



HAL
open science

Analytic Combinatorics and the Probability of the Condorcet Paradox

Emma Caizergues

► **To cite this version:**

Emma Caizergues. Analytic Combinatorics and the Probability of the Condorcet Paradox. Other [cs.OH]. Université Paris sciences et lettres, 2025. English. ⟨NNT : 2025UPSLD059⟩. ⟨tel-05551177⟩

HAL Id: tel-05551177

<https://theses.hal.science/tel-05551177v1>

Submitted on 13 Mar 2026

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization



THÈSE DE DOCTORAT
DE L'UNIVERSITÉ PSL

Préparée à Paris-Dauphine PSL

Analytic Combinatorics
and the Probability of the Condorcet Paradox

Soutenue par

Emma CAIZERGUES

Le 18 décembre 2025

École doctorale n°543

ED SDOSE

Spécialité

Informatique

Composition du jury :

Frédérique BASSINO Professeure, Université Sorbonne Paris Nord	<i>Présidente</i>
Vincent JUGÉ Maitre de conférences, Université Gustave Eiffel	<i>Rapporteur</i>
Antonin MACÉ Chargé de recherche CNRS, Paris School of Economics	<i>Rapporteur</i>
Vincent MERLIN Directeur de recherche CNRS, Université de Caen-Normandie	<i>Examineur</i>
Marc NOY Professeur, Universitat Politècnica de Catalunya	<i>Examineur</i>
François DURAND Ingénieur de recherche, Nokia Bell Labs	<i>Co-encadrant</i>
Élie DE PANAFIEU Ingénieur de recherche, Nokia Bell Labs	<i>Co-encadrant</i>
Jérôme LANG Directeur de recherche CNRS, Université Paris Dauphine-PSL	<i>Directeur de thèse</i>

Abstract

This thesis studies the probability of the Condorcet paradox, a situation in which the aggregation of individual preferences yields a cyclic outcome, despite each voter having transitive preferences.

While most previous work focused on probabilistic approaches, we investigate an alternative route based on analytic combinatorics. We first consider the uniform distribution of preferences using a graph-based encoding of voting profiles, leading to significant results on regular graphs and balanced matrices that stand on their own. We then extend the analysis to more general preference models via saddle point methods, not only recovering known limits for the probability of a Condorcet winner but also refining them by providing explicit convergence rates. This approach also produces results in multivariate analytic combinatorics that are of independent interest.

Finally, since we assume throughout the thesis that voters provide complete rankings, the last chapter departs from the probabilistic study to address a practical question: how can we help voters to efficiently elicit their full preferences, particularly when the number of candidates is large? To tackle this problem, we propose sorting algorithms and discuss their potential applications in real-world voting contexts, illustrated by the 2025 Australian federal election.

Résumé

Cette thèse étudie la probabilité du paradoxe de Condorcet, une situation dans laquelle l'agrégation des préférences individuelles conduit à un cycle, même quand chaque électeurice possède des préférences transitives.

Tandis que la plupart des travaux précédents utilisent des approches probabilistes, nous explorons une voie alternative fondée sur la combinatoire analytique. Nous commençons par étudier la distribution uniforme des préférences en encodant un profil de vote sous forme de graphe. Au passage, nous obtenons des résultats pertinents sur les graphes réguliers et les matrices équilibrées. Nous étendons ensuite l'analyse à des modèles de préférences plus généraux. Via des méthodes de point col, non seulement nous retrouvons des limites connues pour la probabilité d'existence du paradoxe de Condorcet, mais nous obtenons en plus des taux de convergence. Cette approche fournit également des résultats en combinatoire analytique multivariée présentant un intérêt indépendant.

Enfin, puisque l'ensemble de la thèse suppose que les électeurices fournissent des classements complets, le dernier chapitre s'écarte de l'étude probabiliste pour aborder une question pratique : comment aider les électeurices à construire efficacement leur classement ? Nous proposons une procédure de tri et discutons de ses applications potentielles dans des contextes réels de vote à travers l'exemple de l'élection fédérale australienne de 2025.

Acknowledgement

First and foremost, I would like to thank my three advisors: Élie de Panafieu, *archimage des séries biscornues*, François Durand, *thaumaturge de la manipulabilité*, and Jérôme Lang, *grand oracle du choix social*. I am so grateful for their kindness, constant support, and insightful guidance throughout these three years. Élie, your enthusiasm for mathematics is truly contagious. Thanks to you, analytic combinatorics felt far less intimidating than it initially seemed, despite the difficulty of the domain. Fanfan, your clarity and pedagogical skills will always impress and inspire me. Thank you as well for introducing me to the wonderful world of social choice. Jérôme, thank you for making me feel so welcomed in Dauphine, and for always showing genuine excitement and curiosity about my research. Also, thank you all three for carefully revising this thesis and for helping me prepare the defense. I feel incredibly lucky to have had you by my side throughout this experience. Beyond the mathematics, thank you for being such wonderful humans. I will remember both our great working sessions and the many good moments we shared outside of work.

Then, I would like to thank Frédérique Bassino, Vincent Jugé, Antonin Macé, Vincent Merlin, and Marc Noy for accepting to be part of my jury. Especially, thank you Vincent and Antonin for agreeing to be my reviewers and for providing such detailed and insightful reports. Seeing such genuine interest in my thesis truly warmed my heart.

During this PhD, I collaborated with such great researcher: Marc Noy, Vlady Ravelomanana, Guillem Perarnau, Fabien Mathieu, Fanfan, and Élie. Working with you was a pleasure and a very enriching experience, of course because you all are very brilliant, but also because of your kindness and patience, so thank you so much for our collaboration! I owe a special thanks to Fabien, *haut dignitaire du textile bigarré*, notably for revising Chapter 6, and for helping me prepare the defense as well. Thank you grandpa! I would also like to thank the people at IUT d'Orsay with whom I had the opportunity to work in teaching. It was such a pleasant teaching environment, and I truly enjoyed working there. In particular, I would like to thank Banafsheh Farang-Hariri, Francesca Fiorenzi, and Élodie Leducq for always being there to help when I needed it. I want also to warmly thank Benjamin Hellouin for introducing me to research by being my first internship advisor at "la Venise du numérique" (in 2021, time really flies!) and for recommending me to teach at IUT, because it was a very fulfilling experience. Finally, speaking of internships, I owe a huge thank you to Johanne Cohen, who told me about the internship at Nokia that ultimately led to

my PhD. This thesis would probably not exist without her, so thank you Johanne for your good recommendation!

Also, despite my short academic career, I had the chance to meet wonderful people across several labs, and I owe a big thank you to many of them. First, I would like to thank the people at Dauphine, especially the PhD students, for always making me feel welcomed, even though I was a CIFRE outsider. In particular, thank you Ariane, Bastien, Houria, Jiaxin, Louise, Lucas, Matthieu, Nicolas, Sarra, Théo, Thibault, for making the long rides on the RER B to join Dauphine truly worth it! Then, I would like to thank my team at Nokia for making my experience there so enjoyable. I am especially grateful to my manager, Matthew, who placed full trust in my work and always gave me the freedom to explore social choice topics, showing genuine interest along the way. Beyond Nokia, I also had the pleasure of meeting amazing people at LINCS. So, to both the Nokia and LINCS teams, thank you: Ana, Cristina, Élie, Fabien, Fanfan, Francesca, François, Guillaume, Leonardo, Lorenzo, Ludovic, Marc-Olivier, Mathieu, Maxime, Rémy, and Sara. I am grateful for the stimulating discussions and the enjoyable moments we shared together. Finally, I was lucky enough to spend some time working in Barcelona, where I also felt truly welcomed. Thank you Marc for making it possible and for our great collaboration. You always show so much interest in both people and their work and it is truly enjoyable to work by your side. Also, thank you to the rest of the team for welcoming me so warmly both times, in particular Clément, Guillem, Lluís, Miquel, Patrick, and Tabriz. I will always remember my stay there, and will be forever impressed by how calm everyone was when the electricity shut down in the whole country (!!).

I would now like to thank my family for supporting me throughout this huge project that is completing a PhD. They may not have fully understood how I spent my days over the past three years, but they always showed me pride and trust in my project. For that trust, as well as their financial and emotional support, I am deeply grateful.

Finally, I owe a heartfelt thank you to my chosen family. Doing a mostly remote PhD can sometimes feel incredibly lonely, and your presence, whether through messages or in real life, made everything so much more bearable. So, thank you Alice, Anaïs, Andreï, Antoine, Amélie, Apolline, Corentin, Coraline, Eve, Isabella, Juliette, Justine, Léo, Lucie, Manon, Mélissa, Morgane, Nicolas, and Vanessa, for caring so deeply about me and for being such thoughtful and attentive people. When it comes to the PhD, I would especially like to thank Eve, Vanessa, and Nicolas for always answering my (sometimes silly) math questions, and Léo for helping me too many times with my computer problems lol.

Contents

1	Introduction	1
2	Preliminaries	7
2.1	Social Choice Notions and Notations	7
2.2	Introduction to Analytic Combinatorics	10
2.2.1	Symbolic Method	12
2.2.2	Analytic Extraction	23
3	Enumeration of Graphs with Degree Constraints	39
3.1	Introduction	39
3.1.1	Related Work	40
3.1.2	Contributions	42
3.1.3	Limitations	43
3.1.4	Roadmap	43
3.2	Encoding the Condorcet Problem Into Graph Enumeration	44
3.3	Notations and Conventions	46
3.4	Bipartite Graphs	48
3.4.1	Bipartite Weighted Multigraphs	49
3.4.2	Bipartite Simple Graphs	53
3.5	Applications	58
3.5.1	Condorcet Winner	58
3.5.2	Balanced Matrices	61
3.6	General Graphs (Non Bipartite)	65
3.6.1	Weighted Multigraphs	67
3.6.2	Simple Graphs	72
3.7	Conclusion	80
4	Probability of the Condorcet Paradox for Large Electorates	81
4.1	Introduction	81

4.1.1	Motivation	81
4.1.2	Contributions	82
4.1.3	Related Work	83
4.1.4	Limitations	85
4.1.5	Roadmap	86
4.2	Preliminaries	86
4.3	From our Combinatorial Problem to its Analytic Formulation	88
4.3.1	Symbolic Method	88
4.3.2	Coefficient Extraction via Cauchy Integrals	91
4.3.3	Saddle Point Method	92
4.4	Satisfying the Conditions of the Asymptotic Theorems in Chapter 5	95
4.5	Subcritical Case	97
4.5.1	Theoretical Result in the Subcritical Case	97
4.5.2	Application: Mallows Culture \mathcal{M}_3 last	98
4.6	Critical Case	101
4.6.1	Theoretical Result in the Critical Case	101
4.6.2	Application: Impartial Culture	103
4.7	Mixed Case: Subcritical and Critical	105
4.7.1	Theoretical Result in the Mixed Case	106
4.7.2	Application: Mallows Culture \mathcal{M}_4 last	106
4.8	Dealing With Supercriticality	108
4.8.1	Theoretical Result for Supercriticality	108
4.8.2	Application: Mallows Culture \mathcal{M}_3 first	111
4.9	Asymptotic Expansion	114
4.10	Conclusion	116
5	Multivariate Analytic Combinatorics	119
5.1	Introduction	119
5.1.1	Related Work and Contributions	120
5.1.2	Summary of our Contributions and Roadmap	122
5.1.3	Preliminaries	123
5.2	Analytic Conditions for Applying the Saddle Point Method	124
5.2.1	Rank and Periodicity of a Multivariate Generating Function	124
5.2.2	The Daffodil Lemma	129
5.3	Large Powers	132
5.4	Combining Large Powers and Singularity Analysis	136

5.5	Asymptotic Expansion	139
5.6	Conclusion	145
6	Anytime Sorting Algorithms	147
6.1	Introduction	147
6.1.1	Comparison-Based Elicitation	152
6.1.2	Related Work	153
6.1.3	Contributions	154
6.1.4	Limitations	155
6.1.5	Roadmap	156
6.2	Transforming Classical Sort Into an Anytime Version	157
6.2.1	Classical Sorting	157
6.2.2	Universal Estimators	160
6.2.3	Estimator Tailored For Mergesorts	163
6.3	New Algorithms	165
6.3.1	Multizip Sort	165
6.3.2	Comparison-Oriented Sort	169
6.4	Evaluation and Comparative Results	172
6.4.1	Uninterrupted Behavior: Termination Time	173
6.4.2	Performance Profiles	174
6.4.3	Implementation	177
6.5	Australian Federal Election	179
6.5.1	Existing Tools	179
6.5.2	Anytime Sorting Tool	184
6.6	Conclusion	185
7	Conclusion	187
	Bibliography of Chapters 1 and 2	191
	Bibliography of Chapter 3	193
	Bibliography of Chapter 4	197
	Bibliography of Chapter 5	201
	Bibliography of Chapter 6	203

Introduction

What is a voting rule? Before beginning this PhD, I used to think of voting rules simply as the mechanism used on election day, where each voter places a single name on a ballot and the candidate with the most votes either reaches the second round or wins. Yet, voting rules encompass far more than this familiar picture. In theory, voting rules are formal procedures for transforming individual preferences into collective decisions.

In particular, these individual preferences can take the form of *rankings*, where each voter provides an ordered list of all candidates according to their preference. Systems that rely solely on rankings belong to the family of *ordinal voting rules*, which consider only the relative ordering of candidates rather than any numerical score or intensity of preference. The rankings given on the ballots can be either truncated, and in the extreme case contain only the top-ranked candidate, or complete, as illustrated in Figure 1.1, where voters express their preference rankings over a set of 3 candidates (here fruits).

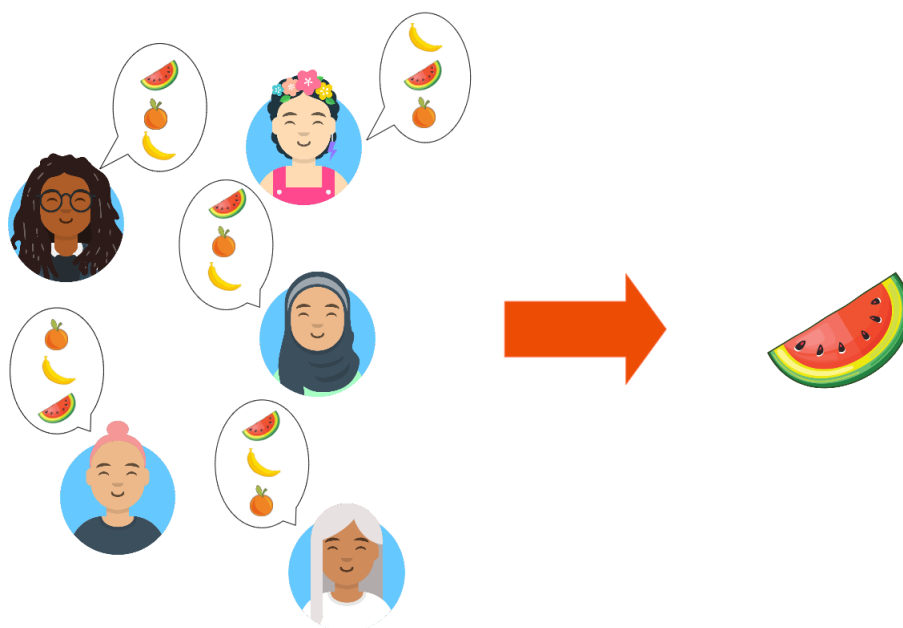


Fig. 1.1: Illustration of a voting rule. On the left, voters having preferences over the 3 candidates: banana, orange and watermelon. The red arrow represents a voting rule that would elect the watermelon.

The difficulty of aggregating rankings. Even before considering the specific mechanisms of voting rules, the aggregation of ranked preferences already gives rise to interesting and sometimes paradoxical phenomena. In 1785, Nicolas de Condorcet [Con85] introduced the notion that now bears his name, the *Condorcet winner*: a candidate who defeats every other candidate in a pairwise majority comparison. Formally, a candidate w is a Condorcet winner if, for any other candidate c , a strict majority of voters prefer w to c . In Figure 1.1, for instance, the watermelon is a Condorcet winner. This concept shows that the preference profile of a population can exhibit a natural structure even before specifying any particular voting rule. Yet, as Condorcet also observed, such a winner does not always exist because individual rational preferences can combine into collectively inconsistent outcomes. This phenomenon, called the *Condorcet paradox*, occurs when majority preferences form a cycle. For instance, if three voters rank the candidates 1, 2, and 3 as

$$1 > 2 > 3, \quad 2 > 3 > 1, \quad 3 > 1 > 2,$$

then each candidate is defeated by another in a head-to-head contest, and no Condorcet winner exists. This paradox exemplifies one of the fundamental difficulties in collective decision-making: even when each voter behaves rationally and expresses a consistent ranking, the aggregation of their preferences may lead to a collective inconsistency.

These difficulties are formalized in several key theorems in social choice theory. The most famous is Arrow's impossibility theorem [Arr50], which considers the problem of aggregating individual rankings into a collective ranking. Arrow identified a few requirements that one might reasonably expect from such an aggregation rule: it should work for all possible preference profiles (*unrestricted domain*), it should not reduce to simply following the will of a single voter (*non-dictatorship*), it should respect unanimous agreement (*Pareto efficiency*), and the relative comparison between two options should not depend on the presence of other alternatives (*independence of irrelevant alternatives*). Arrow's theorem states that no rule can satisfy all of these conditions at once when there are at least three alternatives.

A complementary perspective is provided by the Gibbard-Satterthwaite theorem [Gib73], which considers voting rules that output a single winner rather than a full ranking. It shows that any voting rule that is not dictatorial and can elect at least three different candidates in some profiles is necessarily vulnerable to strategic behaviour: voters may have an incentive to misreport their preferences to obtain a more favourable outcome. In other words, manipulation is unavoidable, except in trivial cases.

The axiomatic framework. Since these impossibility theorems show that no voting rule can satisfy simultaneously all desirable criteria, part of the literature has focused on defining and classifying voting rules according to the properties that they might reasonably be expected to satisfy, namely the *axiomatic* approach. For example, Young [You74] showed that the *Borda rule* is the unique voting rule that satisfies *neutrality* (treating all candidates equally), *reinforcement* (if two disjoint profiles elect the same candidate, their union also elects that candidate), *faithfulness* (if there is only one voter, the winner is their top candidate) and *cancellation* (whenever all pairwise majority margins cancel out across the electorate, the rule must return a perfectly tied outcome).

The axiomatic framework has provided powerful and elegant characterisations of voting rules. By formalizing normative principles, such as *Condorcet consistency* (always electing the Condorcet winner whenever there is one), it allows us to characterize what different aggregation mechanisms can or cannot achieve. Yet, by its very nature, the axiomatic approach usually yields qualitative results: a rule either satisfies a given property or it does not. Even though there exist a few axiomatic works that reason in probabilistic terms about the satisfaction of axioms, such as the framework developed by Bardier [Bar25], these do not aim to compute probabilities of satisfaction or violation, but rather to define an axiomatic perspective grounded in probability.

The probabilistic approach. A complementary probabilistic perspective seeks to enrich this analysis by quantifying how frequently these properties hold or fail under specific models of voter preferences. Rather than replacing the axiomatic framework, it extends it, shifting the focus from the logical possibility of violations to their empirical likelihood. For example, knowing the frequency of the Condorcet paradox can help inform the choice of an appropriate voting rule for a given context. If the probability of encountering the paradox is low, it may be reasonable to favor a rule that always elects the Condorcet winner, since situations without one are unlikely to arise. Conversely, if the probability of the paradox is high, then focusing primarily on electing the Condorcet winner may be less relevant, and other axiomatic properties, such as reinforcement or *participation* (a voter cannot achieve a more favorable outcome by abstaining rather than voting), might be more decisive. Therefore, the axiomatic and probabilistic approaches go hand in hand, each complementing the other to better guide the choice of an appropriate voting rule.

A number of studies have investigated the likelihood of the Condorcet paradox under various models of voter preferences, as documented in the book by Gehrlein [Geh06], which is entirely dedicated to this topic. Early work often focused on specific distributions, such as the uniform distribution, named *Impartial Culture (IC)*, and computed exact probabilities for small numbers of voters or candidates. Later contributions extended these results to asymptotic settings.

Among these contributions, Niemi and Weisberg [NW68] derived the limiting probability of the Condorcet paradox as the number of voters tends to infinity, assuming only that voter preferences are independent. More recently, Krishnamoorthy and Raghavachari [KR05] revisited and modernized this result. In this thesis, we refer to the framework used by these authors as the *General Independent Culture (GIC)*. Another interesting work is the one of Sauermann [Sau22], where she examined scenarios in which the number of voters is fixed while the number of candidates is large ($m \rightarrow \infty$), under the IC model, which is a special case of GIC.

A common denominator of most of these studies is their reliance on probabilistic methods. For instance, Krishnamoorthy and Raghavachari [KR05] employ techniques from multinomial distribution theory to obtain their results, while Sauermann [Sau22] makes use of classical probabilistic tools, such as Bonferroni-type inequalities, to derive bounds on the likelihood of certain events.

Another line of research has leveraged combinatorial and geometric methods, particularly through *Ehrhart polynomials*, which allow to compute probabilities of events such as the Condorcet paradox. These methods are specifically tailored for the *Impartial Anonymous Culture (IAC)*, where voters are anonymous and all rankings are equally likely. Extending these techniques to other settings, such as the GIC model, remains challenging.

Analytic combinatorics. This motivates the use of *analytic combinatorics*, which provides a systematic framework for deriving asymptotic properties of combinatorial structures. Its principle is to encode combinatorial objects into algebraic structures, whose coefficients correspond to quantities of interest. By seeing these algebraic objects as complex functions, one can then apply techniques from complex analysis, thus then extract precise asymptotic estimates for these coefficients. This methodology has been successfully applied in areas ranging from algorithm analysis and random structures to statistical physics.

Its origins can be traced back to the early 20th century: Hardy and Ramanujan [HR17] used complex analysis and number-theoretic ideas to evaluate the asymptotics of integer partitions in 1917. In 1956, Hayman [Hay56] introduced a key technique in the field, the *saddle point method*, providing a generalization of Stirling's formula and laying the foundations for modern asymptotic enumeration.

The field expanded significantly in the following decades, with many contributors refining the methods and extending their scope. In particular, Knuth [Knu99] made extensive use of these techniques in the study of algorithmic complexity, contributing to the emergence of a distinct area now known as the analysis of algorithms. Flajolet and Sedgewick [FS09] also made substantial contributions and standardized the methods, notably through their book *Analytic Combinatorics*,

which presents the subject with a comprehensive framework, clear notation, and a wide range of applications. This work serves as a primary reference throughout this thesis.

Overview of the contributions. In this thesis, we investigate how analytic combinatorics can be applied to the study of the Condorcet paradox under the GIC model. Our goal is to derive asymptotic probabilities for the existence of a Condorcet winner. We begin with a graph-based approach under the uniform distribution, before extending our analysis to the general GIC model using the saddle point methods. Along the way, we will see that this perspective not only sheds light on classical questions in social choice theory, but also leads to results of independent interest from the standpoint of analytic combinatorics. Finally, since the entire manuscript assumes that voters express preferences as complete rankings, we conclude with a side project that steps away from probabilistic analysis and examines how such preferences can be elicited in practice.

Even if the primary focus of the thesis is the probability of the Condorcet paradox, I also aim to present this thesis as both an illustration of applied results in analytic combinatorics and as an accessible introduction to the field. Therefore, even readers without a background in analytic combinatorics should be able to learn the main techniques and, maybe hopefully one day, apply them in their own research. Likewise, readers with no background in social choice theory can follow this thesis without prior preparation.

Structure of the thesis. The chapters of this thesis are designed to be as independent as possible. To this end, each chapter includes its own introduction, discussion of related work, and outline of contributions, allowing readers to approach them selectively. Readers already familiar with analytic combinatorics may freely navigate to any chapter of interest. However, Chapters 3 and 5 rely on techniques explained in Section 2.2 of Chapter 2, which should be read beforehand by non-specialists.

To balance readability and modularity, certain definitions, such as that of the Condorcet winner, are recalled in all chapters using it. While this may occasionally introduce redundancy, I believe it is preferable to forcing constant back-and-forth between sections. At the same time, all social choice terminology is collected in one dedicated place, namely Section 2.1 of Chapter 2, for readers wishing to review this material in a single block.

Chapter 2 therefore serves as a reference chapter, introducing the necessary background in both social choice theory and analytic combinatorics. It also presents a guiding example that illustrates the main steps of the analytic combinatorics approach developed throughout the thesis.

Chapter 3 presents our first method for computing the probability of the Condorcet paradox, based on a graph-theoretic encoding. This method applies to the uniform distribution (Impartial Culture).

Chapter 4 and Chapter 5 present a second method, which extends the analysis to the GIC model, providing a new proof of the result by Niemi and Weisberg [NW68] as well as additional contributions, including convergence rates and applications to specific models such as IC and Mallows. Chapter 4 outlines the overall strategy and states the main results, while Chapter 5 develops the analytic combinatorics machinery required to derive them.

Chapter 6 departs from the probabilistic analysis and addresses a practical question: how can individuals elicit their own preferences efficiently? Throughout the thesis, we assume that voters hold complete rankings over candidates, yet when the number of candidates becomes large, eliciting such rankings can be cognitively demanding. Therefore, we address this question by proposing an efficient sorting algorithm for constructing preferences and discuss its potential applications to real-world voting contexts, notably the 2025 federal elections in Australia.

Chapter 7 concludes.

Preliminaries

2.1 Social Choice Notions and Notations

In this section, I introduce the main definitions and notations from social choice theory that will be used throughout the thesis. Most of these notions will be recalled later in the core chapters, since it would be unrealistic to expect the reader to remember all details. Nonetheless, I find it useful to gather here in one place all the basic concepts that will serve as reference points for the rest of the manuscript.

Throughout the thesis, we define the set of voters as $\{1, 2, \dots, n\}$ where n is the number of voters, and the set of candidates as $\{1, \dots, m\}$, where m is the number of candidates. A *profile* represents the preferences of the voters and is defined as a list of n rankings over the set of candidates. A *culture* refers to a probability distribution over the $(m!)^n$ possible profiles.

We assume that voters have independent and identically distributed preferences, meaning the culture is characterized by a probability p_r for each preference ranking r . Surprisingly, there is no standard terminology for this common assumption, which we refer to as a *General Independent Culture (GIC)* in Chapter 4. We also assume that the culture is *generic*, in the sense that $p_r > 0$ for every ranking r . The notation \mathbb{P} refers to the probability under the given culture.

The GIC model notably encompasses several classical cultures, such as the *Impartial Culture (IC)*, where all rankings are equally likely and which will serve in Chapters 3 and 4, as well as the *Mallows model* [Mal57], which will notably illustrate theoretical results in Chapter 4. The Mallows model is characterized by a reference ranking r_0 over the candidates and a concentration parameter $\rho \in \mathbb{R}_{\geq 0}$. The probability of a ranking r is given by

$$p_r := \gamma e^{-\rho d(r, r_0)},$$

where γ is a normalization constant ensuring $\sum_r p_r = 1$, and $d(r, r_0)$ denotes the Kendall-tau distance [Ken38], which counts the minimum number of adjacent swaps needed to transform r into r_0 . Remark that when $\rho = 0$, all rankings are equally probable, recovering the IC model.

