv.org/abs/1605.04297>

In this talk I will discuss two (admittedly unconnected) aspects of permutation patterns. First, I will discuss the determination of the set of all growth rates of permutation classes. Recently, in joint work with Pantone, we have increased the classification of these growth rates up to approximately 2.30, which is the point at which there begin to be uncountably many such growth rates. Given that Bevan has previously shown that all real numbers greater than approximately 2.36 are growth rates of permutation classes, the gap within which these growth rates remain unclassified is only 0.06 wide.

Secondly, I will discuss various versions of the problem of “containing all permutations”. In one version of this problem, Miller had shown in 2009 that there is a permutation of length $(n^2 + n)/2$ which contains all permutations of length n as subsequences. I will discuss how Engen and I have recently lowered this bound to $\lceil (n^2 + 1)/2 \rceil$. I then discuss another version of this problem, which has attracted media attention from such outlets as *The Verge*, *Quanta Magazine*, and *Wired*. In this version of the problem, we must contain all permutations of length n , but contiguously, not as factors, and the object that is to contain them must be a word over the same alphabet $\{1, 2, \dots, n\}$. There was a long-standing conjecture that the answer in this case was $n! + (n - 1)! + \dots + 3! + 2! + 1!$. This was disproved by Houston in 2014, who constructed such a “superpermutation” in the $n = 6$ case which had length only 872 (one less than the conjectured length). Then, last month, the science fiction author Greg Egan unveiled a construction of such a superpermutation of length only $n! + (n - 1)! + (n - 2)! + (n - 3)! + n - 3$. The best lower bound to-date (which was posted anonymously on the website *4Chan* in 2011 but not read carefully until the recent new interest in the problem) is $n! + (n - 1)! + (n - 2)! + n - 3$, leaving a gap of only $(n - 3)!$.

3.22 On Square Permutations

Stéphane Vialette (*University Paris-Est – Marne-la-Vallée, FR*)

License  Creative Commons BY 3.0 Unported license

© Stéphane Vialette

Given permutations π and σ_1 and σ_2 , the permutation π is said to be a *shuffle* of σ_1 and σ_2 , in symbols $\pi \in \sigma_1 \Delta \sigma_2$, if π (viewed as a string) can be formed by interleaving the letters of two strings p_1 and p_2 that are order-isomorphic to σ_1 and σ_2 , respectively. In case $\sigma_1 = \sigma_2$, the permutation π is said to be a *square* and $\sigma_1 = \sigma_2$ is a *square root* of π . For example, $\pi = 24317856$ is a square as it is a shuffle of the patterns 2175 and 4386 that are both order-isomorphic to $\sigma = 2143$ as shown in $\pi = 2_{43}17_85_6$. However, σ is not the

unique square root of π since π is also a shuffle of patterns 2156 and 4378 that are both order-isomorphic to 2134 as shown in $\pi = 2_{43}1_{78}5_6$.

We shall begin by presenting recent results devoted to recognizing square permutations and related concepts with a strong emphasis on constrained oriented matchings in graphs. Then we shall discuss research directions to address square permutation challenges in both combinatorics and algorithmic fields.

4 Working groups

4.1 Enumerative aspects of multiple RNA design

Mathilde Bouvel (Universität Zürich, CH), Robert Brignall (The Open University – Milton Keynes, GB), Andrew Elvey Price (The University of Melbourne, AU), and Yann Ponty (Ecole Polytechnique – Palaiseau, FR)

License © Creative Commons BY 3.0 Unported license
© Mathilde Bouvel, Robert Brignall, Andrew Elvey Price, and Yann Ponty

In this note, we compute the number of independent sets on certain graphs: unions of paths and cycles, caterpillars (a.k.a. combs), and complete binary trees. This question was raised during the open problem session of the Dagstuhl Seminar 18451 “Genomics, Pattern Avoidance, and Statistical Mechanics”, in relation with applications to RNA design. Indeed, RNA sequences compatible with a set of RNA secondary structures are in correspondence with independent sets on the dependency graph of this set of structures.

Context

At the open problem session of the Dagstuhl Seminar 18451 “Genomics, Pattern Avoidance, and Statistical Mechanics”, Yann Ponty raised the following question: what is the number of independent sets in restricted families of trees, like caterpillars or complete binary plane trees? The main motivation for this question relates to a deep connection between such independent sets and RNA designs, that we elaborate below.