The thesis is dedicated to study the probability of occurrence of a *Condorcet winner*. A *Condorcet winner* is a candidate who is preferred to every other candidate by strictly more than $\frac{n}{2}$ voters. When there is a Condorcet winner, it is unique. The absence of a Condorcet winner is called the *Condorcet paradox*. Without loss of generality, we will study $\mathbb{P}(m \text{ is CW})$, the probability that candidate m is the Condorcet winner.

2	2	2	2	4	3
1	1	2	2	3	3
2	3	1	3	1	2
3	2	3	1	2	1

2	2	2	4	4	1
1	1	2	2	3	3
2	3	1	3	1	2
3	2	3	1	2	1

Fig. 2.1: Voting profiles illustrating the Condorcet winner and the Condorcet paradox for $m = 3$ candidates and $n = 15$ voters. Each profile lists all possible rankings, with the number of voters supporting each ranking shown above in blue. **Top:** a profile in which candidate 3 is the Condorcet winner. **Bottom:** a profile exhibiting the Condorcet paradox, *i.e.* no Condorcet winner exists.

Figure 2.1 illustrates two voting profiles with $m = 3$ candidates and $n = 15$ voters. In the top profile, candidate 3 is the Condorcet winner, since it is preferred to candidate 1 by $2 + 4 + 3 = 9 > \lfloor 15/2 \rfloor$ voters, and similarly preferred to candidate 2 by $2 + 4 + 3 = 9$ voters. The bottom profile exhibits the Condorcet paradox: no candidate is a Condorcet winner. Indeed, candidate 3 is preferred to candidate 1 by 9 voters, candidate 1 is preferred to candidate 2 by 8 voters, and candidate 2 is preferred to candidate 3 by 8 voters, forming a cycle.

In Chapter 4, our theoretical results extend this question to a more general notion, which we call an α -winner, defined as follows. Let $\mathcal{A} := \{1, \dots, m - 1\}$ be the set of adversaries of candidate m . Consider a vector $\alpha := (\alpha_1, \dots, \alpha_{m-1}) \in (0,1)^{m-1}$, where each α_j represents the proportion of voters that candidate m needs on their side to win the pairwise comparison against candidate j . We say that candidate m is an α -winner if m is preferred to each adversary j by more than $\alpha_j n$ voters. The standard notion of a Condorcet winner corresponds to the special case $\alpha = (\frac{1}{2}, \dots, \frac{1}{2})$. The notion of a *generalized Condorcet winner*, as introduced by Sertel and Sanver [SS04], is recovered by considering a vector α whose coordinates are equal.

Formally, a *voting rule* (or *voting system*) is a function that takes a preference profile as input and returns a set of winning candidates (possibly of size one).

In Chapter 6, we notably discuss about the *Instant-Runoff Voting* system. This voting rule is also known as *Ranked-Choice Voting (RCV)*, *Single Transferable Voting (STV)*, *Preferential Voting*, or the *Alternative Vote*, and proceeds as follows:

1. Each voter submits a ranking of all candidates from most to least preferred.
2. The first-place votes are counted.
3. If one candidate obtains an absolute majority (more than half of the votes), that candidate is declared the winner.
4. Otherwise, the candidate with the fewest first-place votes is eliminated. Ballots ranking this candidate first are then transferred to the next preferred candidate listed on each ballot.
5. Steps 2 to 4 are repeated until one candidate achieves an absolute majority and is elected.

When ties occur, two main tie-breaking approaches can be distinguished. The first, *immediate tie-breaking*, resolves ties as they arise using a deterministic or random rule (for instance, a predefined candidate order or a random draw). The second, known as the *parallel universes* approach, simultaneously considers all possible sequences of tie-breaking decisions. In this framework, each possible choice defines a separate "parallel universe" in which IRV is executed, and the overall set of winners corresponds to all candidates who win in at least one of these universes. Note that this method does not necessarily yield a single winner.

Figure 2.2 illustrates the execution of the Instant-Runoff Voting (IRV) process on a profile with $m = 3$ candidates and $n = 11$ voters. The profile of votes is shown in the first table. Each column corresponds to a specific ranking of the candidates (displayed in black), while the number of voters who submitted that ranking is indicated above in blue. In this initial profile, candidate 1 receives 3 first-place votes, while candidates 2 and 3 each receive 4 first-place votes. Therefore, candidate 1 has the fewest first-place votes and is eliminated according to the IRV procedure. The ballots are then updated by removing candidate 1 from all rankings. This updated set of ballots is displayed in the second table, and the remaining votes are transferred to each voter's next preferred candidate. The third table shows these transferred votes to illustrate the resulting profile after the elimination of candidate 1. By examining this resulting profile, we see that candidate 3 now holds the majority of first-place votes among the remaining candidates. Consequently, candidate 3 wins the election in the second round.

1	2	4	0	2	2
1	1	2	2	3	3
2	3	1	3	1	2
3	2	3	1	2	1

1	2	4	0	2	2
2	3	2	2	3	3
3	2	3	3	2	2

5	6
2	3
3	2

Fig. 2.2: Execution of the IRV process on a profile with $m = 3$ candidates and $n = 11$ voters. The first table shows the initial profile of votes. Each column represents a ranking (in black), with the number of voters choosing that ranking indicated above in blue. Candidate 1 receives the fewest first-place votes and is therefore eliminated. The second table displays the ballots after candidate 1 is removed, and the third table shows the aggregated profile. In the second round, candidate 3 secures the majority and thus wins the election.

Instant-Runoff Voting (IRV) is widely implemented in Australia, not only in political elections but also in various non-governmental decision-making contexts, such as job recruitment processes or committee selections. Beyond Australia, IRV is employed in several national elections across the world. Notably, it is used to elect the heads of state in India, Ireland, and Sri Lanka. A variant in which ballots are truncated to the top three candidates is also used to elect members of the National Parliament of Papua New Guinea.

2.2 Introduction to Analytic Combinatorics

The goal of this section is to introduce the main tools from *analytic combinatorics* that will be used throughout the thesis. In broad terms, the approach of analytic combinatorics can often be seen as proceeding in two steps. The first step is the *symbolic method* (presented in Section 2.2.1), which allows one to translate the structure of a combinatorial class into an algebraic object. The second step is the *analytic extraction* (developed in Section 2.2.2), which notably includes a classical

method of analytic combinatorics, namely the *saddle point method*. So, readers already familiar with analytic combinatorics, and in particular with these notions, may safely skip this section.

I use both techniques in Chapters 3 to 5, but while I apply them in *multivariate* settings, I chose to present them here on a *univariate* guiding example, to provide a more accessible introduction. Nonetheless, the results and methods described in this section naturally generalize to the multivariate case, and are detailed when necessary in the thesis.

It was not easy to find a single example that could clearly illustrate both the symbolic method and analytic extraction in a simple way. Indeed, a simple symbolic method example often comes with a rather involved analytic extraction. I will give an intuitive explanation of why this happens in Section 2.2.2. In general, I find analytic extraction to be more challenging to approach than the symbolic method, in particular because it involves theorems from *complex analysis*. By contrast, the symbolic method can be explained step by step, starting with the fundamentals and gradually building up to the final problem. This incremental approach makes it possible to teach the method effectively even when the underlying example is complex. Furthermore, the symbolic method used in Chapter 3 contains several subtleties, so I believe that starting with a slightly more complicated case will better prepare the reader to tackle that chapter. For these reasons, I chose an example that is somewhat more intricate on the symbolic side but relatively classical on the analytic extraction side. Moreover, the example I chose has the advantage to resemble the analytic extraction performed in Chapters 4 and 5.

As a running example, I will use *fragmented permutations* of size n . This example is treated in the book by Flajolet and Sedgewick [FS09]. In this section, I will expand on their treatment and provide additional details.

A *fragmented permutation* on a set of n elements is a set of *fragments*, where each fragment is a non-empty sequence of distinct elements, and every element in $\{1, \dots, n\}$ appears exactly once overall. In other words, a fragmented permutation is a decomposition of the n elements into subsequences.

Consider the set $\{1, 2, 3, 4, 5\}$. A (classical) permutation is a single ordered list containing all elements exactly once, for example:

$$(1, 5, 4, 3, 2).$$

A *fragmented permutation* is a set of *non-empty sequences* (called *fragments*), where each element in $\{1, \dots, 5\}$ appears in exactly one fragment. For instance,

$$\{(1, 5, 4), (3), (2)\}$$

is a fragmented permutation consisting of three fragments: the sequence $(1, 5, 4)$ and the single-element sequences (3) and (2) .

Within each fragment, the order of the elements is significant. However, the *global* order between fragments is not. For example,

$$\{(1, 5, 4), (3), (2)\} \quad \text{and} \quad \{(3), (1, 5, 4), (2)\}$$

represent the *same* fragmented permutation.

On the other hand,

$$\{(1, 5, 4), (3), (2)\} \quad \text{and} \quad \{(4, 1, 5), (3), (2)\}$$

are *different* fragmented permutations, since the order within the first fragment has changed.

2.2.1 Symbolic Method

To find the number of fragmented permutations, we begin by applying what is known as the *symbolic method*. The idea is to introduce formal variables linked to specific parameters that define our counting problem. As mentioned earlier, applying the symbolic method to fragmented permutations is not the easiest task, mainly because the underlying objects are **labeled**. For this reason, I will first present the symbolic method on a simpler application case involving *unlabeled* objects, and only afterwards return to the study of fragmented permutations. To this end, I will begin by motivating the definition of the central object in the symbolic method, namely the *generating function*, as I consider it a crucial notion to understand before applying the method itself.

Motivation

The beauty of the symbolic method is that it establishes a dictionary between constructions on *combinatorial classes* and algebraic operations. Before explaining in detail what this entails, let us first give a formal definition of a combinatorial class.

Formally, a *combinatorial class* (or *combinatorial family*) is a set \mathcal{A} equipped with a *size function*

$$|\cdot| : \mathcal{A} \rightarrow \mathbb{N},$$

such that, for each $k \in \mathbb{N}$, the number of objects of size k in \mathcal{A} is finite. For instance, let \mathcal{A} be the class of binary words, that is, finite sequences over the alphabet $\{0,1\}$. We define the size of a word to be its length. With this definition, \mathcal{A} is indeed a combinatorial class.

Now, to provide a dictionary between combinatorial classes and algebraic operations, we seek an algebraic object that satisfies three criteria:

- It admits a natural operation for extracting the number of objects of size n ;
- Disjoint union corresponds to addition;
- Cartesian product corresponds to multiplication.

As we shall see just below, these requirements lead naturally to the following object.

Definition 1. Let \mathcal{A} be an unlabeled combinatorial class, and let a_k denote the number of objects of size k in the class \mathcal{A} . The *ordinary generating function (OGF)* of \mathcal{A} is defined as the formal power series

$$A(z) = \sum_{k \geq 0} a_k z^k.$$

At this stage, the generating function appears to serve merely as a compact and formal encoding of the combinatorial class under consideration. Let us check that all three conditions are satisfied.

First condition. For the class of binary words we already know the counting sequence $a_k = 2^k$, hence $(1, 2, 4, 8, \dots)$. Therefore, the generating function of \mathcal{A} rewrites

$$A(z) = \sum_{k \geq 0} 2^k z^k.$$

The first criterion is then satisfied: once an explicit expression for $A(z)$ is available, the number a_n is recovered by *coefficient extraction*. The coefficient extraction is denoted by

$$[z^n]A(z) := a_n.$$

Of course, in our case, the generating function is easy to explicit. But in some cases, it requires a few more steps, which is the case of fragmented permutations, as we shall see later.

Second condition. To illustrate how disjoint union corresponds to addition in generating functions, we partition the class of binary words according to the parity of the number of 0's. Let $\mathcal{A}_{\text{even}}$ denote the subclass of words containing an *even* number of 0's, and \mathcal{A}_{odd} the subclass with an *odd* number of 0's. By construction, these two subclasses are disjoint, and their union is exactly the original class:

$$\mathcal{A} = \mathcal{A}_{\text{even}} \cup \mathcal{A}_{\text{odd}}.$$

Let $A_{\text{even}}(z)$ and $A_{\text{odd}}(z)$ be the ordinary generating functions of $\mathcal{A}_{\text{even}}$ and \mathcal{A}_{odd} , respectively. For a word of length k , the number of words with exactly ℓ zeros is $\binom{k}{\ell}$, as it corresponds to choosing ℓ positions among k to place the zeros. Hence, the number of words with an even number of zeros is

$$a_k^{\text{even}} = \sum_{\substack{\ell=0 \\ \ell \text{ even}}}^k \binom{k}{\ell},$$

and the number with an odd number of zeros is

$$a_k^{\text{odd}} = \sum_{\substack{\ell=0 \\ \ell \text{ odd}}}^k \binom{k}{\ell}.$$

The corresponding generating functions are

$$A_{\text{even}}(z) = \sum_{k \geq 0} a_k^{\text{even}} z^k, \quad A_{\text{odd}}(z) = \sum_{k \geq 0} a_k^{\text{odd}} z^k.$$

Now, by the binomial theorem, for each k we have

$$a_k^{\text{even}} + a_k^{\text{odd}} = \sum_{\ell=0}^k \binom{k}{\ell} = 2^k.$$

This yields

$$A_{\text{even}}(z) + A_{\text{odd}}(z) = \sum_{k \geq 0} 2^k z^k = A(z),$$

thus confirming that, in this case, the disjoint union of combinatorial classes corresponds to the sum of their generating functions.

More generally, this property holds for any combinatorial classes. If \mathcal{B} and \mathcal{C} are two disjoint combinatorial families with generating functions $B(z)$ and $C(z)$, then the generating function of their disjoint union $\mathcal{A} = \mathcal{B} \cup \mathcal{C}$ is simply

$$A(z) = B(z) + C(z).$$

Third condition. Now, let us form the product $\mathcal{A} \times \mathcal{A}$, whose objects are ordered pairs (u, v) where u and v are binary words. The size of such a pair is naturally defined as the sum of the lengths of u and v .

The number of pairs (u, v) of total size k is then

$$\sum_{\ell=0}^k a_{\ell} a_{k-\ell},$$

The generating function $A_{\text{prod}}(z)$ associated with $\mathcal{A} \times \mathcal{A}$ is therefore

$$A_{\text{prod}}(z) = \sum_{k \geq 0} \left(\sum_{\ell=0}^k a_{\ell} a_{k-\ell} \right) z^k,$$

which is precisely the Cauchy product (the usual product of power series) of $A(z)$ with itself:

$$A_{\text{prod}}(z) = A(z) \times A(z).$$

This example illustrates concretely that the Cartesian product of combinatorial classes corresponds to the multiplication of their generating functions.

More generally, this property holds for any combinatorial classes. If \mathcal{B} and \mathcal{C} are two combinatorial classes with generating functions $B(z)$ and $C(z)$, respectively, and if the size of a pair $(u, v) \in \mathcal{B} \times \mathcal{C}$ is defined as $|(u, v)| = |u| + |v|$, then the generating function of their Cartesian product is

$$A(z) = B(z) C(z).$$

Explaining the Symbolic Method on Unlabeled Structures.

Now that we are convinced that generating functions are appropriate objects to work with, it remains to explain how to make them explicit using the symbolic method. In the present example we already know the exact value of the sequence of a_k , but let us now attempt to recover it through the symbolic method itself: this consists in introducing a **formal variable** associated with the parameter of interest, and then unfolding the symbolic constructions.

In the case of binary words, the parameter of interest is the word length, *i.e.* the number of letters in the word. We therefore associate to a letter the formal variable z . Since a word of length k can be seen as the Cartesian product of k letters, it is naturally associated with the monomial z^k . This

makes the definition of the generating function even more natural, as the operation of coefficient extraction

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

precisely recovers the number of binary words with n letters.

Let us now apply the symbolic method to recover explicitly the generating function of binary words. As said above, we start from the atomic class \mathcal{Z} , consisting of a single object of size 1, which here consists in a letter. Its generating function is simply

$$Z(z) = z.$$

A *binary letter* is not just an occurrence of the atomic class \mathcal{Z} , but rather a choice between two possible realizations, namely 0 or 1, i.e.,

$$\mathcal{L} = \{0\} \cup \{1\}.$$

Here, I think it is worth clarifying why binary words are *unlabeled* objects, despite the apparent distinction between 0 and 1. In the context of analytic combinatorics, an object is considered labeled if each element carries a unique label that cannot be repeated. In a binary word, the letters 0 and 1 serve only as **types** and not as unique labels: multiple occurrences of 0 or 1 are allowed in the same word.

Symbolically, each of these choices corresponds to one distinct copy of the atomic class, so we can write

$$\mathcal{L} = \mathcal{Z} \cup \tilde{\mathcal{Z}},$$

where both \mathcal{Z} and $\tilde{\mathcal{Z}}$ are distinct copies of the atomic class. Since both alternatives contribute the same factor $Z(z)$, the generating function of the class of binary letters is

$$L(z) = Z(z) + Z(z) = 2z.$$

A *binary word* is a finite sequence of letters. Symbolically, this corresponds to the combinatorial construction

$$\mathcal{A} = \text{Seq}(\mathcal{L}).$$

By definition, a sequence is either empty, or one letter, or the cartesian product of two letters, and so on. Translating this, we sum over all possible lengths:

$$A(z) = 1 + L(z) + L(z)^2 + L(z)^3 + \dots = \sum_{k \geq 0} L(z)^k.$$

Note that we recover the geometric series

$$A(z) = \frac{1}{1 - L(z)}.$$

Finally, substituting $L(z) = 2z$, we get

$$A(z) = \frac{1}{1 - 2z} = \sum_{k \geq 0} 2^k z^k.$$

In a nutshell, we have constructed the generating function of \mathcal{A} step by step, by introducing a formal variable z associated with letters, describing the objects in \mathcal{A} , and applying the dictionary we developed earlier (disjoint union corresponding to sum, and Cartesian product to multiplication).

Notice that we made use of the construction *Sequence*, and showed that it translates to the algebraic operation $z \rightarrow \frac{1}{1-z}$. Along with disjoint union and Cartesian product, it is one of the most classical operators in analytic combinatorics. Other standard operators exist, but for the sake of conciseness, I will not present them here. For readers interested in these constructions, for unlabeled structures, they are thoroughly presented in [FS09, Chapter I], and a comprehensive summary of the standard combinatorial operations is given in [FS09, Figure I.18].

Incorporating Labels.

So far, we worked on the *unlabeled* class of binary words. Fragmented permutations, however, are defined on labeled elements (namely, integers). In this setting, the generating function must be adapted to take the labels into account, so that the standard correspondences, disjoint union to addition and Cartesian product to multiplication, remain valid.

At first sight, one might think of defining the generating function exactly as in the unlabeled case. This would still work for the disjoint union, since counting objects in a disjoint union simply amounts to adding their respective counts, independently of whether the objects are labeled or not. However, it fails as soon as we consider the product of labeled combinatorial classes, since the relabeling of atoms must be taken into account.

Indeed, following Flajolet and Sedgewick [FS09], the product of two labeled objects x and y is defined as the set of all possible **well-labeled** pairs that reduce to (x, y) . The crucial notion here is that of being *well-labeled*. Any object resulting from the product must itself be labeled, and moreover *properly* labeled: this means that all labels appearing in the new object are distinct, and that it is possible to **unambiguously recover** which component came from x and which from y .

Let us see this on a concrete example: consider the combinatorial class \mathcal{A}_{lab} of **labeled** sets of atoms. Warning, this does not correspond to permutations, as we consider **sets** and not **sequences**, for instance, $\{1, 2, 3\}$ is the same set as $\{3, 1, 2\}$, so the number of such sets of size n is 1, up to relabeling.

Consider two labeled sets of sizes 3 and 2. Their product corresponds to a labeled set of size 5. To form this product on the ground set $\{1, 2, 3, 4, 5\}$, we must choose which 3 labels are assigned to the first component, while the remaining 2 labels go to the second component. Equivalently, we can think of it as coloring the ground set $\{1, 2, 3, 4, 5\}$: elements in the first set are colored **blue**, and elements in the second set are colored **red**. Each such coloring corresponds to a distinct labeled product. Therefore, the number of possibilities is simply the number of ways of choosing 3 (or equivalently 2) elements among 5:

$$\begin{aligned} \{1, 2, 3, 4, 5\} &\text{ equivalently } \{1, 2, 3\}, \{4, 5\} \\ \{1, 2, 3, 4, 5\} &\text{ equivalently } \{1, 2, 4\}, \{3, 5\} \\ &\vdots \\ \{1, 2, 3, 4, 5\} &\text{ equivalently } \{3, 4, 5\}, \{1, 2\} \end{aligned}$$

Thus, the product $\{1, 2, 3\} \times \{1, 2\}$ yields $\binom{5}{3} = 10$ distinct labeled objects.

To encode this correctly, a **set of k labeled atoms** should not be represented by the term z^k as before, but rather by the term $\frac{z^k}{k!}$, as we will illustrate shortly. This $k!$ factor, in essence, accounts for all possible ways of assigning labels to the k elements. Consequently, we define the "new" generating function and the associated coefficient extraction as follows.

Definition 2. Let \mathcal{A} be a **labeled** combinatorial class, and let a_n denote the number of objects of size n in the class \mathcal{A} . The *exponential generating function (EGF)* of \mathcal{A} is defined as the formal power series

$$A(z) = \sum_{k=0}^{\infty} a_k \frac{z^k}{k!}.$$

Accordingly, the coefficient extraction is then given by

$$n![z^n]A(z) = a_n.$$

Returning to our example, the exponential generating function of \mathcal{A}_{lab} is

$$A_{\text{lab}}(z) = \sum_{k=0}^{\infty} \frac{z^k}{k!}.$$

Note that coefficient extraction works as expected:

$$n![z^n]A_{\text{lab}}(z) = n! \cdot \frac{1}{n!} = 1.$$

Now, the exponential generating functions of $\{1, 2, 3\}$ and $\{1, 2\}$ are $\frac{z^3}{3!}$ and $\frac{z^2}{2!}$. Their product is

$$\frac{z^3}{3!} \cdot \frac{z^2}{2!} = \frac{z^5}{3!2!}.$$

Extracting the coefficient gives

$$5![z^5] \frac{z^5}{3!2!} = \frac{5!}{3!2!} = \binom{5}{3} = 10,$$

which is exactly the number of labeled products we enumerated manually¹. If we (incorrectly) used an ordinary generating function, the product of a size-3 set and a size-2 set would be encoded as

$$z^3 \cdot z^2 = z^5,$$

so that $[z^5]z^5 = 1$. This would incorrectly yield only one object. This confirms that exponential generating functions provide the correct algebraic translation for the product of labeled combinatorial classes.

Formally, in the general case, if \mathcal{B} and \mathcal{C} are labeled combinatorial classes with exponential generating functions $B(z)$ and $C(z)$, and if \mathcal{A} denotes their *relabelled product*, then

$$A(z) = B(z)C(z).$$

¹Recall that this does not contradict the fact that \mathcal{A}_{lab} has exactly one object of each size: here we are not counting the objects of \mathcal{A}_{lab} themselves, but the different ways in which two such objects can be combined in the product construction, yielding 10 possibilities as said above.

This mirrors the situation in the unlabeled setting, where the Cartesian product also translates into multiplication of generating functions.

To summarize, in the univariate case, we call *ordinary generating functions* the ones that track *unlabeled* objects, while *exponential generating functions* track *labeled* objects via the $k!$ factor. In the multivariate case, we instead speak of *ordinary variables* for unlabeled components and *exponential variables* for labeled ones. We will use this nomenclature throughout the following chapters when dealing with multivariate generating functions.

Note that, unless explicitly computed, the generating function serves primarily as a **compact** and **formal** representation of the combinatorial problem. The main goal is always to make it explicit in order to perform coefficient extraction.

With this background in place, we are now ready to define the generating function of fragmented permutations and to make it explicit.

Applying the Symbolic Method on Fragmented Permutations.

Recall that a *fragmented permutation* is a set of *non-empty sequences* (called *fragments*), where each element in $\{1, \dots, 5\}$ appears in exactly one fragment. For instance,

$$\{(1, 5, 4), (3), (2)\}$$

is a fragmented permutation consisting of three fragments: the sequence $(1, 5, 4)$ and the single-element sequences (3) and (2) .

In the case of binary words, the relevant parameter was the number of letters, thus the formal variable z was representing a letter. Here, the relevant parameter is the number of elements in the permutation, so we introduce the formal variable z associated to each one of these elements. As they are labeled, we define the *exponential generating function* $G(z)$ of fragmented permutations:

$$G(z) = \sum_{k \geq 0} g_k \frac{z^k}{k!},$$

where the coefficient g_n is given by the coefficient extraction

$$n! [z^n] G(z) = n! [z^n] \sum_{k \geq 0} g_k \frac{z^k}{k!} = g_n.$$

Expliciting the generating function. To express an explicit form of the generating function $G(z)$, we follow the same step-by-step combinatorial construction as in the binary words case.

Let us first construct the generating function of a single fragment. A fragment is a *non-empty sequence* of elements. If a sequence contains k elements, it contributes the term

$$k! \frac{z^k}{k!} = z^k.$$

Here, the factor $k!$ accounts for the number of ways to order the k labeled elements within the sequence. Indeed, while a **set** considers only which elements are present, a **sequence** also cares about their order, hence the coefficient $k!$ arises naturally. For example, a fragment of size 3 corresponds to the term z^3 . Whether the labels are $(1, 3, 2)$ or $(2, 5, 4)$, there are always $3!$ permutations of the chosen labels, matching the coefficient extraction

$$3! [z^3] z^3 = 3!.$$

Summing over all $k \geq 1$, we obtain

$$z + z^2 + z^3 + \dots = \frac{z}{1-z}.$$

Now, a fragmented permutation is a *set* of such non-empty sequences. As previously said, a set of k labeled atoms has generating function $\frac{z^k}{k!}$. Replacing z with the generating function of any fragment we get:

$$\frac{1}{k!} \left(\frac{z}{1-z} \right)^k.$$

Let us look at the meaning of this generating function on a small example. For instance,

$$\frac{1}{2} \left(\frac{z}{1-z} \right)^2$$

is the generating function of fragmented permutations with exactly 2 fragments. Expanding the inner sum gives

$$\frac{1}{2} \sum_{k=2}^{\infty} (k-1) z^k.$$

Therefore, the number of fragmented permutations of size 3 with exactly 2 fragments is given by

$$3! [z^3] \frac{1}{2} \sum_{k=2}^{\infty} (k-1) z^k = 3! \times \frac{2}{2} = 3!.$$

This matches exactly the six possibilities:

$$\{(1), (2, 3)\}, \quad \{(1), (3, 2)\}, \quad \{(2), (1, 3)\}, \quad \{(2), (3, 1)\}, \quad \{(3), (1, 2)\}, \quad \{(3), (2, 1)\}.$$

Finally, summing over all $k \geq 0$, we get the generating function for fragmented permutations:

$$G(z) = \exp\left(\frac{z}{1-z}\right).$$

In summary, the generating function we obtain is a direct consequence of the symbolic method: we build generating functions using disjoint unions, which translate into sums of generating functions, and exponential products, which translate into products of generating functions. Beyond these basic operations, we use two additional constructions—*Sequence* and *Set*. The sequence operation corresponding to $z \mapsto \frac{1}{1-z}$ and the set operation to the exponential series.

As we saw previously, in the unlabeled setting, similar operations exist: the sequence operation also corresponds to $z \mapsto \frac{1}{1-z}$. Again, for readers interested in these constructions, they are thoroughly presented in [FS09, Chapter I], and a comprehensive summary of the standard combinatorial operations is given in [FS09, Figure I.18]. In contrast, the constructions for labeled structures are treated in [FS09, Chapter II], with a summary in Figure II.18.

Now, let us go back to finding the number of fragmented permutations of size n . We have

$$g_n = n! [z^n] \exp\left(\frac{z}{1-z}\right).$$

Several techniques can be employed to extract the exact coefficient $[z^n]G(z)$, such as solving recurrence relations, applying known identities, or using direct algebraic manipulations. Here, the exact coefficient can be computed explicitly by expanding $\left(\frac{z}{1-z}\right)^k$ for each k and summing over all k . However, this involves a fair amount of calculation, and in some situations the asymptotic behaviour of the coefficients is more informative than their exact values. In our case, this asymptotic can also be derived directly from the exact formula, but this requires several tedious steps and the use of Stirling's formula, which actually itself follows from the *saddle point method*, which is a part of what follows.

2.2.2 Analytic Extraction

The saddle point method is a cornerstone of analytic combinatorics. To apply it, one must first interpret the *formal* power series as *complex* functions. This is precisely the beauty of analytic combinatorics: it bridges the algebraic viewpoint with the analytic perspective. This idea of using complex analysis to study coefficients originates in arithmetic, where early applications focused on estimating number-theoretic functions such as the distribution of prime numbers. A comprehensive reference on the topic is the textbook by Tenenbaum [Ten15].

In this subsection, we introduce all the necessary background to prepare for the saddle point method, and then apply it to our guiding example.

Cauchy Integrals

As previously said, we now interpret generating functions not merely as formal power series, but as analytic functions of a complex variable. Let us first recall the notion of analyticity.

Definition 3 (Analytic function.). Let $U \subset \mathbb{C}$ be an open set and let $g : U \rightarrow \mathbb{C}$ be a complex-valued function. We say that g is *analytic* at a point $\omega \in U$ if there exists a power series expansion

$$g(z) = \sum_{k=0}^{\infty} g_k(z - \omega)^k,$$

which converges to $g(z)$ for all z in some neighborhood of ω . If g is analytic at every point of U , then g is said to be analytic on U . Note that the multivariate analogue of this definition is presented in Chapter 5, see Definition 10.

Remark. Not every formal power series is analytic. For instance, the series $\sum_{k \geq 0} k! z^k$ diverges for every $z \neq 0$, and thus has radius of convergence 0.

A complex function defined by a power series is analytic only within its disk of convergence. However, even if the series diverges outside this disk or on its boundary, the function it defines may still admit a meaningful analytic extension beyond it. Let $F(z)$ be analytic on an open disk U_F , and let z_0 be a point on the boundary of U_F . If there exists a function $\tilde{F}(z)$, analytic on a disk $U_{\tilde{F}}$ centered at z_0 , such that $F(z) = \tilde{F}(z)$ for all $z \in U_F \cap U_{\tilde{F}}$, then $\tilde{F}(z)$ is called an *analytic continuation* (or *analytic extension*) of $F(z)$.

For instance, consider the series $F(z) = \sum_{n \geq 0} z^n$, whose radius of convergence is 1. The point -1 lies on the boundary of this disk. To obtain an analytic continuation at this point, we use the fact that $F(z) = 1/(1 - z)$. We may rewrite

$$\frac{1}{1 - z} = \frac{1}{2} \cdot \frac{1}{1 - \frac{1+z}{2}} = \sum_{n \geq 0} \frac{(1+z)^n}{2^{n+1}} =: \tilde{F}(z).$$

The new series $\tilde{F}(z)$ has radius of convergence 1, and is therefore analytic at $z = -1$. Moreover, it coincides with $F(z)$ on all points satisfying both $|z| < 1$ and $|1 + z| < 1$, and thus provides an analytic continuation of $F(z)$ at -1 .

Definition 4 (Singularity.). Let g be a complex function analytic on an open domain $\mathcal{D} \subset \mathbb{C}$. A point $z_0 \in \mathbb{C}$ is called a *singularity* (or *singular point*) of g if g cannot be analytically continued to any neighbourhood of z_0 . Equivalently, z_0 is a singularity if it lies on the boundary of \mathcal{D} and g admits no analytic continuation across it.

Having defined what is a singularity, we now turn the notion of a *residue*, which describes the local behavior of a function near an isolated singularity. This concept forms a crucial link between formal power series and complex analysis, allowing us to express coefficients as complex integrals and apply analytic tools to combinatorial problems.

Definition 5 (Residue). Let g be a complex function defined on a punctured disk around $\omega \in \mathbb{C}$, and suppose that g admits a Laurent expansion:

$$g(z) = \sum_{k=-\infty}^{\infty} g_k(z - \omega)^k.$$

The *residue* of g at ω , denoted $\text{Res}_{z=\omega} g(z)$, is the coefficient g_{-1} :

$$\text{Res}_{z=\omega} g(z) := g_{-1}.$$

We now turn to the notion of a *positively oriented simple closed curve*, which provides the proper setting for integrating analytic functions around isolated singularities. This concept ensures that integrals along such curves behave consistently, allowing us to deform paths without changing the integral's value as long as singularities are avoided.

Definition 6 (Positively Oriented Simple Closed Curve). A simple closed curve γ around a point $w \in \mathbb{C}$ is called *positively oriented* if it can be continuously deformed to a parametrization of the form

$$t \mapsto w + re^{2i\pi t}, \quad t \in [0,1],$$

while remaining entirely within the punctured disk

$$\{z \in \mathbb{C}: 0 < |z - w| < R\}.$$

We are now ready to invoke a classical result from complex analysis, stated in Theorem 1.

Theorem 1 (Cauchy's Residue Theorem). *Let g be analytic on an open domain except for isolated singularities, and let γ be a positively oriented, simple closed contour. Suppose ω is an isolated singularity of g lying inside γ , and that g is analytic in a neighbourhood of γ . Then*

$$\int_{\gamma} g(z) dz = 2\pi i \cdot \text{Res}_{z=\omega} g(z).$$

Proof. Let g be analytic on an open domain except at an isolated singularity ω , and let γ be a positively oriented simple closed contour enclosing ω , with g analytic on and inside γ except at ω .

Since ω is an isolated singularity, there exists a punctured neighbourhood around ω where g admits a Laurent expansion:

$$g(z) = \sum_{k=-\infty}^{\infty} g_k(z - \omega)^k.$$

Choose a small circle C_ε of radius $\varepsilon > 0$ around ω entirely contained in this neighbourhood. γ can be continuously deformed to C_ε without crossing any singularity, so

$$\int_{\gamma} g(z) dz = \int_{C_\varepsilon} g(z) dz.$$

On C_ε , parametrized as $z = \omega + \varepsilon e^{i\theta}$, $\theta \in [0, 2\pi]$, we can integrate term-by-term the Laurent series:

$$\int_{C_\varepsilon} g(z) dz = \sum_{k=-\infty}^{\infty} g_k \int_0^{2\pi} (\varepsilon e^{i\theta})^k i \varepsilon e^{i\theta} d\theta = 2\pi i g_{-1}.$$

Hence,

$$\int_{\gamma} g(z) dz = 2\pi i \cdot \text{Res}_{z=\omega} g(z). \quad \square$$

A particularly useful consequence in our setting is the following.

Corollary 1 (Cauchy's coefficient formula). *Let $f(z)$ be analytic in a domain containing a simple closed contour γ encircling the origin. Then, for every $n \in \mathbb{N}$,*

$$[z^n]f(z) = \frac{1}{2\pi i} \oint_{\gamma} \frac{f(z)}{z^{n+1}} dz.$$

Proof. This follows directly from the Residue Theorem applied to the function $g(z) = \frac{f(z)}{z^{n+1}}$, which has a simple pole at $z = 0$ with residue equal to the coefficient $f_n = [z^n]f(z)$. Hence,

$$[z^n]f(z) = \frac{1}{2\pi i} \oint_{|z|=r} \frac{f(z)}{z^{n+1}} dz. \quad \square$$

Remark. Note that this formula is both applicable to ordinary generating functions and exponential generating functions, with the appropriate factorial normalization when necessary, as done below.

In our setting, since $G(z)$ is analytic within the radius of convergence $R = 1$, Cauchy's formula applies and gives an integral representation of the coefficients. In particular, the quantity $[z^n]G(z)$, which corresponds to the number of fragmented permutations of size n , divided by $n!$, can be expressed as the contour integral

$$\frac{g_n}{n!} = [z^n]G(z) = \frac{1}{2\pi i} \oint_{\gamma} \frac{G(z)}{z^{n+1}} dz,$$

where γ is a positively oriented simple closed curve centered at the origin with radius $r < R$.

Saddle Point Method

Framework. To estimate the integral

$$[z^n]G(z) = \frac{1}{2\pi i} \oint_{\gamma} \frac{G(z)}{z^{n+1}} dz$$

for large n , a powerful tool from complex analysis is the *saddle point method*. This method notably permits to evaluate complex integrals of the form

$$\oint_{\mathcal{C}} A(z)e^{n\phi(z)} dz,$$

where n is large and the functions $A(z)$ and $\phi(z)$ are analytic. Integrals of this form correspond to the classical *large powers* setting. Fragmented permutations, however, do not fall under this case. Indeed, in our setting we consider the integral

$$[z^n]G(z) = \frac{1}{2\pi i} \oint_{\gamma} \exp(\log G(z) - n \log z) \frac{dz}{z}.$$

So we have

$$\phi(z) = \log G(z) - n \log z = \frac{z}{1-z} - n \log z, \quad \text{and} \quad A(z) = \frac{1}{z},$$

which is slightly more intricate than the standard large powers case, since the exponential function ϕ itself depends on n . This setting falls within a broader framework known as *Hayman admissibility* that I will not detail, but the reader may refer to [FS09, Definition VIII.1] if interested.

Note that such situations are common when applying the saddle point method to combinatorial examples, due to the form of the integrand $\frac{G(z)}{z^{n+1}}$, where $G(z)$ typically does not depend on n .

Intuition behind the saddle point method. In complex analysis, the value of an integral

$$\oint_{\gamma} g(z) dz$$

remains unchanged under any continuous deformation of the path γ , provided the endpoints are preserved and no singularity of the integrand $g(z)$ is crossed. In our setting, this means that γ must remain a positively oriented simple closed curve avoiding the singularities of $g(z)$. The purpose of such deformations is to select an integration path along which the dominant contributions are concentrated around a few isolated points. This is desirable because we can then apply local approximations, such as Taylor expansions of $g(z)$, and reduce the integral to a simpler form.

Now, to do so, observe that a crude upper bound on any integration path γ is

$$\left| \oint_{\gamma} g(z) dz \right| \leq \oint_{\gamma} |g(z)| |dz| \leq (\text{length of } \gamma) \cdot \sup_{z \in \gamma} |g(z)|.$$

Minimizing this bound ensures that the magnitude of the integral remains close to its theoretical upper limit, thereby forcing the path to pass through the points that contribute most significantly to the integral. To achieve this, we must select a path that minimizes $\sup_{z \in \gamma} |g(z)|$. Geometrically, if we visualize the surface $|g(z)|$ over the complex plane, this corresponds to passing through a "mountain pass" between two peaks: along the chosen contour γ , the point appears as a maximum,

but in the transverse direction it is a minimum, as illustrated in Figure 2.3. No nearby deformation of the contour can reduce this value further.

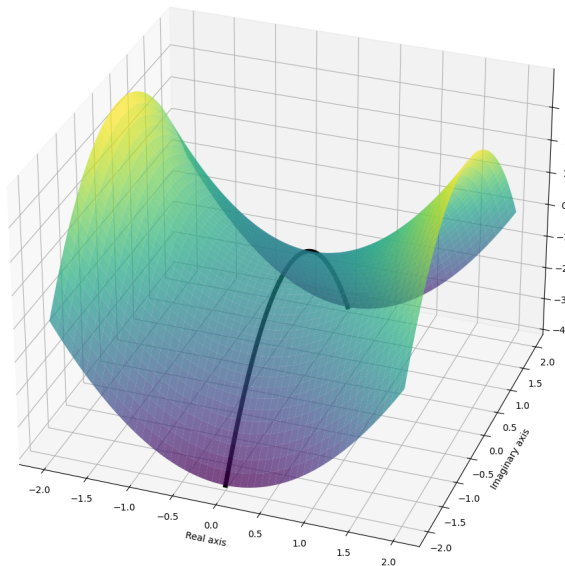


Fig. 2.3: Canonical "mountain pass" surface $z = x^2 + (iy)^2 = x^2 - y^2$, and the adequate path (in black) passing through it. This "mountain pass" geometry provides the intuition behind the saddle point method in complex analysis, where integration paths are deformed to pass through such points.

Since the modulus of a non-constant analytic function cannot have a local maximum (Maximum Modulus Principle), any point ζ with $g'(\zeta) = 0$ exhibits a mountain pass-like behavior. Along the chosen path, it appears as a local maximum, while in some transverse direction it is a local minimum. Consequently, the path of minimal maximum must pass through the points where $g'(z) = 0$. Those points are called the *saddle points* of g .

Definition 7 (Saddle point). Let g be analytic. A point ζ is called a *saddle point* of g if

$$g'(\zeta) = 0.$$

The name "saddle point" comes from the shape of the surface of the modulus of g near such a point: it resembles a saddle, as shown in Figure 2.3.

In practice, we are often interested in integrals of the form

$$\oint_{\gamma} A(z)e^{\phi(z)} dz,$$

where $A(z)$ varies slowly while the exponential term $e^{\phi(z)}$ dominates the asymptotics when the parameter tends to infinity (here n). Since the growth or decay of the integrand is essentially dictated by the real part $\operatorname{Re}(\phi(z))$, it is sufficient to focus on the stationary points of $\phi(z)$. The prefactor $A(z)$ only contributes to multiplicative corrections, but does not determine the location of the main contribution. Thus, in our context, a saddle point ζ is defined as satisfying the condition

$$\phi'(\zeta) = 0.$$

Note that, in the case of fragmented permutations, the function ϕ depends on n , and consequently the saddle point ζ also varies with n . For the sake of readability, we choose not to make this dependence explicit in the notation, but it must be kept in mind.

Finally, when the integrand is a series with nonnegative coefficients, the maximum modulus on any circle centered at zero is reached on the positive real axis (direct consequence of the triangular inequality). Hence, in combinatorial applications, the saddle point is always real and positive, which we can see in Figure 2.4.

Location of the saddle point for fragmented permutations. For fragmented permutation, recall that the function ϕ is equal to

$$\phi(z) = \log(G(z)) - n \log(z).$$

So we look for the saddle point ζ that solves

$$\phi'(z) = \frac{d}{dz} \log G(z) - \frac{n}{z} = 0$$

By classic derivate computations this is equivalent to

$$\frac{1}{(1-z)^2} - \frac{n}{z} = 0$$

which leads to

$$nz^2 - (2n+1)z + n = 0$$

This quadratic equation gives two solutions

$$\zeta_{\pm} = \frac{2n+1 \pm \sqrt{4n+1}}{2n}.$$

We choose the solution that is lower than 1, as 1 is a singularity of the function and we rather avoid it when possible. The exact solution is

$$\zeta = \frac{2n + 1 - \sqrt{4n + 1}}{2n} = 1 - \frac{\sqrt{4n + 1}}{2n} + \frac{1}{2n},$$

Using Taylor expansions, we compute an approximation that will be useful later to justify and perform expansions of ϕ :

$$\zeta = 1 - \frac{1}{\sqrt{n}} + \frac{1}{2n} + \mathcal{O}(n^{-3/2}).$$

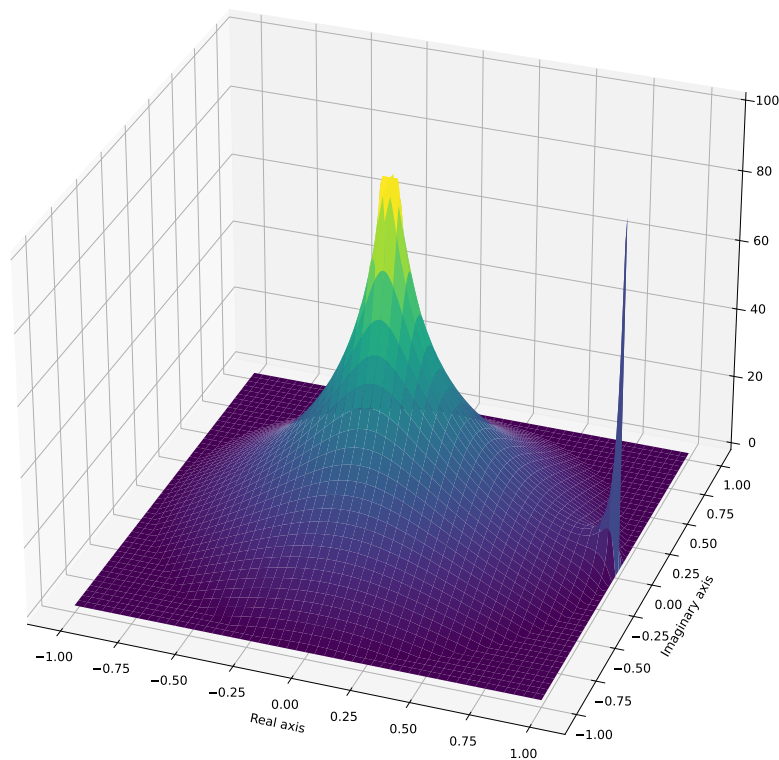


Fig. 2.4: Visualization of the saddle point for our guiding example. The 3D plot shows $\log(|G(z)/z^n|)$ in the complex plane for $n = 30$, where the logarithm is used for better visualization. The plot exhibits two peaks at $z = 0$ and $z = 1$, with the saddle point ζ located near the latter.

Figure 2.4 visualizes $\log(|G(z)/z^n|)$ in the complex plane for $n = 30$, where the logarithm is applied purely for visualization purposes. Therefore, this plot highlights the saddle point structure. At the saddle point ζ (near 1), the surface exhibits a local maximum along the imaginary axis and a

local minimum along the real axis, confirming the characteristic shape that justifies the use of the saddle point method for asymptotic analysis.

Let us choose as our contour the circle centered at the origin with radius ζ . We decompose this contour into two parts:

- The *central part* γ_n , a small arc around the point ζ whose size depends on n . More precisely, it is the arc of the circle of radius ζ passing through ζ , with endpoints $\zeta e^{-i\varepsilon_n}$ and $\zeta e^{i\varepsilon_n}$, where $\varepsilon_n = n^{-\beta}$ for some $\beta > 0$ to be specified later.
- The *tail part* (or *tails*), consisting of the remainder of the circle, which we will show to be negligible in the asymptotic analysis.

Central part. The central part is

$$\frac{1}{2\pi i} \oint_{\gamma_n} \exp(\phi(\zeta)) \frac{dz}{z}$$

We perform the parametrization $z = \zeta e^{i\theta}$. As we have $\theta \in [-\varepsilon_n, \varepsilon_n]$, we get

$$\frac{1}{2\pi} \int_{-\varepsilon_n}^{\varepsilon_n} \exp(\phi(\zeta e^{i\theta})) d\theta$$

Expanding the integrand near the saddle point ζ amounts to performing a Taylor expansion of ϕ around $\theta = 0$, and then discarding terms that become negligible as n tends to infinity. In principle, the required order of the expansion is not known a priori. In practice, it is often sufficient to expand up to order 3 and then determine the conditions under which the cubic term becomes negligible, verifying afterwards that all higher-order terms are indeed negligible. As we shall see later, this approach is valid in our case; hence, we restrict the Taylor expansion to this order. Nevertheless, it is important to keep in mind that, in more delicate situations, higher-order terms may need to be included.

Notations. Because we will need to differentiate $\theta \mapsto \phi(\zeta e^{i\theta})$, we introduce, for any differentiable function $F(z)$, the notations

$$\partial_z F(z) := F'(z) \quad \text{and} \quad (z\partial_z)^2 F(z) := z\partial_z(z\partial_z F(z)).$$

Observe that for any differentiable function $F(z)$

$$\partial_\theta F(\zeta e^{i\theta}) = ((iz\partial_z)F(z))_{z=\zeta e^{i\theta}}.$$

Therefore, for any k ,

$$\partial_{\theta=0}^k \phi(\zeta e^{i\theta}) = ((iz\partial_z)^k \phi(z))_{z=\zeta}.$$

Expanding the integrand around the saddle point ζ yields

$$\frac{1}{2\pi} \int_{-\varepsilon_n}^{\varepsilon_n} \exp \left(\phi(\zeta) + \partial_{\theta=0} \phi(\zeta e^{i\theta}) \theta + \partial_{\theta=0}^2 \phi(\zeta e^{i\theta}) \frac{\theta^2}{2!} + \partial_{\theta=0}^3 \phi(\zeta e^{i\theta}) \frac{\theta^3}{3!} + \dots \right) d\theta.$$

Factoring out the term $e^{\phi(\zeta)}$ and using the fact that the saddle point cancels the linear term $\partial_{\theta=0} \phi(\zeta e^{i\theta}) = \zeta \phi'(\zeta) = 0$, we obtain

$$\frac{e^{\phi(\zeta)}}{2\pi} \int_{-\varepsilon_n}^{\varepsilon_n} \exp \left(\partial_{\theta=0}^2 \phi(\zeta e^{i\theta}) \frac{\theta^2}{2} + \partial_{\theta=0}^3 \phi(\zeta e^{i\theta}) \frac{\theta^3}{3!} + \dots \right) d\theta.$$

We now observe that, if we prove that all terms of order greater than 3 are negligible, and if we consider the variable change $u = \sqrt{|\partial_{\theta=0}^2 \phi(\zeta e^{i\theta})|} \theta$, then the integral converges to a Gaussian, under the condition that the integration bounds tend to infinity. Our goal is therefore to define $\varepsilon_n = n^{-\beta}$ so that the following two conditions are verified:

- **Neglect higher-order terms:** This ensures that, in the central region, the function ϕ may be replaced by its second-order Taylor approximation.
- **Capture the full Gaussian mass:** This requires that, after the change of variable, the integration bounds tend to infinity so that the truncated integral may be replaced by the full Gaussian integral.

Regarding the first condition, as mentioned previously, in principle one should verify that every term in the Taylor expansion is negligible. In practice it will be sufficient here to check that the cubic term vanishes asymptotically, that is,

$$\left| \partial_{\theta=0}^3 \phi(\zeta e^{i\theta}) \right| \varepsilon_n^3 \longrightarrow 0, \quad (2.1)$$

and to verify afterwards that this criterion was indeed sufficient.

For the second point, setting $u = \sqrt{|\partial_{\theta=0}^2 \phi(\zeta e^{i\theta})|} \theta$, changes the bounds to $\pm \sqrt{|\partial_{\theta=0}^2 \phi(\zeta e^{i\theta})|} \varepsilon_n$, and so it implies the condition

$$\sqrt{|\partial_{\theta=0}^2 \phi(\zeta e^{i\theta})|} \varepsilon_n \longrightarrow +\infty. \quad (2.2)$$

Let us then express the conditions 2.1 and 2.2 as conditions on β . Expliciting the derivatives of ϕ we have

$$\partial_{\theta=0}^2 \phi(\zeta e^{i\theta}) = -\frac{\zeta(1+\zeta)}{(1-\zeta)^3} \quad \text{and} \quad \partial_{\theta=0}^3 \phi(\zeta e^{i\theta}) = -i\zeta \frac{1+4\zeta+\zeta^2}{(1-\zeta)}.$$

Using the approximation $\zeta = 1 - \frac{1}{\sqrt{n}} + \frac{1}{2n} + \mathcal{O}(n^{-3/2})$, we see that

$$\zeta = \Theta(1), \quad 1 + \zeta = \Theta(1),$$

and

$$1 - \zeta = \Theta(n^{-1/2}), \quad \text{so} \quad (1 - \zeta)^3 = \Theta(n^{-3/2}).$$

Hence

$$\partial_{\theta=0}^2 \phi(\zeta e^{i\theta}) = \Theta(n^{3/2}).$$

Similar calculations yield

$$\partial_{\theta=0}^3 \phi(\zeta e^{i\theta}) = \Theta(n^2).$$

Thus, the conditions (2.1) and (2.2) become

$$n^{3/4} n^{-\beta} \rightarrow \infty \quad \text{and} \quad n^2 n^{-3\beta} \rightarrow 0.$$

In other terms

$$\frac{3}{4} - \beta > 0 \quad \text{and} \quad 2 - 3\beta < 0.$$

Therefore, in the end, β must satisfy

$$\frac{2}{3} < \beta < \frac{3}{4}.$$

For instance, $\beta = \frac{7}{10}$ is a suitable choice.

Now that β has been chosen using the heuristic condition, we need to actually perform the approximation and check that everything is correct. We apply Taylor's theorem in its integral form:

$$\phi(\zeta e^{i\theta}) = \phi(\zeta) + \partial_{t=0}^2 \phi(\zeta e^{it}) \frac{\theta^2}{2} + \int_0^\theta \partial_t^3 \phi(\zeta e^{it}) \frac{(\theta-t)^2}{2!} dt,$$

which rewrites

$$\phi(\zeta e^{i\theta}) = \phi(\zeta) - ((z\partial_z)^2\phi(z))_{z=\zeta} \frac{\theta^2}{2} - i \int_0^\theta ((z\partial_z)^3\phi(z))_{z=\zeta e^{it}} \frac{(\theta-t)^2}{2!} dt.$$

For $|t| \leq \varepsilon_n$, we have

$$\left| \partial_t^3 \phi(\zeta e^{it}) \right| = \left| z \frac{1+4z+z^2}{(1-z)^4} \right|_{z=\zeta e^{it}} \leq \frac{6}{|1-\zeta e^{it}|^4} \leq \frac{6}{|1-\zeta + \mathcal{O}(t)|^4} \leq \frac{6}{(1-\zeta)^4} \cdot \frac{1}{\left(1 + \mathcal{O}\left(\frac{\varepsilon_n}{1-\zeta}\right)\right)^4}$$

With our choice $\varepsilon_n = n^{-7/10}$ and $\zeta = 1 - \frac{1}{\sqrt{n}} + \mathcal{O}(n^{-1})$, we see that

$$\frac{\varepsilon_n}{1-\zeta} \rightarrow 0, \quad \text{hence} \quad \left| \partial_t^3 \phi(\zeta e^{it}) \right| = \mathcal{O}(n^2).$$

Using the triangle inequality, when $|\theta| \leq \varepsilon_n$ we have

$$\left| \int_0^\theta \partial_t^3 \phi(\zeta e^{it}) \frac{(\theta-t)^2}{2} dt \right| \leq \int_0^\theta \left| \partial_t^3 \phi(\zeta e^{it}) \frac{(\theta-t)^2}{2} \right| dt = \mathcal{O}(\varepsilon_n^3 n^2).$$

With our choice of β , we know that $\varepsilon_n^3 n^2$ tends to zero, therefore

$$\phi(\zeta e^{i\theta}) = \phi(\zeta) - ((z\partial_z)^2\phi(z))_{z=\zeta} \frac{\theta^2}{2} + o(1),$$

where the small o holds uniformly for θ in $[-\varepsilon_n, \varepsilon_n]$.