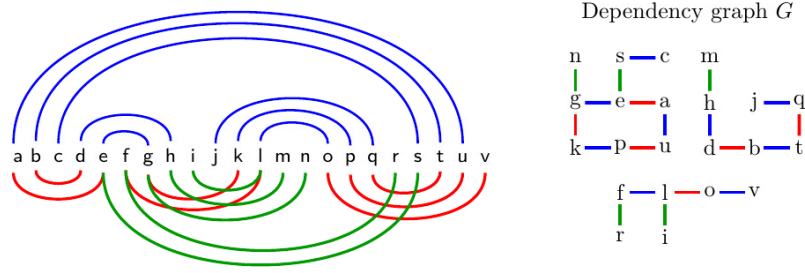
Definitions and problem statement

An *RNA secondary structure* S of length n is a set of base pairs $\{(i, j)\}$ such that $1 \leq i < j \leq n$, such that each position in $[1, n]$ is involved in at most a single base pair. Typical definitions for the secondary structure include additional constraints, for instance to preclude crossing base pairs or to ensure a minimal distance between paired positions, which we omit here for the sake of simplicity.

An RNA sequence $w \in \{A, U, C, G\}^n$ is *compatible* with a single secondary structure S of length n when

$$\forall (i, j) \in S : \{w_i, w_j\} \in \mathcal{B}, \text{ where } \mathcal{B} := \{\{G, C\}, \{A, U\}, \{G, U\}\}.$$

We say that a sequence is compatible with a set of structures \mathcal{S} (all of the same length) when it is compatible with each structure $S \in \mathcal{S}$.



■ **Figure 1** Three secondary structures and their associated dependency graph.

Given a set \mathcal{S} of structures all of length n , the set of RNA sequences compatible with \mathcal{S} depends only on the (*undirected labeled*) *dependency graph* G of \mathcal{S} , which is defined as $G = (V, E)$ with $V = [1, n]$ and $E = \cup_{S \in \mathcal{S}} \{(i, j) \in S\}$. The set of RNA sequences compatible with \mathcal{S} is denoted by $\text{Design}(G)$.

We are interested in finding computable expressions for the number of RNA sequences compatible with a given \mathcal{S} , of dependency graph G . This number is by definition

$$\#\text{Design}(G) = |\{w \in \{A, U, C, G\}^n \mid \forall S \in \mathcal{S}, w \text{ compatible with } S\}|.$$

Note that, whenever G admits an odd cycle, it is impossible to assign letters to G in a way that fulfills the compatibility requirement. It follows that $\#\text{Design}(G) = 0$ for any non-bipartite graph G , thus we restrict our scope to bipartite graphs.

► **Lemma 1.** *Denote by \mathcal{C}_G the set of connected components of a bipartite graph G . One has*

$$\#\text{Design}(G) = 2^{|\mathcal{C}_G|} \times \prod_{c \in \mathcal{C}_G} \#\text{IndSets}(c),$$

where $\#\text{IndSets}(c)$ is the number of independent sets of a (connected) graph c .

Proof. Assume first that G is connected. Given a compatible labeling of the vertices of G by letters in $\{A, U, C, G\}$, the set of vertices labeled by A or C forms an independent set, since $\{A, C\} \notin \mathcal{B}$. Conversely, given an independent set I of G , we build an RNA sequence compatible with G by assigning letters in $\{A, U, C, G\}$ to the vertices as follows:

- vertices in I are assigned A or C ;
- vertices not in I are assigned U or G ;
- choose the label of the vertex 1 (among two possibilities as above, depending on whether $1 \in I$ or not).

Because G is connected, once the label of the vertex 1 has been chosen, then all other labels are determined by the fact that all edges need to be labeled in accordance with $\mathcal{B} = \{\{G, C\}, \{A, U\}, \{G, U\}\}$. This results in a two-to-one correspondence between $\text{Design}(G)$ and the set of independent sets of G .

This immediately extends to graphs G with several connected components: in this case, we just need to choose (among two possibilities) the label of a vertex in each connected component. ◀

Note that computing $\#\text{IndSets}(G)$ is a well-studied $\#P$ -hard problem, even for graphs of maximum degree 3 [1]. Since any bipartite graph G with n vertices can be obtained as the dependency graph of $\Theta(n)$ secondary structures, computing $\#\text{Design}$ is $\#P$ -hard in general,

yet solvable in time polynomial time for dependency graphs of bounded tree width [2]. Given the hardness of the general problem, and its practical relevance to applications based on random generation, we consider enumerative properties of simple classes of dependency graphs.

5 Explicit enumeration results on constrained dependency graphs

A discussion between the four authors of this note resulted in the following results, giving formulas for $\#\text{IndSets}(G)$ (and hence $\#\text{Design}(G)$) when G is a union of paths and cycles, a caterpillar or a complete binary tree.