Thus, we get

$$\begin{aligned} \frac{1}{2\pi} \int_{-\varepsilon_n}^{\varepsilon_n} \exp\left(\phi(\zeta e^{i\theta})\right) d\theta &= \frac{1}{2\pi} \int_{-\varepsilon_n}^{\varepsilon_n} \exp\left(\phi(\zeta) - ((z\partial_z)^2\phi(z))_{z=\zeta} \frac{\theta^2}{2} + o(1)\right) d\theta \\ &\sim \frac{\exp(\phi(\zeta))}{2\pi} \int_{-\varepsilon_n}^{\varepsilon_n} \exp\left(-((z\partial_z)^2\phi(z))_{z=\zeta} \frac{\theta^2}{2}\right) d\theta. \end{aligned}$$

We can now apply the change of variable

$$u = \sqrt{((z\partial_z)^2\phi(z))_{z=\zeta}} \theta \quad \text{so that} \quad d\theta = \frac{du}{\sqrt{((z\partial_z)^2\phi(z))_{z=\zeta}}}.$$

We get

$$\frac{\exp(\phi(\zeta))}{2\pi} \int_{-\varepsilon_n}^{\varepsilon_n} \exp\left(-((z \partial_z)^2 \phi(z))_{z=\zeta} \frac{\theta^2}{2}\right) d\theta = \frac{\exp(\phi(\zeta))}{2\pi \sqrt{((z \partial_z)^2 \phi(z))_{z=\zeta}}} \int e^{-u^2/2} du,$$

where the bounds on the integration domain are

$$\sqrt{((z \partial_z)^2 \phi(z))_{z=\zeta}} \varepsilon_n.$$

Using $((z \partial_z)^2 \phi(z))_{z=\zeta} = \Theta(n^{3/2})$ and $\varepsilon_n = n^{-7/10}$, straightforward calculations give

$$\sqrt{((z \partial_z)^2 \phi(z))_{z=\zeta}} \varepsilon_n = \Theta(n^{1/20}),$$

which tends to infinity. Thus the integral over the whole real line is recovered and converges to $\sqrt{2\pi}$. Hence

$$\frac{1}{2\pi} \int_{-\varepsilon_n}^{\varepsilon_n} \exp(\phi(\zeta e^{i\theta})) d\theta \sim \frac{\exp(\phi(\zeta))}{2\pi} \int_{-\infty}^{\infty} e^{-u^2/2} du = \frac{\exp(\phi(\zeta))}{2\pi} \sqrt{2\pi} \frac{1}{\sqrt{((z \partial_z)^2 \phi(z))_{z=\zeta}}}.$$

So finally, the central part is asymptotically equivalent to

$$\frac{G(\zeta)}{\zeta^n \sqrt{2\pi ((z \partial_z)^2 \phi(z))_{z=\zeta}}}.$$

At this stage, this is a standard result for functions $G(z)$ belonging to a certain class called *Hayman-admissible functions*. Indeed, it coincides with the formula given in [FS09, Theorem VIII.4], where the notation differs slightly: our $((z \partial_z)^2 \phi(z))_{z=\zeta}$ corresponds to their $b(\zeta)$.

We now want to estimate this asymptotic in our particular setting. We already have

$$((z \partial_z)^2 \phi(z))_{z=\zeta} \sim 2n^{3/2}.$$

Now, replacing ζ by its asymptotic $1 - \frac{1}{\sqrt{n}}$, we get

$$G(\zeta) = \exp\left(\frac{\zeta}{1-\zeta}\right) = \exp\left(\frac{1 - \frac{1}{\sqrt{n}}}{\frac{1}{\sqrt{n}}}\right) = e^{-1} e^{\sqrt{n}},$$

and

$$\zeta^n = \left(1 - \frac{1}{\sqrt{n}}\right)^n = \exp\left(n \log\left(1 - \frac{1}{\sqrt{n}}\right)\right) \sim e^{-\sqrt{n}-1/2},$$

which yields

$$\frac{G(\zeta)}{\zeta^n} \sim e^{-1/2} e^{2\sqrt{n}}.$$

Therefore, the central part is equivalent to

$$\frac{e^{-1/2+2\sqrt{n}}}{2\sqrt{\pi n^{3/4}}}.$$

The final task is now to prove that the rest of the integral, namely the tail part, is negligible compared to the asymptotics of the central part.

Tails. The integral on the tail part is

$$\frac{1}{2\pi} \int_{\varepsilon_n}^{2\pi-\varepsilon_n} \exp(\phi(\zeta e^{i\theta})) d\theta.$$

It is bounded by

$$\frac{1}{2\pi} \int_{[-\pi, \pi] \setminus [-\varepsilon_n, \varepsilon_n]} |\exp(\phi(\zeta e^{i\theta}))| d\theta = \frac{1}{2\pi} \int_{[-\pi, \pi] \setminus [-\varepsilon_n, \varepsilon_n]} \exp(\operatorname{Re}(\phi(\zeta e^{i\theta}))) d\theta.$$

Expliciting the real part we have

$$\begin{aligned} \operatorname{Re}(\phi(\zeta e^{i\theta})) &= \operatorname{Re}\left(\frac{1}{1-\zeta e^{i\theta}}\right) - 1 - n \log(\zeta) = \operatorname{Re}\left(\frac{1-\zeta e^{-i\theta}}{|1-\zeta e^{i\theta}|^2}\right) - 1 - n \log(\zeta) \\ &= \frac{1-\zeta \cos \theta}{(1-\zeta \cos \theta)^2 + \zeta^2 \sin^2 \theta} - 1 - n \log(\zeta) \\ &= \frac{1-\zeta \cos \theta}{1+\zeta^2-2\zeta \cos \theta} - 1 - n \log(\zeta) \\ &= \frac{1}{2} \cdot \frac{1-\zeta^2}{1+\zeta^2-2\zeta \cos \theta} - \frac{1}{2} - n \log(\zeta). \end{aligned}$$

As a function of θ , this expression increases on $[-\pi, 0]$ and decreases on $[0, \pi]$. Therefore, its maximum on $[-\pi, \pi] \setminus [-\varepsilon_n, \varepsilon_n]$ is attained at either $-\varepsilon_n$ or ε_n . By the parity of the cosine function, these two values are in fact equal. Hence we have

$$\frac{1}{2\pi} \int_{[-\pi, \pi] \setminus [-\varepsilon_n, \varepsilon_n]} \exp(\operatorname{Re}(\phi(\zeta e^{i\theta}))) d\theta \leq \exp\left(\operatorname{Re}(\phi(\zeta e^{i\varepsilon_n}))\right).$$

Now, recall

$$\phi(\zeta e^{i\theta}) = \phi(\zeta) - ((z \partial_z)^2 \phi(z))_{z=\zeta} \frac{\theta^2}{2} + o(1),$$

which implies

$$\exp\left(\operatorname{Re}\left(\phi(\zeta e^{i\varepsilon_n})\right)\right) = \exp(\phi(\zeta)) \exp\left(-\left((z \partial_z)^2 \phi(z)\right)_{z=\zeta} \frac{\varepsilon_n^2}{2} + o(1)\right).$$

The term $\left((z \partial_z)^2 \phi(z)\right)_{z=\zeta} \varepsilon_n^2$ is of order $\Theta(n^{1/10})$. So the tail is bounded by $\exp(\phi(\zeta)) \exp(-\Theta(n^{1/10}))$, which is super-polynomially small relative to the main term $\exp(\phi(\zeta))$. So, the tail is negligible compared to the central part.

Final expression. We have obtained an asymptotic for the central part and shown that the tail part is negligible compared to it. Therefore,

$$\frac{g_n}{n!} \sim \frac{e^{-1/2+2\sqrt{n}}}{2\sqrt{\pi} n^{3/4}} \quad \text{as } n \rightarrow \infty. \quad (2.3)$$

We can go further by multiplying both sides by $n!$ and using Stirling's formula

$$n! \sim \sqrt{2\pi n} \left(\frac{n}{e}\right)^n.$$

This gives

$$g_n \sim \frac{1}{\sqrt{2}} n^{-1/4} \left(\frac{n}{e}\right)^n e^{-1/2+2\sqrt{n}} \quad \text{as } n \rightarrow \infty.$$

This asymptotic is rather crude and not particularly informative. Indeed, in Flajolet and Sedgewick [FS09], they do not even perform this calculation and simply retain Equation (2.3), which is easier to interpret. For instance, using Landau's notation, we have the compact formula

$$\frac{g_n}{n!} = \mathcal{O}(n^{-3/4} e^{2\sqrt{n}}). \quad (2.4)$$

We immediately see that the coefficients $\frac{g_n}{n!}$ grow to infinity exponentially fast, as the dominant growth is exponential in \sqrt{n} , while the polynomial factor $n^{-3/4}$ provides merely a slow correction. Consequently, Equation (2.3) provides a reliable approximation even for moderately large n , as the exponential term quickly dominates the behavior.

This concludes our introduction to analytic combinatorics. You are now ready to see the extension of all the methods presented here to the multivariate setting, as developed in the core of the thesis.

Enumeration of Graphs with Degree Constraints

3.1 Introduction

In 1785, Nicolas de Condorcet [Con85] introduced the notion of a *Condorcet winner*: a candidate who is preferred to each adversary by a majority of voters; in other words, a candidate who wins all their pairwise comparisons. However, such a candidate does not always exist. For instance, consider three candidates and three voters, with the following preference rankings:

$$1 > 2 > 3, \quad 2 > 3 > 1, \quad 3 > 1 > 2.$$

In this case, no candidate beats both of the others in pairwise comparisons. This phenomenon is known as the *Condorcet paradox*.

As mentioned in the introduction of the thesis, the Condorcet paradox is a major concept in social choice theory, evidenced by Gehrlein's book [Geh83], which is entirely devoted to studying the probability of its occurrence. It shows that collective preferences can be cyclic, even when all individual preferences are transitive and complete. This reveals a fundamental difficulty in aggregating individual rankings into a consistent group decision.

Given the central role of the Condorcet paradox in social choice theory, we were motivated to compute its probability with greater precision. Despite the large body of work on this topic, we identified room for new methodological approaches. This led us to explore a novel encoding of preference profiles into bipartite graphs with some degree constraints, allowing us to reformulate the problem in purely combinatorial terms. This encoding is detailed in Section 3.2.

The present chapter focuses on the enumeration of graphs with degree constraints. Although this approach yielded results for analyzing the Condorcet paradox that are only of limited interest, it led to new findings in graph enumeration that are of independent significance. In particular, we derive results on the enumeration of κ -regular graphs (where all vertices have degree κ), which are of combinatorial interest beyond the context of social choice. For readers less familiar with

analytic combinatorics, a concrete example is provided in Chapter 2 to illustrate the main concepts used in this chapter and the next two ones.

3.1.1 Related Work

Since this graph-based method was eventually set aside as a tool for studying the paradox itself, the related work discussed in this chapter will concern only the combinatorial and graph-theoretic aspects of our contribution. A full review of the literature on the probability of Condorcet paradox will be provided in the next chapter, where we introduce a more suitable methodology for addressing that question directly.

The enumeration of graphs with constrained degrees has a long history. For instance, in 1992, Gropp [Gro92] provided insights on work concerning this topic dating back over 100 years. The most famous graphs with degree constraints are κ -regular graphs, where all vertices have the same degree κ . There are two natural generalizations: graphs with a given degree sequence (see for instance [BC78]), and graphs where all vertices have their degree in a given set, which we consider in this chapter. We focus our review of the literature on the exact enumeration of graphs with a fixed number of vertices and constrained degrees, referring the reader interested in asymptotic results to the survey by Wormald [Wor18].

Before deriving an exact formula for an enumeration sequence, it is sometimes convenient to first establish that it satisfies *P-recursiveness*, either to gain insight into the generating function or because the problem of exact enumeration is too difficult to tackle directly, which was the case for graphs with degree constraints.

A sequence (a_n) is said to be *P-recursive* if it satisfies a linear recurrence with polynomial coefficients, for example

$$(n + 2)a_{n+2} - 3na_{n+1} + a_n = 0.$$

Equivalently, a generating function $A(z) = \sum_{n \geq 0} a_n z^n$ is said to be *D-finite* if it satisfies a linear differential equation with polynomial coefficients, for example

$$(1 - 3z)A''(z) + zA'(z) + A(z) = 0.$$

It is a classical result by Lipshitz [Lip89] that these two notions are equivalent: a sequence is *P-recursive* if and only if its generating function is *D-finite*.

Let $\text{SG}_n^{(\kappa)}$ denote the number of κ -regular graphs on n vertices. Read [Rea59b], applying results from [P6137], notably proved that $\text{SG}_n^{(3)}$ is D-finite and computed the corresponding differential equation. Read and Wormald [RW80] extended this result to $\text{SG}_n^{(4)}$ using a different technique. This approach consists in identifying a finite set of graph-like families containing a small number of loops and multiple edges, together with combinatorial operations transforming one family into another (such as removing an edge or a vertex). These operations translate into analytic relations between the corresponding generating functions, and solving the resulting system produces the differential equation for the generating function of $\text{SG}_n^{(4)}$. This technique has been applied in several subsequent works to enumerate graph families with constrained degrees under various connectivity assumptions (see for instance [CPR07]).

Regarding the explicit computation of generating functions, Read also contributed numerous works [Rea59a; Rea59b; Rea60]. His initial observation was that, letting the variable x_j mark the degree of vertex j for all $j \geq 1$, the generating function of all graphs (with arbitrary sizes) is

$$\prod_{1 \leq j < \ell} (1 + x_j x_\ell). \quad (3.1)$$

Indeed, each pair of vertices (j, ℓ) is either disjoint, contributing 1 to the generating function, or connected by an edge, which increases the degree of vertices j and ℓ by 1 and thus contributes the monomial $x_j x_\ell$. A graph is then a collection of such pairs. Taking the Cartesian product over all distinct pairs of vertices then corresponds to multiplying all factors $(1 + x_j x_\ell)$ for $1 \leq j < \ell$.

The number of κ -regular graphs on n vertices is then obtained by the coefficient extraction

$$[x_1^\kappa \cdots x_n^\kappa] \prod_{1 \leq j < \ell} (1 + x_j x_\ell).$$

This approach has the advantage of providing a generating function that is easily interpretable. Its main drawback, however, is that it requires an infinite number of variables, since graphs of arbitrary size are considered, and the coefficient corresponding to graphs of size n requires n variables. This makes it infeasible in practice to extract explicit numbers for graphs with degree constraints in the general case. This limitation motivates the search for alternative formulations that keep track of degree information while using only finitely many variables, at least when the maximum degree is bounded.

Going back to D-finiteness, starting from Equation (3.1), Goulden et al. [GJR83] and Goulden and Jackson [GJ86] used a method based on expanding a symmetric function in power sums to compute the differential equation of the generating functions of $\text{SG}_n^{(3)}$, $\text{SG}_n^{(4)}$ and of graphs with degrees

at most 4. They conjectured that for any finite set of allowed degrees, the generating function of graphs is D-finite. Gessel [Ges90] proved this conjecture by applying tools developed by Read [Rea60], rewriting the coefficient extraction in a symmetric series as a scalar product. Computer algebraic tools were developed by Mishna [Mis05] to turn the equations from [Ges90] into a program computing, for any given κ , the linear differential equation satisfied by the generating function of $\text{SG}_n^{(\kappa)}$. Those computations are costly and Chyzak and Mishna [CM24] recently obtained the differential equations only up to $\kappa = 7$. Finally, a new proof of the formula for $\text{SG}_n^{(3)}$ based on inclusion-exclusion was obtained by Chen and Louck [CL99]. For general values of κ , the differential equation satisfied by $\text{SG}_n^{(3)}$ is still unknown. We hope that our work, which provides a new expression for the generating function, is a step toward its discovery.

3.1.2 Contributions

This chapter is based on a joint work with Élie de Panafieu [CP23], where we consider both *multigraphs*, which allow multiple edges and loops, and *simple graphs*, which have neither multiple edges nor loops. In the literature, the term "graph" usually refers to simple graphs unless otherwise specified. In this chapter we adopt this convention, but when necessary, we explicitly distinguish between these two types to avoid confusion. Our main contributions are the following:

- We formalized the encoding of the Condorcet problem into a bipartite simple graph enumeration one.
- I derived an explicit formula for the generating function of bipartite simple graphs, where vertex degrees lie in prescribed sets—possibly distinct on the two sides. The approach relies on the symbolic method [BLL97; FS09] and on multivariate Hadamard products. I also proved that the generating function is D-finite.
- I expressed the probability of having a Condorcet winner using the generating function of bipartite simple graphs with degree constraints.
- I showed that the bipartite simple graph enumeration formula can also be applied to count balanced 0-1 matrices, highlighting the broader combinatorial interest of the method beyond the initial social choice motivation.
- Élie de Panafieu extended this construction to simple (non bipartite) graphs without loops, by adapting the bipartite framework to the general case with an adjustment for loop exclusion. This yields, in particular, a new expression for the generating function of κ -regular graphs, and its D-finiteness is also established.

- Building on this framework, Élie de Panafieu recently obtained the full asymptotic expansion of the number $SG_n^{(\kappa)}$ of κ -regular graphs [Pan24]:

$$SG_n^{(\kappa)} \sim \frac{(n\kappa/e)^{n\kappa/2}}{\kappa!^n} e^{-(\kappa^2-1)/4} \sqrt{2} \left(1 + c_1^{(\kappa)} n^{-1} + c_2^{(\kappa)} n^{-2} + \dots + \mathcal{O}(n^{-r}) \right),$$

for any r . Previously, only the dominant term was known [BC78], and the first few coefficients were conjectured in [McK83]. This asymptotic aspect is not detailed in this chapter.

3.1.3 Limitations

While the results presented in this chapter offer new insights and tools for the enumeration of constrained graphs, several limitations remain.

Combinatorics. Although we prove the D-finiteness of our generating functions for various classes of graphs, we do not derive the associated differential equations, nor do we offer an algorithmic method to extract coefficients. Moreover, no asymptotic estimates are obtained in this chapter, although the formulas are well suited to such analysis, as confirmed by Élie de Panafieu's independent work [Pan24].

Connection to voting theory. The application to social choice is currently limited to the *Impartial Culture* model, where all preference profiles are assumed equally likely. While this assumption is standard in the theoretical analysis of voting rules, it is a strong simplification of real-world scenarios. Furthermore, the resulting expression for the probability of the Condorcet paradox, although exact, remains intricate and currently resists simplification or asymptotic analysis.

Despite these limitations, the combinatorial framework developed here introduces a novel perspective on the problem and yields independently valuable results in enumerative graph theory.

3.1.4 Roadmap

This chapter is structured as follows. Section 3.2 introduces the encoding of the Condorcet winner problem as a graph enumeration problem. Section 3.3 introduces the notations and formal setup required for the computation of generating functions in the subsequent sections. The core results appear in Section 3.4, which addresses the enumeration of bipartite simple graphs and multigraphs. Section 3.5 presents how to apply the results on bipartite graphs to the calculation of the Condorcet

paradox probability, and to the enumeration of 0-1 matrices. Section 3.6, extends the computation of generating functions to non bipartite simple graphs and multigraphs. Section 3.7 concludes.

3.2 Encoding the Condorcet Problem Into Graph Enumeration

To determine whether a candidate is a *Condorcet winner* (i.e., a candidate who defeats every other one in pairwise majority comparisons), it is sufficient to record, for each voter, which opponents they rank above the candidate of interest. This naturally leads to encoding the problem as a bipartite graph that captures these pairwise defeats.

Figure 3.1 illustrates this process on a concrete example with four voters and four candidates. We focus on candidate 4 and highlight their position in each ranking. For each voter, we identify the candidates ranked above 4. This yields a 0-1 matrix where entry (i, j) equals 1 if voter i ranks candidate j above candidate 4, and 0 otherwise.

This matrix can be interpreted as the adjacency matrix of a bipartite graph: each left-hand vertex corresponds to the ranking ballot r_i casted by voter i , and the right-hand vertices correspond to the opponents of candidate 4. An edge between ranking r_i and candidate j indicates that j is ranked above 4 in voter i 's ranking. In this representation, candidate 4 is a Condorcet winner if and only if every right-hand vertex (i.e., opponent) has degree strictly less than $n/2$, where n is the number of voters. In our example, with four voters, this means degree strictly less than 2.

It is important to note that multiple voting profiles may correspond to the same 0-1 matrix, since only the pairwise comparisons involving candidate 4 matter. Figure 3.1 illustrates two distinct profiles yielding the same adjacency matrix. Therefore, in order to enumerate the voting profiles where a candidate, say candidate 4, is the Condorcet winner, it is not enough to count the bipartite graphs satisfying the appropriate degree constraints on the right vertices. One must also account for the multiplicity arising from the fact that a given bipartite graph corresponds to multiple profiles.

More precisely, consider the general case with m candidates and focus on the event where candidate m is the Condorcet winner. Suppose that a ranking places candidate m in position k . The number of full rankings that induce the same comparison pattern with respect to m is equal to the number of ways to permute the $k - 1$ candidates ranked above m , and the $m - k$ candidates ranked below. This yields a multiplicity factor of $(k - 1)!(m - k)!$. In the graph representation, a left-hand vertex of degree $k - 1$ corresponds to such a ranking and must therefore be associated with this

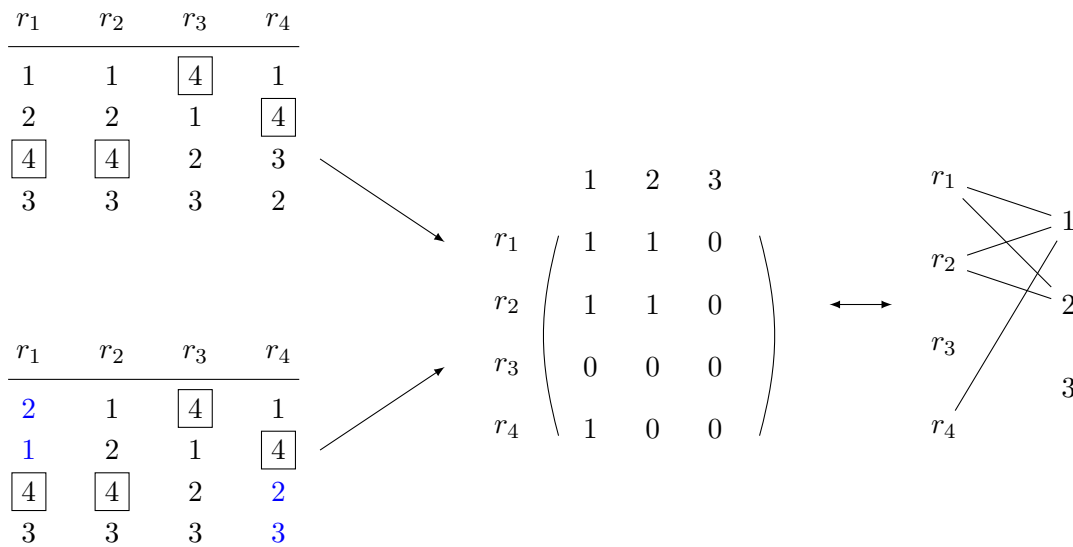


Fig. 3.1: Illustration of the transformation from a voting profile to a bipartite graph. Each column shows a voter's ranking with candidate 4 highlighted. In the center, the corresponding 0-1 matrix records each occurrence of a voter ranking a candidate above candidate 4. On the right, the bipartite simple graph encodes these pairwise preferences as edges between voters and adversaries. The two example profiles on the left differ slightly (e.g., swaps in r_1 and r_4 highlighted in blue in the bottom profile), yet yield the same graph, showing that multiple profiles may correspond to the same structure.

factor $(k - 1)!(m - k)!$. Then the bipartite graph is counted with a multiplicity that is equal to the product of these factors. This framework reduces the problem to enumerating bipartite graphs with degree constraints on the right vertices, and where the multiplicity depends solely on the degrees of the left vertices.

Thus, the probability that candidate m is a Condorcet winner corresponds to the ratio of the number of bipartite graphs in which all right-hand vertices have degree strictly less than $n/2$, to the total number of such graphs; in both cases, each graph is counted according to the multiplicity described above.

Both the multiplicity and the degree constraints can be accounted for by appropriately defining the generating function of bipartite graphs, which we do in Section 3.4. We postpone the explicit computation of probabilities to Section 3.5.1, as it becomes more straightforward once the generating functions are defined properly.

3.3 Notations and Conventions

Classic notations. We denote by \mathbf{x} the infinite vector (x_0, x_1, x_2, \dots) . This bold notation extends naturally to other indexed families of variables. The notation $\mathbf{1}$ stands for $(1, 1, \dots, 1)$, with length clarified by the context.

In Section 3.6, we use a double factorial, denoted by

$$(2k - 1)!! = \frac{(2k)!}{2^k k!}.$$

Given an integer vector $\mathbf{k} = (k_1, \dots, k_\ell)$, we define

$$\begin{aligned} \mathbf{k}! &= k_1! \cdots k_\ell!, \\ (2\mathbf{k} - 1)!! &= (2k_1 - 1)!! \cdots (2k_\ell - 1)!!, \\ \mathbf{w}^{\mathbf{k}} &= w_1^{k_1} \cdots w_\ell^{k_\ell}. \end{aligned} \tag{3.2}$$

We follow the definition of *ordinary* and *exponential* variables from [FS09]. Namely, in a generating function of the form

$$F(x, y) = \sum_{\ell, \mathbf{k}} f_{\ell, \mathbf{k}} x^\ell \frac{y^{\mathbf{k}}}{\mathbf{k}!},$$

we call x an *ordinary variable* and y an *exponential variable*. As explained in Chapter 2, exponential variables are used for labeled elements (such as vertices), and ordinary variables for unlabeled ones (such as degrees). Also, the n th coefficient of a formal power series is denoted by

$$[z^n] \sum_k a_k z^k = a_n \quad \text{or} \quad n! [z^n] \sum_k a_k \frac{z^k}{k!} = a_n.$$

This notation extends to the multivariate case

$$[z^n] \sum_{\mathbf{k}} a_{\mathbf{k}} z^{\mathbf{k}} = a_n \quad \text{or} \quad n! [z^n] \sum_{\mathbf{k}} a_{\mathbf{k}} \frac{z^{\mathbf{k}}}{\mathbf{k}!} = a_n.$$

Useful objects. Throughout Section 3.4, where we enumerate bipartite graphs, the variable δ_d is associated with left-hand vertices of degree d , and $\tilde{\delta}_d$ with right-hand vertices of degree d . We denote by $\boldsymbol{\delta}$ the infinite vector $(\delta_0, \delta_1, \delta_2, \dots)$, and similarly by $\tilde{\boldsymbol{\delta}}$ the vector $(\tilde{\delta}_0, \tilde{\delta}_1, \tilde{\delta}_2, \dots)$. When these variables are bounded by some D , we also write $\boldsymbol{\delta}$ for the finite vector $(\delta_0, \delta_1, \dots, \delta_D)$, and likewise for $\tilde{\boldsymbol{\delta}}$, with the size of the vector understood from the context.

The sequences δ and $\tilde{\delta}$ can be viewed as parameter functions tailored to the problem at hand. For instance, in the context developed above, setting

$$\delta_d = d!(\tilde{m} - d)! \quad \text{and} \quad \tilde{\delta}_d = \mathbb{1}_{d < n/2},$$

where n is the number of left-hand vertices and $\tilde{m} := m - 1$ is the number of right-hand vertices, precisely encodes the multiplicities and degree constraints discussed earlier. These functions will be incorporated into the generating function in a suitable way.

Throughout Section 3.6, where we enumerate "general" graphs, *i.e.* graphs that are not necessarily bipartite, the variable δ_d marks vertices of degree d . As in the bipartite case, a particularly useful specialization is obtained by fixing a set \mathcal{D} of nonnegative integers and evaluating $\delta_d = \mathbb{1}_{d \in \mathcal{D}}$. This restricts the count to graphs in which all vertex degrees lie in \mathcal{D} . In particular, choosing $\mathcal{D} = \{\kappa\}$ yields the generating function of κ -regular graphs.

Given an exponential variable y , we associate to y and δ the generating function

$$\Delta(y, \delta) = \sum_{d \geq 0} \delta_d \frac{y^d}{d!}.$$

Given variables (x_1, x_2, \dots) and y , we also define the series

$$H(y, \mathbf{x}) = \sum_{j \geq 1} x_j y^j.$$

The rationale behind introducing these generating functions stems from the step-by-step construction of graphs and the use of symbolic method. Its precise interpretation will be discussed when it first appears, in the proof of Lemma 1.

The *support* of a vector δ is the set of indices corresponding to nonzero values

$$\text{support}(\delta) = \{d \mid \delta_d \neq 0\}.$$

We say that δ has *finite support* when its support is a finite set. Observe that $\Delta(y, \delta)$ is then a polynomial.

Hadamard product. Throughout this chapter, we make extensive use of the *exponential Hadamard product*, that is, the coefficient-wise product of exponential generating functions. We refer the

reader to [Joy81, Theorem 3] and [BLL97, Section 2.1] for a broad treatment of the topic. It is denoted by and defined as

$$\sum_j a_j \frac{z^j}{j!} \odot_z \sum_j b_j \frac{z^j}{j!} = \sum_j a_j b_j \frac{z^j}{j!}.$$

We denote by $\odot_{z=u}$ the exponential Hadamard product followed by the evaluation at $z = u$

$$\sum_j a_j \frac{z^j}{j!} \odot_{z=u} \sum_j b_j \frac{z^j}{j!} = \sum_j a_j b_j \frac{u^j}{j!}.$$

The *multivariate exponential Hadamard product* is defined as

$$\sum_j a_j \frac{z^j}{j!} \odot_z \sum_j b_j \frac{z^j}{j!} = \sum_j a_j b_j \frac{z^j}{j!}.$$

This extension is compatible with the univariate product in the sense that

$$A(z_1)B(z_2) \odot_{(z_1, z_2)} C(z_1, z_2) = A(z_1) \odot_{z_1} (B(z_2) \odot_{z_2} C(z_1, z_2)).$$

In particular, we have

$$e^{x_1^2/2 + \dots + x_D^2/2} \odot_{\mathbf{x}} A(\mathbf{x}) = e^{x_1^2/2} \odot_{x_1} e^{x_2^2/2} \odot_{x_2} \dots e^{x_D^2/2} \odot_{x_D} A(\mathbf{x}).$$

3.4 Bipartite Graphs

In this section, we enumerate successively several bipartite graph-like families with degree constraints: *bipartite weighted multigraphs*, *bipartite weighted simple graphs*, and finally *bipartite simple graphs*. They are all depicted in Figure 3.2. The generating function of bipartite weighted multigraphs is expressed directly in Lemma 1. In Lemma 2, the generating function of bipartite weighted multigraphs is expressed using the generating function of bipartite weighted simple graphs, and this relation is then inverted. Finally, the generating function of bipartite simple graphs is obtained in Theorem 2 by restricting the weights to 1.

This approach is motivated by the fact that computing the generating function of multigraphs is straightforward. In particular, the symbolic method applies naturally when connecting edges, as there are no constraints beyond matching the total number of left and right endpoints. Moreover, transforming a bipartite simple graph into a bipartite weighted multigraph is intuitive, as illustrated

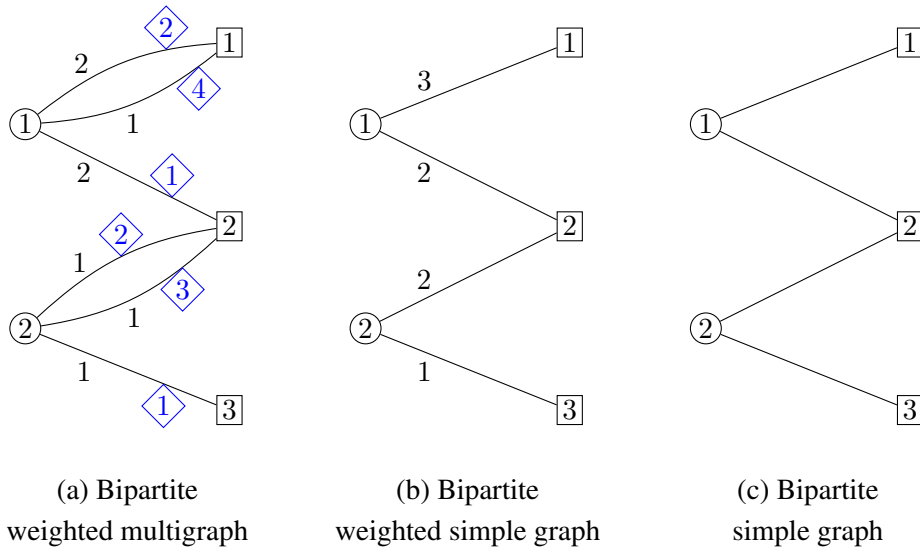


Fig. 3.2: Steps of the transformation of a bipartite weighted multigraph into a bipartite simple graph. The left-vertices are circled while the right-vertices are framed. The edges' labels are represented by blue integers inside diamond-shaped frames, while weights are represented by black integers.

in Figure 3.2: when viewing the graphs from right to left, one can see that it suffices to assign weights and replace each edge by multiple parallel edges, with the only condition that their total weight matches that of the original edge.

3.4.1 Bipartite Weighted Multigraphs

As mentioned above, our quest starts by expressing the generating function of bipartite weighted multigraphs with degree constraints.

Definition. A bipartite weighted multigraph G is a finite sequence

$$G = (V(G), \tilde{V}(G), E_1(G), E_2(G), \dots)$$

where $V(G)$ (resp. $\tilde{V}(G)$) is the set of n left-vertices (resp. \tilde{m} right-vertices), and for all $j \geq 1$, $E_j(G)$ is the finite set of k_j edges of weight j . The labels of the edges in $E_j(G)$ are $\{1, \dots, k_j\}$ and each edge is a triplet (u, v, ℓ) where u is a left-vertex, v is a right-vertex and ℓ is the label. Thus, all edges are undirected, and loops are not allowed. We define the degree of a vertex as the sum of the weights of its adjacent edges.

Note that the notations are not standard in graph theory, but we chose them to align with the Condorcet setting: there are n left vertices corresponding to n voters, and $\tilde{m} = m - 1$ representing the $m - 1$ adversaries.

Example. Let G denote the graph (a) from Figure 3.2. We have

$$\begin{aligned} V(G) &= \{1, 2\} \\ \tilde{V}(G) &= \{1, 2, 3\} \\ E_1(G) &= \{(2, 3, 1), (2, 2, 2), (2, 2, 3), (1, 1, 4)\} \\ E_2(G) &= \{(1, 2, 1), (1, 1, 2)\} \end{aligned}$$

The left-vertices 1 and 2 have degree $2 + 1 + 2 = 5$ and $1 + 1 + 1 = 3$ respectively. The right-vertices 1, 2 and 3 have degree $2 + 1 = 3$, $2 + 1 + 1 = 4$ and 1.

Generating function. We use the exponential variable z (resp. \tilde{z}) to mark the left-vertices (resp. right-vertices). For all $j \geq 1$, we use the exponential variable w_j to mark each edge of weight j . Additionally, we introduce δ and $\tilde{\delta}$, the vectors of variables associated respectively with the degrees of the left- and right-vertices.

The generating function of a bipartite weighted multigraphs family \mathcal{A} is then defined as

$$A(z, \tilde{z}, \mathbf{w}, \boldsymbol{\delta}, \tilde{\boldsymbol{\delta}}) = \sum_{G \in \mathcal{A}} \left(\prod_{u \in V(G)} \delta_{\deg(u)} \right) \left(\prod_{v \in \tilde{V}(G)} \tilde{\delta}_{\deg(v)} \right) \left(\prod_{j \geq 1} \frac{w_j^{k_j}}{k_j!} \right) \frac{z^n \tilde{z}^{\tilde{m}}}{n! \tilde{m}!}. \quad (3.3)$$

Note that since each bipartite weighted multigraph G has a finite number of edges, each product in the definition of $A(z, \tilde{z}, \mathbf{w}, \boldsymbol{\delta}, \tilde{\boldsymbol{\delta}})$ has finite support.

In particular, we note $\text{BWMG}(z, \tilde{z}, \mathbf{w}, \boldsymbol{\delta}, \tilde{\boldsymbol{\delta}})$ the generating function of all bipartite weighted multigraphs.

Lemma 1. Let $\Delta(y, \boldsymbol{\delta})$ and $H(y, \mathbf{x})$ denote the formal power series

$$\Delta(y, \boldsymbol{\delta}) = \sum_{d \geq 0} \delta_d \frac{y^d}{d!}, \quad H(y, \mathbf{x}) = \sum_{j \geq 1} x_j y^j.$$

And let $P_{\text{WMG}}(\mathbf{x}, \boldsymbol{\delta})$, denote the formal power series

$$P_{\text{WMG}}(\mathbf{x}, \boldsymbol{\delta}) = \Delta(y, \boldsymbol{\delta}) \odot_{y=1} e^{H(y, \mathbf{x})}.$$

Then, the generating function of bipartite weighted multigraphs is equal to

$$\text{BWMG}(z, \tilde{z}, \boldsymbol{\omega}, \boldsymbol{\delta}, \tilde{\boldsymbol{\delta}}) = e^{z P_{\text{WMG}}(\mathbf{w}, \boldsymbol{\delta})} \odot_{\mathbf{w}} e^{\tilde{z} P_{\text{WMG}}(\mathbf{w}, \tilde{\boldsymbol{\delta}})}.$$

When $\boldsymbol{\delta}$ and $\tilde{\boldsymbol{\delta}}$ are vectors of real values with $\delta_d = 0$ and $\tilde{\delta}_d = 0$ for $d > D$, $P_{\text{WMG}}(\mathbf{w}, \boldsymbol{\delta})$ and $P_{\text{WMG}}(\mathbf{w}, \tilde{\boldsymbol{\delta}})$ become polynomials and

$$\text{BWMG}(z, \tilde{z}, (w_1, \dots, w_D), \boldsymbol{\delta}, \tilde{\boldsymbol{\delta}}) = e^{z P_{\text{WMG}}(w_1, \dots, w_D, \boldsymbol{\delta})} \odot_{\mathbf{w}} e^{\tilde{z} P_{\text{WMG}}(w_1, \dots, w_D, \tilde{\boldsymbol{\delta}})}.$$

Remark. $P_{\text{WMG}}(\mathbf{x}, \boldsymbol{\delta})$ is a generating function that retains meaning even beyond the context of bipartite graphs, hence its name. We will see in the proof exactly what this generating function enumerates.

D-finiteness. Observe that the function $e^{z P_{\text{WMG}}(w_1, \dots, w_D, \boldsymbol{\delta})}$ is D-finite with respect to the variables (w_1, \dots, w_D) , since it is the exponential of a polynomial in these variables. An identical argument applies to $e^{\tilde{z} P_{\text{WMG}}(w_1, \dots, w_D, \tilde{\boldsymbol{\delta}})}$, which is likewise D-finite. Moreover, it is known that D-finite power series are closed under the multivariate Hadamard product (see [Lip89, Theorem 2.7]). It follows that the generating function $\text{BWMG}(z, \tilde{z}, (w_1, \dots, w_D), \boldsymbol{\delta}, \tilde{\boldsymbol{\delta}})$, expressed as the multivariate Hadamard exponential product of the two functions above, is itself D-finite.

Proof. The proof essentially relies on the use of the symbolic method. The process is illustrated by Figure 3.3.

Consider a bipartite weighted multigraph and let us replace each edge (u, v, ℓ) of weight j and label ℓ with two half-edges of weight j and label ℓ , one attached to the vertex u , the other to the vertex v . This yields a unique decomposition of the bipartite weighted multigraph into a set of labeled vertices, each equipped with a set of labeled, weighted half-edges. The vertex set is partitioned into two subsets: the left vertices, to which are attached **left** half-edges, and the right vertices, to which are attached **right** half-edges. Accordingly we introduce, for each j , an exponential variable u_j (resp. v_j) marking left half-edges (resp. right half-edges) of weight j .

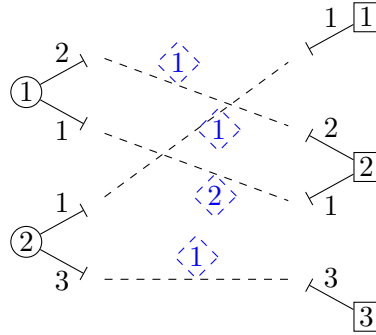


Fig. 3.3: Illustration of the decomposition used in the proof of Lemma 1, on a bipartite weighted multigraph with $\mathbf{k} = (2,1,1)$. Each edge is replaced by two half-edges, one on each side, so that left and right vertex contributions can be treated separately in the generating function.

Now we introduce an ordinary variable y marking the degree of a vertex. With those conventions, the generating function of one left half-edge is

$$\sum_{j \geq 1} u_j y^j = H(y, \mathbf{u}).$$

The generating function of a set of left half-edges is then $e^{H(y, \mathbf{u})}$. We want to replace, for all $d \geq 1$, each y^d with δ_d . This operation corresponds to an ordinary Hadamard product of $e^{H(y, \mathbf{u})}$ with $\sum_{d \geq 0} \delta_d y^d$ with respect to y . We write it instead as an exponential Hadamard product with $\sum_{d \geq 0} \delta_d \frac{y^d}{d!} = \Delta(y, \boldsymbol{\delta})$

$$e^{H(y, \mathbf{u})} \odot_{y=1} \Delta(y, \boldsymbol{\delta})$$

and recognize the expression of $P_{\text{WMG}}(\mathbf{u}, \boldsymbol{\delta})$. Thus, $P_{\text{WMG}}(\mathbf{u}, \boldsymbol{\delta})$ denotes here the generating function of a set of left half-edges. Note that it can also be used in a non bipartite context to denote a set of half-edges, if \mathbf{u} represents half-edges in general, which explains the chosen notation. The generating function of a set of left vertices, each attached to a set of labeled left half-edges, is then

$$e^{z P_{\text{WMG}}(\mathbf{u}, \boldsymbol{\delta})}.$$

Similarly, The generating function of a set of right vertices, each attached to a set of labeled right half-edges, is

$$e^{\tilde{z} P_{\text{WMG}}(\mathbf{v}, \tilde{\boldsymbol{\delta}})}.$$

Now, for each weight j , the number of left and right half-edges of weight j must be equal. This constraint is enforced by replacing for every j the variables u_j and v_j by a common exponential

variable w_j , and by performing an exponential Hadamard product between the left and right generating functions with respect to \mathbf{w}

$$\text{BWMG}(z, \tilde{z}, \boldsymbol{\omega}, \boldsymbol{\delta}, \tilde{\boldsymbol{\delta}}) = e^{zP_{\text{WGM}}(\mathbf{w}, \boldsymbol{\delta})} \odot_{\mathbf{w}} e^{\tilde{z}P_{\text{WGM}}(\mathbf{w}, \tilde{\boldsymbol{\delta}})}. \quad \square$$

Now that we have found the generating function BWMG of bipartite weighted multigraphs, we want to express the one of bipartite simple graphs as a function of BWMG.

3.4.2 Bipartite Simple Graphs

To get the generating function of bipartite simple graphs, we first find the generating function of bipartite weighted simple graphs, and then restrict it to the case where all weights equal 1.

Definition. A bipartite weighted simple graph is a bipartite weighted multigraph without multiple edge and without label on the edges. So, it is a finite sequence

$$G = (V(G), \tilde{V}(G), E_1(G), E_2(G), \dots)$$

where $V(G) = \{1, 2, \dots, n\}$ is the set of n left vertices, $\tilde{V}(G) = \{1, 2, \dots, \tilde{m}\}$ is the set of \tilde{m} right vertices, and $\bigcup_j E_j(G)$ is a finite set of edges, which are distinct unordered pairs $\{u, v\}$ of vertices, where $u \in V(G)$ and $v \in \tilde{V}(G)$.

Bipartite simple graphs correspond to the case $G = (V(G), \tilde{V}(G), E_1(G))$, so $E_j = \emptyset$ for all $j \geq 2$.

Generating function. The generating function of a bipartite weighted simple graph family \mathcal{B} is defined as

$$B(z, \tilde{z}, \mathbf{w}, \boldsymbol{\delta}, \tilde{\boldsymbol{\delta}}) = \sum_{G \in \mathcal{B}} \left(\prod_{u \in V(G)} \delta_{\deg(u)} \right) \left(\prod_{v \in \tilde{V}(G)} \tilde{\delta}_{\deg(v)} \right) \left(\prod_{j \geq 1} w_j^{k_j} \right) \frac{z^n}{n!} \frac{\tilde{z}^{\tilde{m}}}{\tilde{m}!}. \quad (3.4)$$

The difference with Equation (3.3) is that because edges are unordered, every w_j is now an ordinary variable, hence it is not divided by $k_j!$.

In particular, we denote by $\text{BWSG}(z, \tilde{z}, \mathbf{w}, \boldsymbol{\delta}, \tilde{\boldsymbol{\delta}})$ the generating function of all bipartite weighted simple graphs, and by $\text{BSG}(z, \tilde{z}, \mathbf{w}, \boldsymbol{\delta}, \tilde{\boldsymbol{\delta}})$ the generating function of all bipartite simple graphs.

Condorcet winner. Now that we have introduced the generating function BSG, we can formally express the probability that candidate m is a Condorcet winner. To do so, we recall how to define δ and $\tilde{\delta}$ for encoding the Condorcet probability. On the left side (voters), we set $\delta_d = d!(\tilde{m} - d)!$ to ensure that each graph is counted with the correct multiplicity. On the right side (adversaries), we set $\tilde{\delta}_d = \mathbb{1}_{d < n/2}$, which encodes the condition that candidate m must beat each other candidate in a strict majority of votes—this being the definition of a Condorcet winner.

Under the *impartial culture* assumption, all rankings are equally likely and independently drawn by each of the n voters. In this setting, every voting profile has the same probability. The generating function $\text{BSG}(z, \tilde{z}, w, \delta, \tilde{\delta})$ is thus well-suited to this model, as the role of δ is to ensure that each bipartite graph is counted with the correct multiplicity, corresponding to the number of ranking profiles it represents. The desired probability is then obtained by taking the ratio of the number of Condorcet-favorable graphs to the total number of graphs

$$\mathbb{P}(m \text{ is CW}) = \frac{[z^n \tilde{z}^{\tilde{m}}] \sum_k [w^k] \text{BSG}(z, \tilde{z}, w, \delta, \tilde{\delta})}{[z^n \tilde{z}^{\tilde{m}}] \sum_k [w^k] \text{BSG}(z, \tilde{z}, w, \delta, \mathbf{1})}.$$

Under the impartial culture assumption, all candidates are symmetric, so the probability that any given candidate is a Condorcet winner does not depend on m . Summing over all candidates then gives the probability that a Condorcet winner exists

$$\mathbb{P}(\text{there exists a CW}) = m \frac{[z^n \tilde{z}^{\tilde{m}}] \sum_k [w^k] \text{BSG}(z, \tilde{z}, w, \delta, \tilde{\delta})}{[z^n \tilde{z}^{\tilde{m}}] \sum_k [w^k] \text{BSG}(z, \tilde{z}, w, \delta, \mathbf{1})}.$$

Having seen how BSG can be used to express the probability that a Condorcet winner exists, we now return to the combinatorial analysis. In particular, we have not yet made the expression of BSG explicit. The next lemma provides a closed-form expression for BWSG, from which BSG will directly follow.

Lemma 2. Recall that $v_j(\mathbf{w})$ denotes the formal power series

$$v_j(\mathbf{w}) = [y^j] \log(1 + H(y, \mathbf{w})).$$

Then, the generating function of bipartite weighted simple graphs is equal to

$$\text{BWSG}(z, \tilde{z}, \mathbf{w}, \delta, \tilde{\delta}) = e^{z P_{\text{WMG}}(\mathbf{u}, \delta)} \odot_{\mathbf{u}=(v_j(\mathbf{w}))_{j \geq 1}} e^{\tilde{z} P_{\text{WMG}}(\mathbf{u}, \tilde{\delta})}.$$

Proof. We associate to any bipartite weighted simple graph G the following family $M(G)$ of bipartite weighted multigraphs, illustrated on a example graph in Figure 3.4.

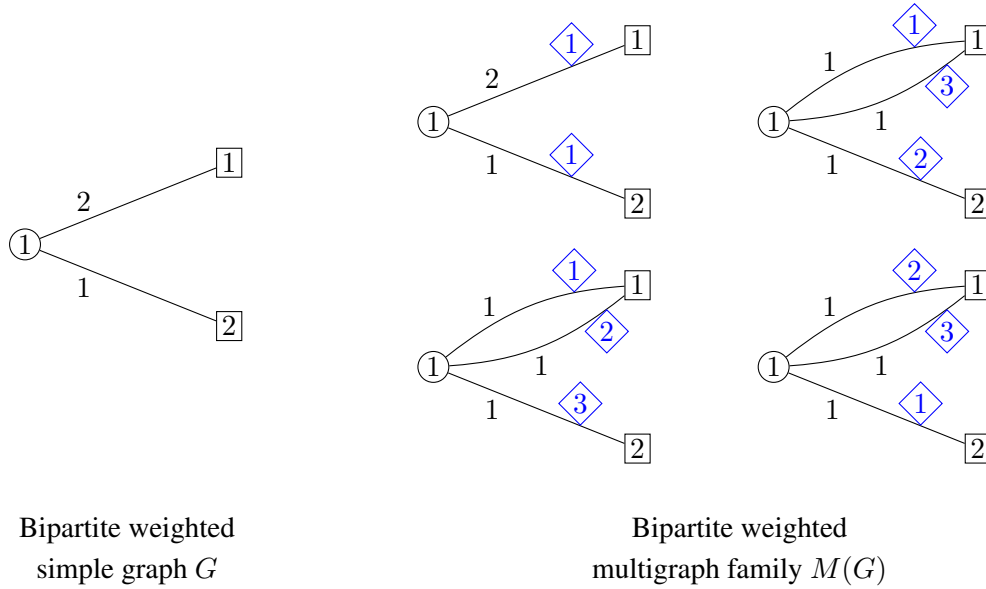


Fig. 3.4: Left, a bipartite weighted simple graph G . Right, the four bipartite weighted multigraphs of the family $M(G)$ defined in the proof of Lemma 2. The labels of the left vertices are circled, the labels of the right vertices are in squares, the labels of the edges are framed in blue diamond shapes, the weights of the edges are the non-circled numbers.

$M(G)$ contains all bipartite multigraphs such that for any left vertex u and right vertex v , the sum of the weights of the edges between u and v is equal to the weight of the edge $\{u,v\}$ in G . In particular, if $\{u,v\}$ is not an edge of G , there is no edge of any weight between u and v in any bipartite multigraph from $M(G)$. Thus, one way to construct all bipartite multigraphs from $M(G)$ is to start with the bipartite simple graph G and replace each edge $\{u,v\}$ of weight j with a set of labeled weighted edges whose weights sum to j . As usual, we use a different label set for each weight. There might be many different ways to decompose one weight j into a set of weights summing to j and to label the edges. Each gives rise to a different bipartite multigraph from $M(G)$.

Let v_j denote the exponential variable marking labeled edges of weight j in bipartite weighted multigraphs. The exponential generating function of nonempty sets of labeled edges (one label type for each weight) whose weights is marked by the ordinary variable y is

$$e^{H(y,v)} - 1.$$

For example, the number of such sets with sum of the weights j , containing k_2 edges of weight 2 and k_5 edges of weight 5, but not other edge, is

$$k_2!k_5![y^j v_2^{k_2} v_5^{k_5}](e^{H(y,v)} - 1).$$

Of course, this is 0 when $2k_2 + 5k_5 \neq j$. For any $j \geq 0$,

$$w_j(\mathbf{v}) = [y^j](e^{H(y,v)} - 1) \quad (3.5)$$

is the generating function of nonempty sets of labeled edges whose weights sum to j , with the convention $w_0(\mathbf{v}) = 0$.

Let us go back to our bipartite weighted simple graph G . For all d , let n_d (resp. \tilde{m}_d) denote its number of left (resp. right) vertices of degree d , let n (resp. \tilde{m}) denote the total number of left (resp. right) vertices, and for any $j \geq 1$, let k_j be the number of edges of weight j . Then the generating function of $M(G)$, following (3.3), is

$$\left(\prod_{j \geq 1} w_j(\mathbf{v})^{k_j} \right) \left(\prod_{d \geq 0} \delta_d^{n_d} \right) \left(\prod_{d \geq 0} \tilde{\delta}_d^{\tilde{m}_d} \right) \frac{z^n}{n!} \frac{\tilde{z}^{\tilde{m}}}{\tilde{m}!}.$$

Using the variable w_j to mark edges of weight j in bipartite weighted simple graphs, the generating function of G (following (3.4)) is

$$\left(\prod_{j \geq 1} w_j^{k_j} \right) \left(\prod_{d \geq 0} \delta_d^{n_d} \right) \left(\prod_{d \geq 0} \tilde{\delta}_d^{\tilde{m}_d} \right) \frac{z^n}{n!} \frac{\tilde{z}^{\tilde{m}}}{\tilde{m}!}.$$

We observe that the generating function of $M(G)$ is obtained from the generating function of G by replacing, for all $j \geq 1$, the variable w_j with $w_j(\mathbf{v})$. By summation, we conclude that the generating functions of bipartite weighted simple graphs and bipartite weighted multigraphs are linked by the relation

$$\text{BWSG}(z, \tilde{z}, (w_j(\mathbf{v}))_{j \geq 1}, \delta, \tilde{\delta}) = \text{BWMG}(z, \tilde{z}, \mathbf{v}, \delta, \tilde{\delta}).$$

Equation (3.5) expresses w_j as a function of \mathbf{v} for all $j \geq 0$. Let us now invert it to express v_j as a function of \mathbf{w} for all $j \geq 0$. We have

$$H(y, \mathbf{w}(\mathbf{v})) = \sum_{d \geq 0} w_d(\mathbf{v}) y^d = \exp \left(\sum_{j \geq 1} v_j y^j \right) - 1,$$

a straightforward calculation gives $v_j = [y^j] \log(1 + H(y, \mathbf{w}(\mathbf{v})))$. We define

$$v_j(\mathbf{w}) = [y^j] \log(1 + H(y, \mathbf{w}))$$

so

$$\text{BWMG}(z, \tilde{z}, (v_j(\mathbf{w}))_{j \geq 1}, \boldsymbol{\delta}, \tilde{\boldsymbol{\delta}}) = \text{BWSG}(z, \tilde{z}, \mathbf{w}, \boldsymbol{\delta}, \tilde{\boldsymbol{\delta}}).$$

Now, recall that

$$\text{BWMG}(z, \tilde{z}, \boldsymbol{\omega}, \boldsymbol{\delta}, \tilde{\boldsymbol{\delta}}) = e^{zP_{\text{WMG}}(\boldsymbol{\omega}, \boldsymbol{\delta})} \odot_{\boldsymbol{\omega}} e^{\tilde{z}P_{\text{WMG}}(\boldsymbol{\omega}, \tilde{\boldsymbol{\delta}})}.$$

Substituting \mathbf{w} with the vector $(v_j(\mathbf{w}))_{j \geq 1}$ in BWMG amounts to evaluating the Hadamard product at $(v_j(\mathbf{w}))_{j \geq 1}$, as the vector \mathbf{w} plays the role of a dummy variable in both components of the product. As a result, we obtain

$$\text{BWSG}(z, \tilde{z}, \mathbf{w}, \boldsymbol{\delta}, \tilde{\boldsymbol{\delta}}) = e^{zP_{\text{WMG}}(\mathbf{u}, \boldsymbol{\delta})} \odot_{\mathbf{u}=(v_j(\mathbf{w}))_{j \geq 1}} e^{\tilde{z}P_{\text{WMG}}(\mathbf{u}, \tilde{\boldsymbol{\delta}})}. \quad \square$$

With all the preliminary components in place, we are now ready to derive a closed-form expression for the generating function of bipartite simple graphs with degree constraints.