5.1 Union of paths and cycles

By definition, when G is the dependency graph of a set \mathcal{S} containing two structures, every vertex of G has degree at most two. Thus, every connected component of G is either a path or a cycle.

► **Lemma 1.** *The number p_n of independent sets of a path with n vertices satisfies the recurrence*

$$p_n = p_{n-1} + p_{n-2},$$

with initial conditions $p_0 = 1$, $p_1 = 2$ i.e., p_n is the $(n+2)$ -th Fibonacci number ([3, A000045]).

Proof. Let P be a path with n vertices. Consider a vertex v at an extremity of the path. There are p_{n-1} independent sets of P not containing v , and p_{n-2} which do contain v . ◀

► **Lemma 2.** *The number c_n of independent sets of a cycle with n vertices is*

$$c_n = p_{n-1} + p_{n-3}.$$

Proof. Let C be a cycle with n vertices, and v be a vertex (for instance, the one with label 1). The number of independent sets of C which do not contain v is p_{n-1} , and then number of those containing v is p_{n-3} . ◀

Putting Lemmas 1, 1 and 2 together, we obtain the following.

► **Proposition 1.** *Let f_n be the n -th Fibonacci number, defined by $f_0 = 0$, $f_1 = 1$ and $f_{n+2} = f_{n+1} + f_n$. The number of designs of a dependency graph G associated with a set \mathcal{S} of two structures is given by*

$$\#\text{Design}(G) = \prod_{p \in \mathcal{C}_G \text{ is a path}} 2 f_{|p|+2} \times \prod_{c \in \mathcal{C}_G \text{ is a cycle}} (2 f_{|c|+1} + 2 f_{|c|-1}).$$

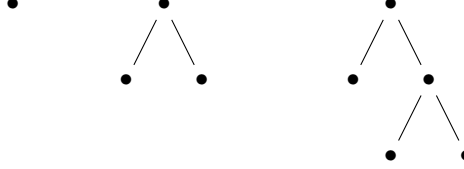
5.2 Caterpillars

We define a family of binary trees which we call *caterpillars* (and are sometimes called combs or centipede graphs in the literature) as follows. There is exactly one caterpillar of each

size $n \geq 0$. The caterpillar of size 0 is the tree with just one vertex. For any $n \geq 1$, the caterpillar of size n is the tree whose root has two children, the left child being a leaf, and the right child being the caterpillar of size $n - 1$.

Note that these trees are rooted, plane, and unlabeled.

The caterpillars of size 0 to 2 are shown in Figure 2.



■ **Figure 2** The caterpillars of size 0 to 2.

► **Lemma 1.** *The number a_n of independent sets of the caterpillar of size n satisfies the recurrence*

$$a_n = 2a_{n-1} + 2a_{n-2}$$

with initial conditions $a_0 = 2$, $a_1 = 5$. Denoting (a'_n) the sequence [3, A052945], we have $a_n = a'_{n+1}$.

Proof. Let G be the caterpillar of size n , and let v be the root of G . Let also ℓ be the left child of v , r be its right child, and u_L and u_R be the left and right children of r , respectively. An independent set I of G may or not contain v .

- If I does not contain v , then ℓ may or not be in I , and I restricted to the subtree rooted at r is a generic independent set of a caterpillar of size one less. So, there are $2a_{n-1}$ independent sets of G which do not contain v .
- If I contains v , then $\ell \notin I$ and $r \notin I$. But then, similarly to the above case, u_L may or not be in I , and I restricted to the subtree rooted at u_R is a generic independent set of a caterpillar of size two less. So, there are $2a_{n-2}$ independent sets of G which contain v . ◀

Combining Lemmas 1 and 1 yields the following.

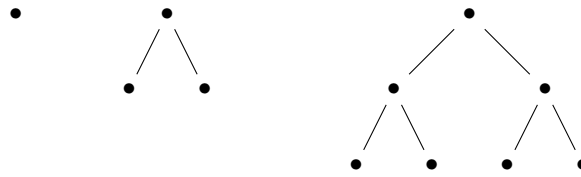
► **Proposition 2.** *Let G be a (labeled) dependency graph with $2n + 1$ vertices which admits a rooting and a planar embedding that allows to see G as a labeled caterpillar (necessarily of size n). The number of designs of G is $2a_n$.*

5.3 Complete binary trees

We consider now another family of trees containing one tree of each size: namely, the complete binary trees (where the size is the depth). Again, these trees are rooted, plane, and unlabeled. Examples are shown in Figure 3.