Theorem 2. Let $\Delta(y, \boldsymbol{\delta})$ and $H(y, \mathbf{x})$ denote the formal power series

$$\Delta(y, \boldsymbol{\delta}) = \sum_{d \geq 0} \delta_d \frac{y^d}{d!}, \quad H(y, \mathbf{x}) = \sum_{j \geq 1} x_j y^j.$$

Let P_{WMG} denote the formal power series

$$P_{\text{WMG}}(\mathbf{x}, \boldsymbol{\delta}) = \Delta(y, \boldsymbol{\delta}) \odot_{y=1} e^{H(y, \mathbf{x})}.$$

Define, for all $j \geq 1$

$$h_j(w) = -\frac{(-w)^j}{j}.$$

Then the generating function of bipartite simple graphs is equal to

$$\text{BSG}(z, \tilde{z}, w, \boldsymbol{\delta}, \tilde{\boldsymbol{\delta}}) = e^{zP_{\text{WMG}}(\mathbf{u}, \boldsymbol{\delta})} \odot_{\mathbf{u}=(h_j(w))_{j \geq 1}} e^{\tilde{z}P_{\text{WMG}}(\mathbf{u}, \tilde{\boldsymbol{\delta}})}.$$

When $\delta_d = 0$ and $\tilde{\delta}_d = 0$ for all $d > D$, $P_{\text{WMG}}(\mathbf{u}, \boldsymbol{\delta})$ becomes a polynomial in (u_1, \dots, u_D) and

$$\text{BSG}(z, \tilde{z}, w, \boldsymbol{\delta}, \tilde{\boldsymbol{\delta}}) = e^{zP_{\text{WMG}}(\mathbf{u}, \boldsymbol{\delta})} \odot_{\mathbf{u}=(h_j(w))_{1 \leq j \leq D}} e^{\tilde{z}P_{\text{WMG}}(\mathbf{u}, \tilde{\boldsymbol{\delta}})}.$$

D-finiteness. Note that the same reasoning used to establish the D-finiteness of BWMG applies to BSG. The only difference is that the Hadamard product involved is univariate rather than multivariate, which still preserves D-finiteness as a special case of the result in [Lip89].

Proof. Remember that a bipartite simple graph is a bipartite weighted simple graph with every edge of weight 1. Therefore, we have the following equality

$$\text{BSG}(z, \tilde{z}, w, \boldsymbol{\delta}, \tilde{\boldsymbol{\delta}}) = \text{BWSG}(z, \tilde{z}, (w, 0, 0, \dots), \boldsymbol{\delta}, \tilde{\boldsymbol{\delta}}).$$

Now, using Lemma 2

$$\text{BWSG}(z, \tilde{z}, \mathbf{w}, \boldsymbol{\delta}, \tilde{\boldsymbol{\delta}}) = e^{zP_{\text{WMG}}(\mathbf{u}, \boldsymbol{\delta})} \odot_{\mathbf{u}=(v_j(\mathbf{w}))_{j \geq 1}} e^{\tilde{z}P_{\text{WMG}}(\mathbf{u}, \tilde{\boldsymbol{\delta}})},$$

and replacing \mathbf{w} with $(w, 0, \dots)$

$$v_j(\mathbf{w}) = [y^j] \log(1 + wy) = -\frac{(-w)^j}{j} = h_j(w),$$

gives

$$\text{BSG}(z, \tilde{z}, w, \boldsymbol{\delta}, \tilde{\boldsymbol{\delta}}) = e^{zP_{\text{WMG}}(\mathbf{u}, \boldsymbol{\delta})} \odot_{\mathbf{u}=(h_j(w))_{j \geq 1}} e^{\tilde{z}P_{\text{WMG}}(\mathbf{u}, \tilde{\boldsymbol{\delta}})}. \quad \square$$

3.5 Applications

This section first details the original motivating application: computing the probability that a Condorcet winner exists under the impartial culture assumption. Then, we demonstrate the versatility of our framework by applying the same formulas to the enumeration of 0-1 balanced matrices, illustrating its relevance beyond the context of social choice.

3.5.1 Condorcet Winner

Having obtained a closed-form expression for the generating function of bipartite simple graphs with degree constraints, we now turn to its application to the existence of a Condorcet winner. This is done by specializing the degree functions $\boldsymbol{\delta}$ and $\tilde{\boldsymbol{\delta}}$ to encode the Condorcet winning condition.

First, with this explicit formula at hand, the probability that a Condorcet winner exists under the impartial culture assumption can be rewritten as

$$\mathbb{P}(\text{there exists a CW}) = m \cdot \frac{[z^n \tilde{z}^{\tilde{m}}] \sum_{k \leq n \cdot \tilde{m}} [w^k] e^{z P_{\text{WGM}}(\mathbf{u}, \boldsymbol{\delta})} \odot_{\mathbf{u}=(h_j(w))_{1 \leq j \leq k}} e^{\tilde{z} P_{\text{WGM}}(\mathbf{u}, \tilde{\boldsymbol{\delta}})}}{[z^n \tilde{z}^{\tilde{m}}] \sum_{k \leq n \cdot \tilde{m}} [w^k] e^{z P_{\text{WGM}}(\mathbf{u}, \boldsymbol{\delta})} \odot_{\mathbf{u}=(h_j(w))_{1 \leq j \leq k}} e^{\tilde{z} P_{\text{WGM}}(\mathbf{u}, \mathbf{1})}}.$$

Note that in both the numerator and denominator, the sum over k is restricted to $0 \leq k \leq n \cdot \tilde{m}$. This restriction arises because we are enumerating bipartite **simple** graphs, where the number of edges is bounded by the product of the sizes of the two vertex sets (here, n and \tilde{m}). In the numerator, which corresponds to Condorcet graphs, the number of edges is in fact further bounded by $(\lceil n/2 \rceil - 1) \cdot \tilde{m}$; however, this tighter bound is implicitly enforced by the choice of $\boldsymbol{\delta}$, so there is no need to explicitly incorporate it in the summation limits.

Since the Hadamard exponential product acts only on the \mathbf{u} variables, we can extract the coefficients in z and \tilde{z} independently. Applying the standard identity

$$[z^n] e^{z P_{\text{WGM}}(\mathbf{u}, \boldsymbol{\delta})} = [z^n] \sum_k \frac{(z P_{\text{WGM}}(\mathbf{u}, \boldsymbol{\delta}))^k}{k!} = \frac{1}{n!} (P_{\text{WGM}}(\mathbf{u}, \boldsymbol{\delta}))^n,$$

a direct computation yields

$$m \cdot \frac{\sum_{k \leq n \cdot \tilde{m}} [w^k] \left((P_{\text{WGM}}(\mathbf{u}, \boldsymbol{\delta}))^n \odot_{\mathbf{u}=(h_j(w))_{1 \leq j \leq k}} (P_{\text{WGM}}(\mathbf{u}, \tilde{\boldsymbol{\delta}}))^{\tilde{m}} \right)}{\sum_{k \leq n \cdot \tilde{m}} [w^k] \left((P_{\text{WGM}}(\mathbf{u}, \boldsymbol{\delta}))^n \odot_{\mathbf{u}=(h_j(w))_{1 \leq j \leq k}} (P_{\text{WGM}}(\mathbf{u}, \mathbf{1}))^{\tilde{m}} \right)}.$$

Therefore, the key is to make explicit the forms of $P_{\text{WGM}}(\mathbf{u}, \boldsymbol{\delta})$, $P_{\text{WGM}}(\mathbf{u}, \tilde{\boldsymbol{\delta}})$, and $P_{\text{WGM}}(\mathbf{u}, \mathbf{1})$, given the following degree sequences

$$\delta_d = d!(\tilde{m} - d)! \quad \text{and} \quad \tilde{\delta}_d = \mathbb{1}_{d \leq \lceil n/2 \rceil - 1}.$$

By direct computation, we have the following expressions

$$\begin{aligned} P_{\text{WGM}}(\mathbf{u}, \boldsymbol{\delta}) &= \left(\sum_{d=0}^{\tilde{m}} d!(\tilde{m} - d)! \frac{y^d}{d!} \right) \odot_{y=1} e^{H(y, \mathbf{u})}, \\ P_{\text{WGM}}(\mathbf{u}, \tilde{\boldsymbol{\delta}}) &= \left(\sum_{d=0}^{\lceil n/2 \rceil - 1} \frac{y^d}{d!} \right) \odot_{y=1} e^{H(y, \mathbf{u})}, \\ P_{\text{WGM}}(\mathbf{u}, \mathbf{1}) &= \left(\sum_{d=0}^{n \cdot \tilde{m}} \frac{y^d}{d!} \right) \odot_{y=1} e^{H(y, \mathbf{u})}. \end{aligned}$$

In each case, the expansion of the exponential generating function $e^{H(y,\mathbf{u})}$ plays a central role. This expansion can be expressed in terms of the Bell polynomials (see, for instance, [Bel19]),

$$\exp\left(\sum_{j=1}^k u_j y^j\right) = \sum_{d=0}^k B_d(\mathbf{u}) \frac{y^d}{d!},$$

where $B_d(\mathbf{u}) = B_d(u_1, u_2, \dots, u_k)$ denotes the truncated Bell polynomial of order d , i.e., the coefficient of y^d in $\exp\left(\sum_{j=1}^k u_j y^j\right)$. Using this notation, we can rewrite these expressions as

$$P_{\text{WMG}}(\mathbf{u}, \boldsymbol{\delta}) = \sum_{d=0}^{\tilde{m}} (\tilde{m} - d)! \cdot B_d(\mathbf{u}), \quad (3.6)$$

$$P_{\text{WMG}}(\mathbf{u}, \tilde{\boldsymbol{\delta}}) = \sum_{d=0}^{\lceil n/2 \rceil - 1} \frac{B_d(\mathbf{u})}{d!}, \quad (3.7)$$

$$P_{\text{WMG}}(\mathbf{u}, \mathbf{1}) = \sum_{d=0}^{n \cdot \tilde{m}} \frac{B_d(\mathbf{u})}{d!}, \quad (3.8)$$

Just for the sake of completeness, let us now write the expression for the probability using Bell polynomials

$$m \cdot \frac{\sum_{k \leq n \cdot \tilde{m}} [w^k] \left(\left(\sum_{d=0}^{\tilde{m}} (\tilde{m} - d)! \cdot B_d(\mathbf{u}) \right)^n \odot_{\mathbf{u}=(h_j(w))_{1 \leq j \leq k}} \left(\sum_{d=0}^{\lceil n/2 \rceil - 1} \frac{B_d(\mathbf{u})}{d!} \right)^{\tilde{m}} \right)}{\sum_{k \leq n \cdot \tilde{m}} [w^k] \left(\left(\sum_{d=0}^{\tilde{m}} (\tilde{m} - d)! \cdot B_d(\mathbf{u}) \right)^n \odot_{\mathbf{u}=(h_j(w))_{1 \leq j \leq k}} \left(\sum_{d \geq 0} \frac{B_d(\mathbf{u})}{d!} \right)^{\tilde{m}} \right)}.$$

At this point, we are faced with the sheer complexity of the expression. Ideally, one would hope to at least extract the coefficient $[w^k]$ by hand. However, the interplay between the Hadamard product and the Bell polynomials renders even this task unmanageable.

Thus, while the formula is, in principle, computationally tractable (especially thanks to the efficiency of evaluating Bell polynomials and Hadamard products via symbolic computation), it is clearly intractable by hand, even for small values of n and m . Furthermore, despite being formally a closed-form expression, it provides no direct combinatorial or probabilistic insight. The presence of nested sums, Hadamard products, and symbolic Bell polynomials makes the structure highly nontransparent, and thus difficult to interpret.

Finally, the extension of this method to other voting problems or cultures appears to be limited. Indeed, the encoding via $\boldsymbol{\delta}$ depends solely on the number of adversaries ranked above a given candidate, without capturing finer details of the rankings. As a result, it fails to encompass other

natural cultures commonly studied in social choice theory. Furthermore, more intricate voting problems may involve dependencies on the voting rule itself or introduce structural constraints, such as interactions between adversaries, that break the bipartite structure underlying the current approach. This makes the method inapplicable in more general settings.

This discussion highlights the limitations of the method in addressing the Condorcet problem, which ultimately led us to abandon this approach in that context. Nevertheless, a more effective and conceptually coherent method will be developed and explored in detail in Chapters 4 and 5. For now, let us turn to a different application, outside the realm of social choice theory, where the current method proves more naturally suited.

3.5.2 Balanced Matrices

Time to play! After all these intricate combinatorial constructions, its time to take a well-deserved break - let's unwind with two puzzles to keep our minds sharp. Fill in the grids of Figure 3.5 following the rules detailed above the grids. If you prefer to skip the fun, feel free to move on to the next paragraph. Otherwise, you can check your solutions in Figure 3.6.

Binairo

- Each row and each column must contain an equal number of zeros and ones.
- No more than two identical digits can appear consecutively in any row or column (i.e., no sequences like 000 or 111).
- All rows and all columns must be distinct.

			0	0	
1		1	1		
		1			
				0	
1	1				

Not Alone

- Each row and each column must contain an equal number of zeros and ones.
- The patterns 010 and 101 are forbidden in rows and columns, that is, no isolated digits, except at the borders.

1	0				0
		1			
0	1		0		
		1			
0			1		
1			1		

Fig. 3.5: Left - a 6×6 Binairo puzzle. Right - a 6×6 Not Alone puzzle.

In [Dou+24], the authors consider the enumeration of 0-1 balanced matrices subject to certain pattern-avoidance constraints. Their motivation originates from a variant of the well-known *Takuzu puzzle* (also known as *Binairo*), namely the *Not Alone* puzzle published by *The New York Times*. The objective of this puzzle is to fill a square grid with zeros and ones such that each row and column contains an equal number of zeros and ones, and the patterns 010 and 101 are forbidden.

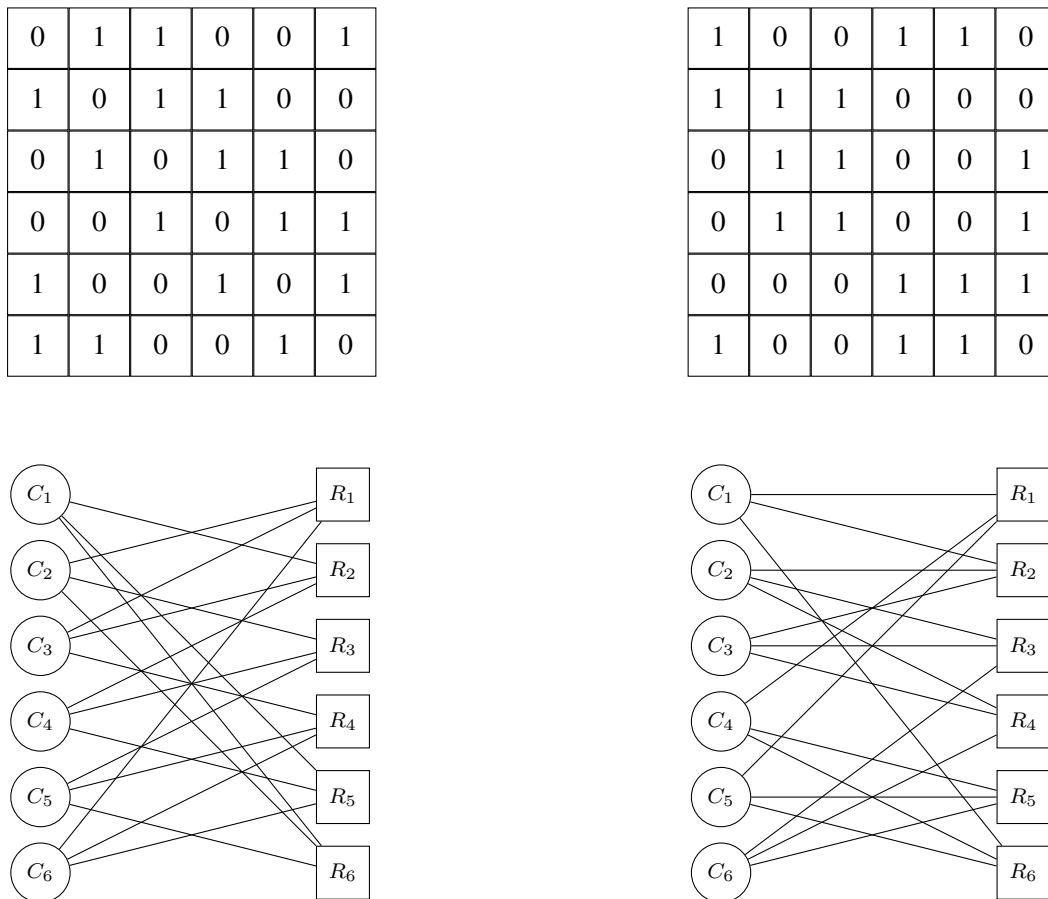


Fig. 3.6: Top - solutions to the Binairo and Not Alone puzzles proposed in Figure 3.5. Bottom - associated bipartite graphs.

As a first step, they study the enumeration of $2k \times 2n$ 0-1 balanced matrices without any pattern constraints, for a fixed number k and an arbitrary symbolic n . Formally, a 0-1 balanced matrix is a matrix with entries in $\{0,1\}$, such that each row and each column contains exactly the same number of zeros and ones. This enumeration appears in OEIS as sequence A376935, which can be seen as the most general form of such enumeration currently available in the database. Noting $b_k(n)$ the number of $2k \times 2n$ balanced matrices, they show that the family of sequences $b_k(n)$ is D-finite for each fixed k . To do so, they use the ‘*holonomic nonsense*’, referring [AZ06; Kau23; Zei90], which

guarantees the existence of such a recurrence and also outlines a method to construct it. Using the efficient implementations of [Kou09; Kou10; Kou14] they were able to compute these recurrences for $k = 2$ and $k = 3$.

The enumeration of such objects can be performed using Theorem 2. Indeed, a 0-1 matrix can be interpreted as the biadjacency matrix of a bipartite graph, where the columns C_1, \dots, C_{2n} correspond to the left vertex set, the rows R_1, \dots, R_{2k} to the right vertex set, and an edge connects R_i to C_j if and only if the entry (i, j) of the matrix is 1. The condition that each row and column contains an equal number of zeros and ones translates to each vertex R_i having degree n and each vertex C_j having degree k .

Therefore, our approach provides an alternative proof of the D-finiteness of $b_k(n)$ for each fixed k , as noted in the remark following Theorem 2. We hope that the explicit formula below will serve as a stepping stone toward obtaining more numbers in the already existing OEIS sequence, and toward deriving concrete recurrence relations for $k \geq 4$. However, note that our method does not accommodate pattern avoidance constraints, as these would require modeling dependencies between adjacent edges, which fall outside the scope of our framework.

Computing the number of $2k \times 2n$ balanced 0-1 matrices is similar to evaluating the probability of the Condorcet paradox: both ultimately rely on defining δ and $\tilde{\delta}$ appropriately, followed by extracting coefficients from power series expansions. However, the definitions of δ and $\tilde{\delta}$ used here will significantly simplify the task.

We begin by expressing the quantity as a coefficient extraction

$$(2n)!(2k)! [z^{2n} \tilde{z}^{2k} w^{2kn}] \text{BSG}(z, \tilde{z}, w, \delta, \tilde{\delta}),$$

where $\delta_d = \mathbb{1}_{d=k}$ and $\tilde{\delta}_d = \mathbb{1}_{d=n}$ for every $d \geq 1$. Note that, unlike the Condorcet case, there is no summation over the number of edges. This simplifies the overall expression. Expliciting BSG we obtain

$$(2n)!(2k)! [z^{2n} \tilde{z}^{2k} w^{2kn}] \left(e^{zP_{\text{WGM}}(\mathbf{u}, \delta)} \odot_{\mathbf{u}=(h_j(w))_{1 \leq j \leq 2kn}} e^{\tilde{z}P_{\text{WGM}}(\mathbf{u}, \tilde{\delta})} \right).$$

As in the Condorcet computation, the Hadamard product is independent of z and \tilde{z} , allowing us to extract the corresponding coefficients beforehand

$$(2n)!(2k)! [w^{2kn}] \left([z^{2n}] e^{zP_{\text{WGM}}(\mathbf{u}, \delta)} \odot_{\mathbf{u}=(h_j(w))_{1 \leq j \leq 2kn}} [\tilde{z}^{2k}] e^{\tilde{z}P_{\text{WGM}}(\mathbf{u}, \tilde{\delta})} \right).$$

Extracting the coefficients in the the exponentials, we get

$$[w^{2kn}] \left(P_{\text{WMG}}(\mathbf{u}, \boldsymbol{\delta})^{2n} \odot_{\mathbf{u}=(h_j(w))_{1 \leq j \leq 2kn}} P_{\text{WMG}}(\mathbf{u}, \tilde{\boldsymbol{\delta}})^{2k} \right).$$

Here lies the main simplification, we have

$$P_{\text{WMG}}(\mathbf{u}, \boldsymbol{\delta}) = \delta_k B_k(u_1, \dots, u_{2kn}) = B_k(\mathbf{u}), \quad P_{\text{WMG}}(\mathbf{u}, \tilde{\boldsymbol{\delta}}) = \delta_n B_n(u_1, \dots, u_{2kn}) = B_n(\mathbf{u}),$$

where $B_k(\mathbf{u})$ is the (truncated) Bell polynomial in $\mathbf{u} = (u_1, \dots, u_{2kn})$. This expression is much simpler than its counterpart in the Condorcet setting. Substituting yields

$$[w^{2kn}] \left((B_k(\mathbf{u}))^{2n} \odot_{\mathbf{u}=(h_j(w))_{1 \leq j \leq 2kn}} (B_n(\mathbf{u}))^{2k} \right).$$

We now expand the Hadamard product and evaluate at $\mathbf{u} = (h_j(w))_{1 \leq j \leq 2kn}$

$$[w^{2kn}] \sum_{\ell_1, \dots, \ell_{2kn} \geq 0} \left(\ell! [\mathbf{u}^\ell] B_k(\mathbf{u})^{2n} \right) \cdot \left(\ell! [\mathbf{u}^\ell] B_n(\mathbf{u})^{2k} \right) \cdot \prod_{j=1}^{2kn} \frac{1}{\ell_j!} \left(\frac{-(-w)^j}{j} \right)^{\ell_j}.$$

Now, note that the degree constraints already impose that the total number of edges equals $2kn$. Therefore, there is no need to extract the coefficient in w ; simply evaluating the expression at $w = 1$ yields the same result. Hence we obtain

$$\sum_{\ell_1, \dots, \ell_{2kn} \geq 0} \left([\mathbf{u}^\ell] B_k(\mathbf{u})^{2n} \right) \cdot \left([\mathbf{u}^\ell] B_n(\mathbf{u})^{2k} \right) \cdot \ell! \prod_{j=1}^{2kn} \left(\frac{-(-1)^j}{j} \right)^{\ell_j}.$$

The remaining steps involve computing the coefficients of Bell polynomials, which can be made fully explicit. First, remember that

$$\exp \left(\sum_{j=1}^k u_j y^j \right) = \sum_{d=0}^k B_d(\mathbf{u}) \frac{y^d}{d!},$$

Therefore,

$$B_k(\mathbf{u}) = k! [y^k] \exp \left(\sum_{j=1}^{2kn} u_j y^j \right) \quad \text{and} \quad B_n(\mathbf{u}) = n! [y^n] \exp \left(\sum_{j=1}^{2kn} u_j y^j \right).$$

Expanding $(B_k(\mathbf{u}))^{2n}$ and extracting the coefficient of \mathbf{u}^ℓ yields

$$[u_1^{\ell_1} \dots u_{2kn}^{\ell_{2kn}}] (B_k(u_1, \dots, u_{2kn}))^{2n} = \sum_{\substack{s^{(1)}, \dots, s^{(2n)} \in \mathbb{N}^n \\ \sum_{i=1}^{2n} s_j^{(i)} = \ell_j \quad \forall j \in \{1, \dots, 2kn\} \\ \sum_{j=1}^{2kn} j s_j^{(i)} = k \quad \forall i \in \{1, \dots, 2n\}}} \prod_{i=1}^{2n} \frac{k!}{\prod_{j=1}^{2kn} s_j^{(i)}! j!^{s_j^{(i)}}},$$

and similarly for $[\mathbf{u}^\ell] B_n(\mathbf{u})^{2k}$.

Now, observe that the Bell polynomial conditions

$$\sum_{j=1}^{2kn} j s_j^{(i)} = k \quad \text{and} \quad \sum_{i=1}^{2n} s_j^{(i)} = \ell_j$$

imply that

$$\sum_{j=1}^{2kn} j \ell_j = 2kn.$$

This shows the consistency with the fact that we only want to consider terms where the total number of edges is $2kn$; this condition is automatically encoded in the Bell polynomial coefficients.

Consequently, the expression becomes

$$\sum_{\ell_1, \dots, \ell_{2kn} \geq 0} ([\mathbf{u}^\ell] B_k(\mathbf{u})^{2n}) \cdot ([\mathbf{u}^\ell] B_n(\mathbf{u})^{2k}) \cdot \ell! \prod_{j=1}^{2kn} \left(\frac{-1}{j}\right)^{\ell_j}.$$

While the resulting expression remains somewhat heavy, as the coefficients in the Bell polynomials are not particularly simple, unlike in the Condorcet case, it has the advantage of being fully explicit and combinatorially meaningful. Also, we obtain a finite formula that allows for polynomial computation.

With this manageable formula at hand, we can now return to the enumeration of graphs with degree constraints, this time moving beyond the bipartite setting.

3.6 General Graphs (Non Bipartite)

Note that this section can safely be skipped by readers primarily interested in social choice aspects. Many of the underlying arguments closely mirror those developed in the bipartite case. However, for those curious about the enumeration of general (non bipartite) graphs, it offers a self-contained

discussion of the additional subtleties involved, most notably the proper handling of loops, which are absent in the bipartite setting, and the fact that the vertex set is no longer partitioned into two parts.

Thus, similarly to the bipartite case, we enumerate successively several graph-like families with degree constraints: *weighted multigraphs*, *loopless weighted multigraphs*, *weighted simple graphs* and finally *simple graphs*. They are all depicted in Figure 3.7. The generating function of weighted multigraphs is expressed directly in Lemma 3. Inclusion-exclusion is applied to express the generating function of loopless weighted multigraphs in Lemma 5. In Lemma 7, the generating function of loopless weighted multigraphs is expressed using the generating function of weighted simple graphs, then this relation is inverted. Finally, the generating function of simple graphs is obtained in Theorem 3 by restricting the weights to 1.

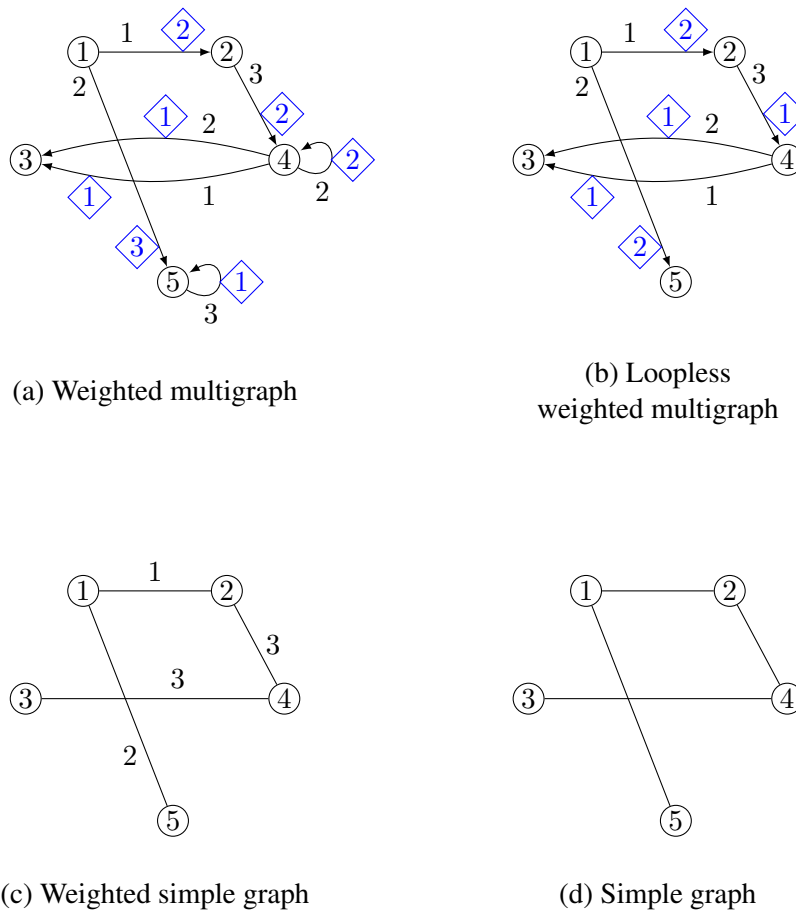


Fig. 3.7: Steps of the transformation from a weighted multigraph to a simple graph. Vertex labels are shown as circled integers, edge labels appear in blue diamond-shaped frames, and edge weights are indicated by unframed black integers.

3.6.1 Weighted Multigraphs

Definition. A weighted multigraph G is a finite sequence

$$G = (V(G), E_1(G), E_2(G), \dots)$$

where $V(G) = \{1, 2, \dots, n\}$ is the set of n vertices, and for all $j \geq 1$, $E_j(G)$ is the finite set of k_j edges of weight j . The labels of the edges in $E_j(G)$ are $\{1, \dots, k_j\}$ and each edge is a triplet (u, v, ℓ) where u and v are vertices from $V(G)$ and ℓ is the label. Thus, all edges are oriented, and loops (edges containing twice the same vertex) and multiple edges (set of edges linking the same vertices) are allowed. For example, in the multigraph G from Figure 3.7 (a), we have $E_2(G) = \{(4, 3, 1), (4, 4, 2), (1, 5, 3)\}$.

The degree $\deg(u)$ of a vertex u is defined as the sum of the weights of all edges adjacent to it, counted twice if they are loops. For example, in Figure 3.7 (a), vertex 5 has degree $2 \times 3 + 2 = 8$.

Generating function. We use the exponential variable z to mark the vertices, and for all $j \geq 1$, we use the exponential variable $w_j/2$ to mark each edge of weight j . This factor $1/2$ will prove useful in Lemma 6 to compare the generating functions of multigraph and graph families (see also the discussion in [Pan18, Section 2.3]). Additionally, for each $d \geq 0$, the ordinary variable δ_d is introduced to mark vertices of degree d . The generating function of a weighted multigraph family \mathcal{A} is then defined as

$$A(z, \mathbf{w}, \boldsymbol{\delta}) = \sum_{G \in \mathcal{A}} \left(\prod_{u=1}^n \delta_{\deg(u)} \right) \left(\prod_{j \geq 1} \frac{(w_j/2)^{k_j}}{k_j!} \right) \frac{z^n}{n!}. \quad (3.9)$$

Note that since each weighted multigraph G has a finite number of edges, each product in the definition of $A(z, \mathbf{w}, \boldsymbol{\delta})$ has finite support. In particular, we note $\text{WMG}(z, \mathbf{w}, \boldsymbol{\delta})$ the generating function of all weighted multigraphs.

Lemma 3. Recall that $\Delta(y, \boldsymbol{\delta})$ and $H(y, \mathbf{x})$ denote the formal power series

$$\Delta(y, \boldsymbol{\delta}) = \sum_{d \geq 0} \delta_d \frac{y^d}{d!}, \quad H(y, \mathbf{x}) = \sum_{j \geq 1} x_j y^j.$$

And $P_{\text{WMG}}(\mathbf{x}, \boldsymbol{\delta})$ denotes the formal power series

$$P_{\text{WMG}}(\mathbf{x}, \boldsymbol{\delta}) = \Delta(y, \boldsymbol{\delta}) \odot_{y=1} e^{H(y, \mathbf{x})}.$$

The generating function of weighted multigraphs is equal to

$$\text{WMG}(z, \mathbf{w}, \boldsymbol{\delta}) = e^{x_1^2/2+x_2^2/2+\dots} \odot_{\mathbf{x}=\mathbf{1}} e^{zP_{\text{WMG}}((w_j^{1/2}x_j)_{j \geq 1}, \boldsymbol{\delta})}.$$

When $\boldsymbol{\delta}$ is a vector of real values with $\delta_d = 0$ for all $d > D$, $P_{\text{WMG}}(\mathbf{x}, \boldsymbol{\delta})$ becomes a polynomial in (x_1, \dots, x_D) and

$$\text{WMG}(z, \mathbf{w}, \boldsymbol{\delta}) = e^{x_1^2/2+\dots+x_D^2/2} \odot_{\mathbf{x}=\mathbf{1}} e^{zP_{\text{WMG}}(w_1^{1/2}x_1, \dots, w_D^{1/2}x_D, \boldsymbol{\delta})}.$$

Remark. The right-hand side of our last result contains half-powers in w_j . However, after the Hadamard product with $e^{x_j^2/2}$, only even powers in x_j are kept, so the result is, as expected, a formal power series in w_j (instead of $w_j^{1/2}$).

Before proceeding with the proof, we first state this standard result in the literature.

Lemma 4. Consider $F(y, \mathbf{x})$ a formal power series in y with coefficients in the polynomial ring on $(x_j)_{j \geq 1}$. Then

$$e^{x_1^2/2+x_2^2/2+\dots} \odot_{\mathbf{x}=\mathbf{1}} F(y, \mathbf{x}) = \sum_{\mathbf{k}} (2\mathbf{k} - \mathbf{1})!! [\mathbf{x}^{2\mathbf{k}}] F(y, \mathbf{x}). \quad (3.10)$$

with the sum over all infinite vectors of nonnegative integers.

Proof. (Lemma 4) First observe that

$$e^{x^2/2} \odot_{x=1} x^\ell = \sum_{k \geq 0} \frac{(2k)!}{2^k k!} \frac{x^{2k}}{(2k)!} \odot_{x=1} \ell! \frac{x^\ell}{\ell!} = \begin{cases} 0 & \text{if } \ell \text{ is odd,} \\ \frac{(2k)!}{2^k k!} & \text{if } \ell = 2k. \end{cases}$$

so for any polynomial $Q(\mathbf{x})$ of the polynomial ring on $(x_j)_{j \geq 1}$, we have

$$e^{x_1^2/2+x_2^2/2+\dots} \odot_{\mathbf{x}=\mathbf{1}} Q(\mathbf{x}) = \sum_{\mathbf{k}} (2\mathbf{k} - \mathbf{1})!! [\mathbf{x}^{2\mathbf{k}}] Q(\mathbf{x})$$

where the sum is over all infinite vectors of nonnegative integers. Note that $[\mathbf{x}^{2\mathbf{k}}]Q(\mathbf{x})$ is nonzero for finitely many vectors \mathbf{k} , all with finite support (as no monomial of $Q(\mathbf{x})$ can have infinitely

many variables), so the sum is always finite. Now, consider $F(y, \mathbf{x})$ a formal power series in y with coefficients in the polynomial ring on $(x_j)_{j \geq 1}$. First, decompose $F(y, \mathbf{x})$ with respect to z :

$$F(y, \mathbf{x}) = \sum_{t \geq 0} [z^t] F(z, \mathbf{x}) y^t.$$

Applying the Hadamard product with $e^{x_1^2/2+x_2^2/2+\dots}$ gives

$$e^{x_1^2/2+x_2^2/2+\dots} \odot_{\mathbf{x}=\mathbf{1}} F(y, \mathbf{x}) = e^{x_1^2/2+x_2^2/2+\dots} \odot_{\mathbf{x}=\mathbf{1}} \sum_{t \geq 0} [z^t] F(z, \mathbf{x}) y^t.$$

We can then exchange the Hadamard product and the summation:

$$e^{x_1^2/2+x_2^2/2+\dots} \odot_{\mathbf{x}=\mathbf{1}} \sum_{t \geq 0} [z^t] F(z, \mathbf{x}) y^t = \sum_{t \geq 0} \left(e^{x_1^2/2+x_2^2/2+\dots} \odot_{\mathbf{x}=\mathbf{1}} [z^t] F(z, \mathbf{x}) \right) y^t.$$

By definition of the exponential Hadamard product, this becomes

$$\sum_{t \geq 0} \sum_{\mathbf{k}} (2\mathbf{k} - 1)!! [z^t \mathbf{x}^{2\mathbf{k}}] F(z, \mathbf{x}) y^t,$$

and finally we can regroup by powers of y , yielding

$$e^{x_1^2/2+x_2^2/2+\dots} \odot_{\mathbf{x}=\mathbf{1}} F(y, \mathbf{x}) = \sum_{\mathbf{k}} (2\mathbf{k} - 1)!! [\mathbf{x}^{2\mathbf{k}}] F(y, \mathbf{x}). \quad \square$$

With this result at hand, we can now turn to the proof of Lemma 3.

Proof. (Lemma 3) Consider a weighted multigraph G . Let us cut each edge (u, v, ℓ) of weight j into two labeled half-edges of weight j , one linked to the vertex u and of label $2\ell - 1$, the other linked to the vertex v and of label 2ℓ . We see that any weighted multigraph decomposes uniquely as a set of labeled vertices, each attached to a set of labeled weighted half-edges, where the number of half-edges is even for any weight j . Let us now derive the corresponding generating functions, using the symbolic method from [FS09]. We start with just one half-edge. We use the exponential variable x_j to mark the labeled half-edges of weight j and the ordinary variable y to mark the degree (for now equal to the weight), so the generating function of one half-edge is

$$H(y, \mathbf{x}) = \sum_{j \geq 1} x_j y^j.$$

We use the exponential variable z to mark vertices. Since each vertex has a set of labeled half-edges of various weights attached to it, and its degree is the sum of the weights of those half-edges, the generating function of one vertex with attached half-edges is

$$ze^{H(y,x)}.$$

For each d , instead of marking each vertex of degree d with y^d , we want to use the variable δ_d . Since the variable y is ordinary, this could be achieved by taking the *ordinary* Hadamard product of $ze^{H(y,x)}$ with $\sum_{d \geq 1} \delta_d y^d$. However, we are reluctant to introduce two notations for the exponential and ordinary Hadamard products. Instead, we observe that the ordinary Hadamard product of any series $A(y)$ with $\sum_{d \geq 1} \delta_d y^d$ is equal to the exponential Hadamard product of $A(y)$ with $\sum_{d \geq 1} \delta_d \frac{y^d}{d!} = \Delta(y, \delta)$. The generating function of one vertex then becomes

$$z\Delta(y, \delta) \odot_{y=1} e^{H(y,x)} = zP_{\text{WMG}}(x, \delta).$$

The generating function of a set of vertices, each attached to a set of half-edges, is

$$e^{zP_{\text{WMG}}(x, \delta)}.$$

For any sequence of nonnegative integers \mathbf{k} with finite support, the coefficient

$$(2\mathbf{k})![\mathbf{x}^{2\mathbf{k}}]e^{zP_{\text{WMG}}(x, \delta)}$$

is the generating function of sets of labeled vertices, each coming with a set of half-edges, with globally $2k_j$ half-edges of weight j , for all $j \geq 1$.

For all $j \geq 1$, we introduce the exponential variable $w_j/2$ to mark edges of weight j . We deduce that the generating function of weighted multigraphs is equal to

$$\text{WMG}(z, \mathbf{w}, \delta) = \sum_{\mathbf{k}} (2\mathbf{k})![\mathbf{x}^{2\mathbf{k}}]e^{zP_{\text{WMG}}(x, \delta)} \prod_{j \geq 1} \frac{(w_j/2)^{k_j}}{k_j!}.$$

Using the double factorial notation (3.2), this generating function becomes

$$\text{WMG}(z, \mathbf{w}, \delta) = \sum_{\mathbf{k}} (2\mathbf{k} - \mathbf{1})![\mathbf{x}^{2\mathbf{k}}]e^{zP_{\text{WMG}}(x, \delta)} \mathbf{w}^{\mathbf{k}}.$$

This expression is simplified by applying Lemma 4

$$\text{WMG}(z, \mathbf{w}, \delta) = e^{x_1^2/2 + x_2^2/2 + \dots} \odot_{\mathbf{x}=\mathbf{1}} e^{zP_{\text{WMG}}((w_j^{1/2}x_j)_{j \geq 1}, \delta)}. \quad \square$$

A multigraph is *loopless* if it has no edge containing twice the same vertex. The generating function $\text{LWMG}(z, \mathbf{w}, \boldsymbol{\delta})$ of those weighted multigraphs is defined following the convention from Equation (3.9) and we use inclusion-exclusion to express it.

Lemma 5. *Let $P_{\text{LWMG}}(\mathbf{x}, \mathbf{w}, \boldsymbol{\delta})$ denote the formal power series*

$$P_{\text{LWMG}}(\mathbf{x}, \mathbf{w}, \boldsymbol{\delta}) = \Delta(y, \boldsymbol{\delta}) \odot_{y=1} e^{H(y, \mathbf{x}) - H(y^2, \mathbf{w})/2},$$

then the generating function of loopless weighted multigraphs is equal to

$$\text{LWMG}(z, \mathbf{w}, \boldsymbol{\delta}) = e^{x_1^2/2 + x_2^2/2 + \dots} \odot_{\mathbf{x}=1} e^{z P_{\text{LWMG}}((w_j^{1/2} x_j)_{j \geq 1}, \mathbf{w}, \boldsymbol{\delta})}.$$

When $\boldsymbol{\delta}$ is a vector of real values with $\delta_d = 0$ for all $j > d$, then $P_{\text{LWMG}}(\mathbf{x}, \mathbf{w}, \boldsymbol{\delta})$ becomes a polynomial in $(x_1, \dots, x_D, w_1, \dots, w_D)$ and

$$\text{LWMG}(z, \mathbf{w}, \boldsymbol{\delta}) = e^{x_1^2/2 + \dots + x_D^2/2} \odot_{\mathbf{x}=1} e^{z P_{\text{LWMG}}((w_j^{1/2} x_j)_{1 \leq j \leq D}, \mathbf{w}, \boldsymbol{\delta})}.$$

Proof. To enumerate loopless weighted multigraphs, we will apply the generating function formulation of the inclusion-exclusion principle [FS09, Section III. 7.4.]. Let $\text{WMG}(z, \mathbf{w}, \boldsymbol{\delta}, u)$ denote the generating function of weighted multigraphs from Equation (3.9), with an additional parameter u marking all loops (edges containing twice the same vertex). The motivation for introducing this generating function is that the generating function of loopless weighted multigraphs is then obtained by evaluation at $u = 0$

$$\text{LWMG}(z, \mathbf{w}, \boldsymbol{\delta}) = \text{WMG}(z, \mathbf{w}, \boldsymbol{\delta}, 0). \quad (3.11)$$

The inclusion-exclusion principle suggests to consider weighted multigraphs where an arbitrary subset of loops is distinguished and marked by an ordinary variable v . The generating function of this family is $\text{WMG}(z, \mathbf{w}, \boldsymbol{\delta}, v + 1)$, because each loop is either marked by v , or left unmarked. To express this generating function, we follow the proof of Lemma 3. We cut each edge that is not a distinguished loop into two weighted labeled half-edges. Using the exponential variable x_j to mark half-edges of weight j and the ordinary variable y to mark the degree, the generating function of one half-edge is, as previously,

$$\sum_{j \geq 1} x_j y^j = H(y, \mathbf{x}).$$

We use the exponential variable $w_j/2$ to mark the distinguished loops of weight j . Each loop increases the degree by $2j$, so the generating function of distinguished loops is

$$v \sum_{j \geq 1} \frac{w_j}{2} y^{2j} = vH(y^2, \mathbf{w})/2.$$

The exponential variable z marks the vertices. Each vertex is attached to a set of half-edges and a set of distinguished loops, so the generating function of one vertex is

$$ze^{H(y, \mathbf{x}) + vH(y^2, \mathbf{w})/2}.$$

For each d , instead of marking the vertices of degree d with y^d , we want to mark them with the ordinary variable δ_d . As in the previous proof, this corresponds to an exponential Hadamard product with $\Delta(y, \boldsymbol{\delta})$, so the generating function of the half-edges and distinguished loops attached to one vertex is

$$z\Delta(y, \boldsymbol{\delta}) \odot_{y=1} e^{H(y, \mathbf{x}) + vH(y^2, \mathbf{w})/2}.$$

Let us set

$$Q(\mathbf{x}, \mathbf{w}, \boldsymbol{\delta}, v) = \Delta(y, \boldsymbol{\delta}) \odot_{y=1} e^{H(y, \mathbf{x}) + vH(y^2, \mathbf{w})/2}.$$

The rest of the proof is identical to the one of Lemma 3 and we deduce the following expression for the generating function of weighted multigraphs where v marks a subset of loops

$$\text{WMG}(z, \mathbf{w}, \boldsymbol{\delta}, v + 1) = e^{x_1^2/2 + x_2^2/2 + \dots} \odot_{x=1} e^{zQ((w_j^{1/2}x_j)_{j \geq 1}, \mathbf{w}, \boldsymbol{\delta}, v)}.$$

In order to express the generating function $\text{LWMG}(z, \mathbf{w}, \boldsymbol{\delta})$ of loopless weighted multigraphs, equal to $\text{WMG}(z, \mathbf{w}, \boldsymbol{\delta}, 0)$ according to Equation (3.11), we evaluate the last equation at $v = -1$. The result of the lemma follows, after defining

$$P_{\text{LWMG}}(\mathbf{x}, \mathbf{w}, \boldsymbol{\delta}) = Q(\mathbf{x}, \mathbf{w}, \boldsymbol{\delta}, -1). \quad \square$$

3.6.2 Simple Graphs

Definitions. A weighted simple graph G is a finite sequence

$$G = (V(G), E_1(G), E_2(G), \dots)$$

where $V(G) = \{1, 2, \dots, n\}$ is the set of n vertices, and $\bigcup_j E_j(G)$ is a finite set of distinct edges, which are unordered pairs of distinct vertices. $E_j(G)$ denotes the set of edges of weight j and its

cardinality is k_j . Thus, edges are undirected and unlabeled, loops and multiple edges are forbidden, and two vertices cannot be linked by two edges, even if those edges have different weights. The degree of a vertex is still defined as the sum of the weights of its adjacent edges.

Simple graphs correspond to the case $G = (V(G), E_1(G))$, so $E_j = \emptyset$ for all $j \geq 2$.

Generating function. The generating function of a weighted simple graph family \mathcal{B} is defined as

$$B(z, \mathbf{w}, \boldsymbol{\delta}) = \sum_{G \in \mathcal{B}} \left(\prod_{u=1}^n \delta_{\deg(u)} \right) \left(\prod_{j \geq 1} w_j^{k_j} \right) \frac{z^n}{n!}. \quad (3.12)$$

Since vertices are labeled and edges are unlabeled, this generating function is exponential with respect to the variable z marking the vertices, and ordinary with respect to the variables $(w_j)_{j \geq 1}$ marking the edges. Note that the generating function of an edge of weight j is w_j (for weighted multigraphs, we used the convention $w_j/2$, linked to the fact that edges were oriented).

In particular, we denote by $\text{WSG}(z, \mathbf{w}, \boldsymbol{\delta})$ the generating function of all weighted simple graphs.

From a weighted simple graph family, a natural way to construct a weighted multigraph family is to orient and label the edges in all possible ways. Our next result is that the generating functions of both families, using the distinct conventions for simple graphs and multigraphs, are equal. We will not need this result in the rest of this chapter, but we believe it sheds light on the connections between simple graphs and multigraphs.

Lemma 6. *Consider a weighted simple graph family \mathcal{B} and the weighted multigraph family \mathcal{A} obtained from \mathcal{B} by labeling and orienting the edges in all possible ways. Let $A(z, \mathbf{w}, \boldsymbol{\delta})$ and $B(z, \mathbf{w}, \boldsymbol{\delta})$ denote the generating functions associated to the families \mathcal{A} and \mathcal{B} using the conventions from (3.9) and (3.12), then*

$$A(z, \mathbf{w}, \boldsymbol{\delta}) = B(z, \mathbf{w}, \boldsymbol{\delta}). \quad (3.13)$$

Proof. The number of weighted multigraphs identical up to changes of edge orientation and edge label depends on the number of loops and multiple edges. However, the key point of this lemma is that by constructions, the multigraphs from \mathcal{A} contain neither loops nor multiple edges.

Consider a weighted simple graph G from \mathcal{B} with k_j edges of weight j for all j . The number of ways to label and orient its edges is exactly $\prod_j 2^{k_j} k_j!$, which is also the number of multigraphs in \mathcal{A} corresponding to G . Grouping them, we deduce

$$\begin{aligned}
A(z, \mathbf{w}, \boldsymbol{\delta}) &= \sum_{G \in \mathcal{A}} \left(\prod_{u=1}^n \delta_{\deg(u)} \right) \left(\prod_{j \geq 1} \frac{(w_j/2)^{k_j}}{k_j!} \right) \frac{z^n}{n!} \\
&= \sum_{G \in \mathcal{B}} \left(\prod_{j \geq 1} 2^{k_j} k_j! \right) \left(\prod_{u=1}^n \delta_{\deg(u)} \right) \left(\prod_{j \geq 1} \frac{(w_j/2)^{k_j}}{k_j!} \right) \frac{z^n}{n!} \\
&= \sum_{G \in \mathcal{B}} \left(\prod_{u=1}^n \delta_{\deg(u)} \right) \left(\prod_{j \geq 1} w_j^{k_j} \right) \frac{z^n}{n!} \\
&= B(z, \mathbf{w}, \boldsymbol{\delta}). \quad \square
\end{aligned}$$

Observe that the convention on the generating function of weighted multigraph families to mark edges of weight j with $w_j/2$ instead of w_j allows Equation (3.13) to be particularly elegant.

Lemma 7. For $j \geq 0$, let $v_j(\mathbf{w})$ and $P_{\text{WSG}}(\mathbf{x}, \mathbf{w}, \boldsymbol{\delta})$ denote the formal power series

$$v_j(\mathbf{w}) = [y^j] \log(1 + H(y, \mathbf{w})), \quad P_{\text{WSG}}(\mathbf{x}, \mathbf{w}, \boldsymbol{\delta}) = \Delta(y, \boldsymbol{\delta}) \odot_{y=1} \frac{e^{H(y, \mathbf{x})}}{\sqrt{1 + H(y^2, \mathbf{w})}},$$

then the generating function of weighted simple graphs is equal to

$$\text{WSG}(z, \mathbf{w}, \boldsymbol{\delta}) = e^{x_1^2/2 + x_2^2/2 + \dots} \odot_{\mathbf{x}=\mathbf{1}} e^{z P_{\text{WSG}}((v_j(\mathbf{w})^{1/2} x_j)_{j \geq 1}, \mathbf{w}, \boldsymbol{\delta})}.$$

When $\boldsymbol{\delta}$ is a vector of real values with $\delta_d = 0$ for all $d > D$, then $P_{\text{WSG}}(\mathbf{x}, \mathbf{w}, \boldsymbol{\delta})$ becomes a polynomial in $(x_1, \dots, x_D, w_1, \dots, w_D)$ and

$$\text{WSG}(z, \mathbf{w}, \boldsymbol{\delta}) = e^{x_1^2/2 + \dots + x_D^2/2} \odot_{\mathbf{x}=\mathbf{1}} e^{z P_{\text{WSG}}((v_j(\mathbf{w})^{1/2} x_j)_{1 \leq j \leq D}, \mathbf{w}, \boldsymbol{\delta})}.$$

Proof. We associate to each weighted simple graph G the following family $L(G)$ of loopless weighted multigraphs, illustrated in Figure 3.8. $L(G)$ contains all weighted multigraphs such that for any two vertices u, v , the sum of the weights of the edges linking u to v or v to u is equal to the weight of the edge $\{u, v\}$ in G . In particular, if there is no edge $\{u, v\}$ in G , then there is no edge linking u and v in any multigraph from $L(G)$. Observe that, by construction, the degree of a vertex in G is equal to the degree of the corresponding vertex in any multigraph from $L(G)$. Said otherwise, any weighted multigraph from $L(G)$ is obtained by replacing each edge $\{u, v\}$ of weight j from G with an arbitrary set of labeled oriented weighted edges linking u to v or v to u , and whose weights sum to j .