► **Lemma 1.** *The number b_n of independent sets of the complete binary tree of size n satisfies the recurrence*

$$b_n = b_{n-1}^2 + b_{n-2}^4$$



■ **Figure 3** The complete binary trees of size 0 to 2.

with initial conditions $b_0 = 2$, $b_1 = 5$. Denoting (b'_n) the sequence $[3, A052945]$, we have $b_n = b'_{n+2}$.

Proof. The formula follows as in the previous proofs, the term b_{n-1}^2 (resp. b_{n-2}^4) counting the independent sets which do not (resp. do) contain the root of the binary tree. ◀

From the above and Lemma 1, we have:

► **Proposition 3.** *Let G be a (labeled) dependency graph with $2^{n+1} - 1$ vertices which admits a rooting and a planar embedding that allows to see G as a labeled complete binary tree (necessarily of size n). The number of designs of G is $2b_n$.*

6 Going further

This note just records the results of a discussion during the seminar. However, it will hopefully serve as a basis for starting a collaboration. The following questions may be of interest.

- Find enumerative information on the sequences a_n and b_n , like a closed form or asymptotic behavior. Note that it will just be a routine exercise to obtain this information for a_n , but is not immediate for the case of b_n .
- Compute $\#\text{Design}(G)$ in other cases, starting with other families of trees. In particular, consider some families of trees which contain *several* trees of a given size, and express $\#\text{Design}(G)$ by a formula involving not only the size but also the value of a parameter to determine.

References

- 1 Martin Dyer and Catherine Greenhill. On markov chains for independent sets. *Journal of Algorithms*, 35(1):17 – 49, 2000. <http://dx.doi.org/https://doi.org/10.1006/jagm.1999.1071>
- 2 Stefan Hammer, Yann Ponty, Wei Wang, and Sebastian Will. Fixed-Parameter Tractable Sampling for RNA Design with Multiple Target Structures. In *RECOMB 2018 – 22nd Annual International Conference on Research in Computational Molecular Biology*, Paris, France, April 2018. Extended version under review. URL: <https://hal.inria.fr/hal-01631277>.
- 3 OEIS Foundation Inc. The encyclopedia of integer sequences, 2011. URL: <http://oeis.org>.

Participants

- Michael Albert
University of Otago, NZ
- David Bevan
University of Strathclyde –
Glasgow, GB
- Miklós Bóna
University of Florida –
Gainesville, US
- Mathilde Bouvel
Universität Zürich, CH
- Marília Braga
Universität Bielefeld, DE
- Robert Brignall
The Open University –
Milton Keynes, GB
- Cedric Chauve
Simon Fraser University –
Burnaby, CA
- Anders Claesson
University of Iceland –
Reykjavik, IS
- Michael W. Deem
Rice University – Houston, US
- Sergi Elizalde
Dartmouth College –
Hanover, US
- Andrew Elvey Price
The University of Melbourne, AU
- Péter L. Erdős
Alfréd Rényi Institute of
Mathematics – Budapest, HU
- Guillaume Fertin
University of Nantes, FR
- Yoong Kuan Goh
University of Technology –
Sydney, AU
- Torin Greenwood
Rose-Hulman Inst. of Technology
– Terre Haute, US
- Sylvie Hamel
Université de Montréal –
Québec, CA
- Christine E. Heitsch
Georgia Institute of Technology –
Atlanta, US
- E. J. Janse van Rensburg
York University – Toronto, CA
- László Kozma
FU Berlin, DE
- Anthony Labarre
University Paris-Est –
Marne-la-Vallée, FR
- Olya Mandelshtam
Brown University –
Providence, US
- István Miklós
Alfréd Rényi Institute of
Mathematics – Budapest, HU
- Alois Panholzer
TU Wien, AT
- Jay Pantone
Marquette University, US
- Seth Pettie
University of Michigan –
Ann Arbor, US
- Yann Ponty
Ecole Polytechnique –
Palaiseau, FR
- Svetlana Poznanovik
Clemson University, US
- Thomas Prellberg
Queen Mary University of
London, GB
- Pijus Simonaitis
University of Montpellier 2, FR
- Fiona Skerman
Uppsala University, SE
- Jakub Sliacan
University of Umeå, SE
- Heather Smith
Davidson College, US
- Jason Smith
University of Aberdeen, GB
- Rebecca Smith
The College at Brockport, US
- Einar Steingrímsson
University of Strathclyde –
Glasgow, GB
- Jens Stoye
Universität Bielefeld, DE
- Jessica Striker
North Dakota State University –
Fargo, US
- Krister Swenson
University of Montpellier &
CNRS, FR
- Vincent Vatter
University of Florida –
Gainesville, US
- Stéphane Vialette
University Paris-Est –
Marne-la-Vallée, FR