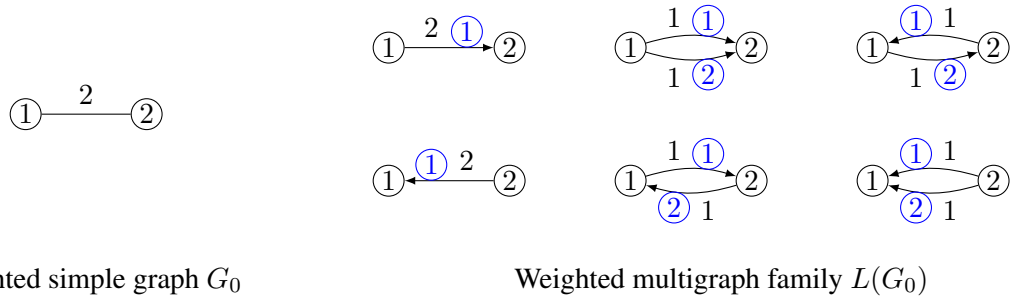


Fig. 3.8: Left, a weighted simple graph G_0 . Right, the six weighted multigraphs of the family $L(G_0)$ defined in the proof of Lemma 7. The labels of the vertices are circled, the labels of the edges are circled and in blue, the weights of the edges are the non-circled numbers.

For each $j \geq 0$, we use the exponential variable $v_j/2$ to mark edges of weight j in loopless weighted multigraphs, with the convention $v_0 = 0$ (as we forbid edges of weight 0). Thus, the generating function of one edge, oriented from a vertex s to a vertex t , of arbitrary weight marked by y , is

$$\sum_{d \geq 1} \frac{v_d}{2} y^d.$$

Since this edge cannot be a loop, it has two distinct possible orientations, so the generating function of one edge either oriented from s to t or from t to s , is obtained by multiplying the previous expression by 2

$$\sum_{d \geq 1} v_d y^d.$$

The generating function of a nonempty set of such edges is

$$\exp\left(\sum_{d \geq 1} v_d y^d\right) - 1.$$

Set for all $j \geq 0$

$$w_j(\mathbf{v}) = [y^j] \left(\exp\left(\sum_{d \geq 1} v_d y^d\right) - 1 \right), \quad (3.14)$$

so $w_j(\mathbf{v})$ is the generating function of a nonempty set of edges whose weights sum to j and $w_0(\mathbf{v}) = 0$. Consider our weighted simple graph G and let n_d denote the number of vertices of degree d , n the total number of vertices and k_j the number of edges of weight j . Then $L(G)$ has generating function (following the convention from Equation (3.12))

$$\left(\prod_{j \geq 1} w_j(\mathbf{v})^{k_j} \right) \left(\prod_{d \geq 0} \delta_d^{n_d} \right) \frac{z^n}{n!}$$

Let us use the ordinary variable w_j to mark edges of weight j in our weighted simple graph G . Then the generating function of G (following the convention from Equation (3.9)) is

$$\left(\prod_{j \geq 1} w_j^{k_j} \right) \left(\prod_{d \geq 0} \delta_d^{n_d} \right) \frac{z^n}{n!}.$$

Observe that the generating function of $L(G)$ is obtained from the generating function of G by replacing w_j with $w_j(\mathbf{v})$ for all j . Summing over all weighted simple graphs G yields

$$\text{WSG}(z, \mathbf{w}(\mathbf{v}), \boldsymbol{\delta}) = \text{LWMG}(z, \mathbf{v}, \boldsymbol{\delta}).$$

Equation (3.14) expresses w_j as a function of \mathbf{v} for all $j \geq 0$. Let us now invert it to express v_j as a function of \mathbf{w} for all $j \geq 0$. We have

$$H(y, \mathbf{w}(\mathbf{v})) = \sum_{d \geq 0} w_d(\mathbf{v}) y^d = \exp \left(\sum_{j \geq 1} v_j y^j \right) - 1,$$

a straightforward calculation gives

$$v_j = [y^j] \log(1 + H(y, \mathbf{w}(\mathbf{v}))).$$

Defining

$$v_j(\mathbf{w}) = [y^j] \log(1 + H(y, \mathbf{w}))$$

yields

$$\text{WSG}(z, \mathbf{w}, \boldsymbol{\delta}) = \text{LWMG}(z, \mathbf{v}(\mathbf{w}), \boldsymbol{\delta}).$$

Injecting the expression of $\text{LWMG}(z, \mathbf{w}, \boldsymbol{\delta})$ from Lemma 5, we obtain

$$\text{WSG}(z, \mathbf{w}, \boldsymbol{\delta}) = e^{x_1^2/2 + x_2^2/2 + \dots} \odot_{\mathbf{x}=\mathbf{1}} e^{z P_{\text{LWMG}}((v_j(\mathbf{w})^{1/2} x_j)_{j \geq 1}, \mathbf{v}(\mathbf{w}), \boldsymbol{\delta})},$$

where

$$P_{\text{LWMG}}(\mathbf{x}, \mathbf{w}, \boldsymbol{\delta}) = \Delta(y, \boldsymbol{\delta}) \odot_{y=1} e^{H(y, \mathbf{x}) - H(y^2, \mathbf{w})/2}.$$

Let us set

$$P_{\text{WSG}}(\mathbf{x}, \mathbf{w}, \boldsymbol{\delta}) = P_{\text{LWMG}}(\mathbf{x}, \mathbf{v}(\mathbf{w}), \boldsymbol{\delta})$$

so

$$\text{WSG}(z, \mathbf{w}, \boldsymbol{\delta}) = e^{x_1^2/2 + x_2^2/2 + \dots} \odot_{\mathbf{x}=\mathbf{1}} e^{z P_{\text{WSG}}((v_j(\mathbf{w})^{1/2} x_j)_{j \geq 1}, \mathbf{w}, \boldsymbol{\delta})}.$$

By definition of $\mathbf{v}(\mathbf{w})$, we have

$$\begin{aligned} H(y, \mathbf{v}(\mathbf{w})) &= \sum_{j \geq 1} v_j(\mathbf{w}) y^j \\ &= \sum_{j \geq 1} [t^j] \log(1 + H(t, \mathbf{w})) y^j \\ &= \log(1 + H(y, \mathbf{w})) \end{aligned}$$

so

$$e^{H(y, \mathbf{x}) - H(y^2, \mathbf{v}(\mathbf{w})) / 2} = \frac{e^{H(y, \mathbf{x})}}{\sqrt{1 + H(y^2, \mathbf{w})}},$$

and

$$P_{\text{WSG}}(\mathbf{x}, \mathbf{w}, \boldsymbol{\delta}) = \Delta(y, \boldsymbol{\delta}) \odot_{y=1} \frac{e^{H(y, \mathbf{x})}}{\sqrt{1 + H(y^2, \mathbf{w})}}. \quad \square$$

We finally arrive at (unweighted) simple graphs in our next result by keeping only edges of weight 1.

Theorem 3. Recall that $\Delta(y, \boldsymbol{\delta})$ and $H(y, \mathbf{x})$ denote the formal power series

$$\Delta(y, \boldsymbol{\delta}) = \sum_{d \geq 1} \delta_d \frac{y^d}{d!}, \quad H(y, \mathbf{x}) = \sum_{j \geq 1} x_j y^j.$$

Let $P_{\text{SG}}(\mathbf{x}, w, \boldsymbol{\delta})$ denote the formal power series

$$P_{\text{SG}}(\mathbf{x}, w, \boldsymbol{\delta}) = \Delta(y, \boldsymbol{\delta}) \odot_{y=1} \frac{e^{H(y, \mathbf{x})}}{\sqrt{1 + wy^2}},$$

and set for all $j \geq 1$

$$h_j(w) = -\frac{(-w)^j}{j}.$$

Then the generating function of simple graphs is equal to

$$\text{SG}(z, w, \boldsymbol{\delta}) = e^{x_1^2/2 + x_2^2/2 + \dots} \odot_{\mathbf{x}=1} e^{z P_{\text{SG}}((h_j(w)^{1/2} x_j)_{j \geq 1}, w, \boldsymbol{\delta})}.$$

When $\delta_d = 0$ for all $d > D$, $P_{\text{SG}}(\mathbf{x}, w, \boldsymbol{\delta})$ becomes a polynomial in (x_1, \dots, x_D, w) and

$$\text{SG}(z, w, \boldsymbol{\delta}) = e^{x_1^2/2 + \dots + x_D^2/2} \odot_{\mathbf{x}=1} e^{z P_{\text{SG}}((h_j(w)^{1/2} x_j)_{1 \leq j \leq D}, w, \boldsymbol{\delta})}.$$

Observe again that the square roots of $h_j(w)$ are just apparent, as only the even powers survive the exponential Hadamard product with $e^{x_1^2/2+\dots+x_D^2/2}$.

Proof. A simple graph is just a weighted simple graph where all edges have weight 1. Thus, simple graphs are enumerated by application of Lemma 7 with $\mathbf{w} = (w, 0, 0, \dots, 0)$, so

$$v_j(\mathbf{w}) = [y^j] \log(1 + wy) = -\frac{(-w)^j}{j} = h_j(w)$$

and

$$H(y^2, \mathbf{w}) = wy^2,$$

and we define

$$P_{\text{SG}}(\mathbf{x}, w, \boldsymbol{\delta}) = P_{\text{WSG}}(\mathbf{x}, (w, 0, \dots, 0), \boldsymbol{\delta}). \quad \square$$

We now turn to κ -regular graphs, a case of particular interest.

Corollary 2. Consider $\kappa \geq 2$ and the polynomial

$$P_{\text{RG}, \kappa}(\mathbf{x}) = [y^\kappa] \frac{\exp \left[\sum_{j=1}^{\kappa} \frac{(-1)^{(j+1)/2}}{\sqrt{j}} x_j y^j \right]}{\sqrt{1 + y^2}}$$

The generating function of κ -regular graphs is equal to

$$e^{x_1^2/2+\dots+x_\kappa^2/2} \odot_{\mathbf{x}=\mathbf{1}} e^{z P_{\text{RG}, \kappa}(\mathbf{x})}.$$

Thus, setting $i = \sqrt{-1}$, the number $\text{SG}_n^{(\kappa)}$ of κ -regular graphs with n vertices is

$$\text{SG}_n^{(\kappa)} = i^{n\kappa} e^{x_1^2/2+\dots+x_\kappa^2/2} \odot_{\mathbf{x}=\mathbf{1}} \left([y^\kappa] \frac{\exp \left[i \sum_{j=1}^{\kappa} x_j y^j / \sqrt{j} \right]}{\sqrt{1 - y^2}} \right)^n. \quad (3.15)$$

Proof. Theorem 3 is applied with

$$\delta_d = \begin{cases} 1 & \text{if } d = \kappa, \\ 0 & \text{otherwise.} \end{cases}$$

Since the number of edges in a κ -regular graphs is determined by the number of vertices, the variable w marking the edges is redundant and simply evaluated at 1. The notation $h_j(w) = -\frac{(-w)^j}{j}$ from

Theorem 3 becomes $\frac{(-1)^{j+1}}{j}$, the function $\Delta(y, \delta)$ becomes $\frac{y^\kappa}{\kappa!}$ and $P_{\text{SG}}\left((h_j(w)^{1/2}x_j)_{j \geq 1}, w, \delta\right)$, by definition equal to

$$\Delta(y, \delta) \odot_{y=1} \frac{e^{H(y, (h_j(w)^{1/2}x_j)_{j \geq 1})}}{\sqrt{1 + wy^2}},$$

becomes

$$\frac{y^\kappa}{\kappa!} \odot_{y=1} \frac{\exp\left[\sum_{j=1}^{\kappa} \frac{(-1)^{(j+1)/2}}{\sqrt{j}} x_j y^j\right]}{\sqrt{1 + y^2}} = [y^\kappa] \frac{\exp\left[\sum_{j=1}^{\kappa} \frac{(-1)^{(j+1)/2}}{\sqrt{j}} x_j y^j\right]}{\sqrt{1 + y^2}},$$

which we denote by $P_{\text{RG}, \kappa}(\mathbf{x})$. Thus, Theorem 3 expresses the generating function of κ -regular graphs as

$$e^{x_1^2/2 + \dots + x_\kappa^2/2} \odot_{\mathbf{x}=\mathbf{1}} e^{z P_{\text{RG}, \kappa}(\mathbf{x})}.$$

The number $\text{SG}_n^{(\kappa)}$ of κ -regular graphs on n vertices is obtained by coefficient extraction

$$\begin{aligned} \text{SG}_n^{(\kappa)} &= n! [z^n] e^{x_1^2/2 + \dots + x_\kappa^2/2} \odot_{\mathbf{x}=\mathbf{1}} e^{z P_{\text{RG}, \kappa}(\mathbf{x})} \\ &= e^{x_1^2/2 + \dots + x_\kappa^2/2} \odot_{\mathbf{x}=\mathbf{1}} \left(n! [z^n] e^{z P_{\text{RG}, \kappa}(\mathbf{x})} \right) \\ &= e^{x_1^2/2 + \dots + x_\kappa^2/2} \odot_{\mathbf{x}=\mathbf{1}} \left([y^\kappa] \frac{\exp\left[\sum_{j=1}^{\kappa} \frac{(-1)^{(j+1)/2}}{\sqrt{j}} x_j y^j\right]}{\sqrt{1 + y^2}} \right)^n. \end{aligned}$$

Applying the identity $[y^\kappa] F(\alpha y) = \alpha^\kappa [y^\kappa] F(y)$ with $\alpha = i$ concludes the proof of the second result (3.15). \square

Now, we have the following equation

$$e^{x_1^2/2 + \dots + x_\kappa^2/2} \odot_{\mathbf{x}=\mathbf{1}} Q(\mathbf{x}) = \int_{(-\infty, +\infty)^\kappa} Q(\mathbf{t}) e^{t_1^2/2 + \dots + t_\kappa^2/2} dt. \quad (3.16)$$

Applying it to our last result yields the following integral expression for $\text{SG}_n^{(\kappa)}$

$$\text{SG}_n^{(\kappa)} = i^{n\kappa} \int_{(-\infty, +\infty)^\kappa} \left([y^\kappa] \frac{\exp\left[i \sum_{j=1}^{\kappa} t_j y^j / \sqrt{j}\right]}{\sqrt{1 - y^2}} \right)^n dt$$

where the integrand is a polynomial in t_1, \dots, t_κ . This provides a path to apply analytic techniques to our exact formula. Continuing in this direction, Élie de Panafieu recently computed the asymptotic expansion of regular graphs [Pan24].

3.7 Conclusion

In this chapter, we have explored the interplay between social choice theory and enumerative combinatorics, with a particular focus on the classical Condorcet paradox. By adopting a combinatorial perspective, we reformulated the probability of the paradox in terms of the enumeration of bipartite graphs with degree constraints, offering a novel viewpoint on this well-studied problem.

Building on this idea, we developed generating functions for both bipartite and non bipartite graphs with degree constraints. While this approach turned out to be somewhat inefficient for addressing social choice questions, it nonetheless revealed interesting structural features and proved relevant in other contexts, such as the enumeration of balanced 0-1 matrices or κ -regular graphs. These objects, though originating in different areas, benefit from the same underlying combinatorial framework, highlighting the broader applicability of the method.

That said, we now turn to a more robust framework for studying the Condorcet paradox using *analytic combinatorics*. Grounded in complex analysis and singularity theory, this methodology will allow us to go beyond explicit formulas and address asymptotic behaviors.

Probability of the Condorcet Paradox for Large Electorates

4.1 Introduction

We now turn to the second method for computing the probability of occurrence of the Condorcet paradox. Before reading this chapter, I recommend referring to Section 2.2 of Chapter 2, where I develop the guiding example of *fragmented permutations* to illustrate the methods that will be used extensively throughout this chapter and the following one.

The previous method, while instructive, proved to be too rigid. It primarily allowed for the recovery of known probabilities through direct enumeration, without offering much flexibility for extension or deeper analysis.

This limitation motivates the introduction of more powerful tools from analytic combinatorics. This chapter is devoted to encoding the problem with generating functions and to presenting the main asymptotic results. Chapter 5 then introduces the multivariate analytic theorems from complex analysis that provide the theoretical foundation for these results.

4.1.1 Motivation

As mentioned in the introduction of this thesis, a long-standing tradition in social choice theory is to study the probability that a *Condorcet winner* exists, *i.e.* a candidate preferred to every other candidate by a majority of voters. A comprehensive overview of this research is presented in the book of Gehrlein [Geh06], which is entirely dedicated to this topic. Niemi and Weisberg [NW68] derived the limit of this probability as the number of voters tends to infinity, under the only assumption of independent preferences. More recently, Krishnamoorthy and Raghavachari [KR05] revisited and modernized this result.

Beyond limiting probabilities, however, it would be valuable to understand convergence rates. For instance, in the context of epistemic democracy, one seeks to recover a hidden ground truth by aggregating the votes of agents who each receive noisy information. Knowing how fast the

probability of electing the correct candidate converges to 1, or how fast the probability of a wrong outcome vanishes, provides a principled way to estimate the epistemic error as a function of the number of voters, or to determine how many voters are needed to meet a desired confidence level.

To achieve this objective, we propose a methodological shift. To the best of our knowledge, all previous studies on this question have relied on classical combinatorial or probabilistic techniques. For instance, the results of Niemi and Weisberg [NW68] and Krishnamoorthy and Raghavachari [KR05] are primarily based on Gaussian approximations. In contrast, we introduce tools from analytic combinatorics into this domain, drawing on the approach outlined in the reference book by Flajolet and Sedgewick [FS09].

4.1.2 Contributions

This chapter and the next one are based on a joint work with François Durand, Élie de Panafieu, Marc Noy and Vlady Ravelomanana [Cai+25]. Since our contributions are distributed across both chapters, we summarise here those related to social choice, while the multivariate analytic aspects are presented in Chapter 5.

- Similarly to Niemi and Weisberg [NW68] and Krishnamoorthy and Raghavachari [KR05], we consider a general probability distribution over voter preferences which we call *General Independent Culture (GIC)*, with the primary assumption that individual preferences are independent. Additionally, we assume that the distribution is *generic*, in the sense that every possible ranking over the candidates has a positive probability.
- To obtain results as general as possible, we introduce the notion of an α -winner, a generalization of the Condorcet winner, where the required victory threshold for each pairwise comparison is not necessarily one-half of the voters.
- We encode the problem via the symbolic method, deriving an adequate multivariate generating function. Using this encoding, we analyze the probability that a fixed candidate is an α -winner, with emphasis on its asymptotic behavior as the number of voters n grows, while the number of candidates m remains fixed.
- We illustrate our theoretical results through applications to the Impartial Culture and Mallows model. In particular, Élie and I carried out the asymptotic calculations for Impartial Culture, while François and I jointly derived the calculations for the Mallows model.

- These computations allow us to recover classical limiting probabilities while also providing finer asymptotic expansions, including convergence rates and higher-order terms. In particular, we establish that the probability of a given candidate being the Condorcet winner in Impartial Culture is $a_0 + a_{1,n}n^{-1/2} + \mathcal{O}(n^{-1})$, where we explicitly provide the values of the constant a_0 and the coefficient $a_{1,n}$, which depends solely on the parity of the number of voters n .
- Although we focus primarily on the Mallows model and Impartial Culture, our theoretical results apply directly to a broad class of probabilistic models, including the Perturbed Culture [WS67; Geh06], the Plackett-Luce model [Pla75; Luc59], Thurstonian models [Thu27], and, more generally, random utility models (RUMs) [APX12]. While we do not develop these applications in detail in this thesis, they fall within the scope of our framework.
- François Durand implemented the Python package `Actinvoting` [Dur25], which we used for symbolic mathematics and numerical simulations.

Beyond the specific problem we study, our approach opens new perspectives for the analysis of other problems in social choice theory. The techniques we develop offer a blueprint for encoding such problems into an analytic framework, with the explicit goal of serving as a tutorial for researchers interested in applying analytic combinatorics in this domain. We highlight some of the possible directions in the conclusion.

4.1.3 Related Work

Regarding the probability of a Condorcet winner, the standard reference is Gehrlein [Geh06]. In the General Independent Culture (GIC), which assumes independent voter preferences, the cases for finite values of voters n and candidates m were explored by Gehrlein and Fishburn [GF76b], Gillett [Gil78], and Gehrlein [Geh83]. The most relevant works to ours are the papers by Niemi and Weisberg [NW68] and Krishnamoorthy and Raghavachari [KR05], which derive the limiting probability as n tends to infinity in this general setting. Our approach differs in methodology and results: we use analytic combinatorics and provide more precise asymptotic behaviors.

Among the specific cases of GIC, the most studied is Impartial Culture (IC), where all rankings are equally likely [Geh06]. For finite values of n and m or for the asymptotic regime where $m \rightarrow \infty$, see the paper by Sauermann [Sau22]. Other works, like ours, focus on large electorates (*i.e.* $n \rightarrow \infty$). An explicit formula exists for $m = 3$ [Gui52; NW68; GK68], while the limit for $m = 4$ follows from a recurrence relation [May71; Fis73]. Explicit formulas for $m \in \{5, 6, 7\}$

have been derived by Gehrlein and Fishburn [GF78] and Gehrlein [Geh83]. The general case of arbitrary m with $n \rightarrow \infty$ was addressed by Niemi and Weisberg [NW68] and Krishnamoorthy and Raghavachari [KR05], whose results we recall in Theorem 8. In IC, as in the broader GIC framework, our main contribution is to provide the rate of convergence and a method for computing higher-order terms in the asymptotic expansion.

Other specific cases of GIC have been studied, notably the *Dual Culture* [Geh06] and the *Perturbed Culture* [WS67; Geh06]. Although the *Mallows model* [Mal57] is popular, the probability of a Condorcet winner in this framework has received little attention. This is likely because, for large electorates, the limiting behavior is trivial: the candidate favored by the culture becomes the Condorcet winner with probability tending to 1. The question of the convergence rate, which we explore here, is much more interesting.

More distant from our work, other models of culture fall outside the GIC framework as they do not assume voter independence. Notable examples include *Impartial Anonymous Culture (IAC)* [GF76a; Lep89; Geh06] and, more generally, *Pólya-Eggenberger models* [BL92; Geh06].

One can also study the notion of a *Weak Condorcet winner*, a candidate who wins or ties in every pairwise comparison. Krishnamoorthy and Raghavachari [KR05] note that the probability of a tie in a pairwise comparison tends to zero as the electorate grows, implying that the limiting probability of a Weak Condorcet winner is the same as for a Condorcet winner. However, the convergence speeds may differ, as discussed in our Section 4.5.2. Other works examine the probability that the *pairwise majority relation* is transitive [Geh06; Kel74; Bel78; Bel81]. While this could be addressed with analytic combinatorics, it is beyond the scope of this thesis.

Our analysis is based on representing voting configurations by polynomials, expressing probabilities via coefficient extraction, and deriving asymptotics using the saddle point method. In Impartial Anonymous Culture, many voting questions [Geh02; HC00; LLS08; WP07] have been studied using Ehrhart polynomials [Bar94], reducing the enumeration of voting configurations to counting integral points in a polyhedron. Maassen and Bezembinder [MB02] use generating functions to express the probability of a Condorcet winner in Impartial Culture or its variant with weak orders, obtaining exact expressions and asymptotics for several cases. However, none of these works apply complex analysis to derive asymptotics. May [May71] uses the saddle point method like us, but to study the asymptotics as $m \rightarrow \infty$ of the limit as $n \rightarrow \infty$ of the probability that a candidate is the Condorcet winner in Impartial Culture. This double limit problem is not addressed in this thesis and leads to very different computations, involving only a univariate real integral.

4.1.4 Limitations

A limitation of our work is that we focus on the probability that a *given* candidate is an α -winner, rather than the probability that at least one exists. For certain values of α , multiple candidates may satisfy this condition simultaneously, requiring inclusion-exclusion techniques to compute the probability of existence. We exclude this issue from the scope of this thesis, as it does not arise for the Condorcet winner, who, if existent, is unique; thus, its probability is simply the sum of individual probabilities.

Moreover, our main results concern the asymptotic behavior as n tends to infinity. While our approach also provides exact expressions for finite n , these expressions are essentially the same as those obtained using classical combinatorial methods. The main advantage of our method, therefore, lies in its ability to analyze asymptotic behavior. Indeed, although the expressions we obtain often involve complex integrals that cannot be exactly solved, they remain amenable to efficient numerical evaluation. This contrasts with exact combinatorial computations, which tend to become intractable as n increases—see, for instance, the discussion in Section 4.7.2 about Figure 4.5.

We assume that voters have independent preferences, excluding cases with more complex interactions. However, De Finetti’s theorem [DF29; HS55] asserts that *exchangeable* random variables (i.e., roughly speaking, symmetric ones) are conditionally independent given a latent variable. In our case, the preferences of voters are exchangeable up to random relabeling. Therefore, the problem can, in principle, be reduced to independent preferences by conditioning on the latent variable and integrating over its possible values.

To avoid degenerate cases, we also assume that the preference distribution is *generic*, meaning that all rankings have positive probability. While analytic combinatorics methods could be applied without this assumption, doing so would introduce additional subcases that we prefer to avoid, where the *saddle point* (which is defined in Chapter 5 in the univariate case, and which we will define shortly in the multivariate case) has null or infinite coordinates.

Finally, in some situations, our method requires computing a sum that, in the worst case, may contain up to $\mathcal{O}(2^m)$ terms (cf. Section 4.8). However, it is worth noting that merely specifying the full probability distribution already requires $m!$ real numbers in general.

4.1.5 Roadmap

Section 4.2 presents the necessary preliminaries. Section 4.3 translates our combinatorial problem into an analytic question and introduces the notion of *saddle point*. Depending on the coordinates of the saddle point, the analysis then branches into different cases, detailed in Sections 4.5 to 4.8. Section 4.9 extends the analysis by showing how to derive higher-order terms in the asymptotic expansion. Section 4.10 concludes. Chapter 5 establishes the technical results of analytic combinatorics used throughout this chapter.

4.2 Preliminaries

In this section, we begin by introducing the main definitions and notations, and we formulate our research question.

Recall that the number of voters is denoted by n . We define the set of candidates as $\{1, \dots, m\}$, where m denotes the number of candidates. We assume that voters have independent preferences, meaning the culture is characterized by a probability p_r for each preference ranking r . We refer to this terminology as a *General Independent Culture (GIC)*. We also assume that the culture is *generic*, in the sense that $p_r > 0$ for every ranking r . The notation \mathbb{P} refers to the probability under the given culture.

A *Condorcet winner* is a candidate who is preferred to every other candidate by more than $\frac{n}{2}$ voters. Without loss of generality, we study the probability $\mathbb{P}(m \text{ is CW})$ that candidate m is the Condorcet winner.

Our theoretical results extend this question to a more general notion, which we call an α -winner, defined as follows. Let $\mathcal{A} := \{1, \dots, m-1\}$ be the set of adversaries of candidate m . Consider a vector $\alpha := (\alpha_1, \dots, \alpha_{m-1}) \in (0,1)^{m-1}$, where each α_j represents the proportion of voters that candidate m needs on their side to win the pairwise comparison against candidate j .

For an adversary $j \in \mathcal{A}$, we write $m \succ_{\alpha} j$, and we say that m wins against j in the sense of α , if candidate m is preferred to j by more than $\alpha_j n$ voters. For a subset of adversaries $\mathcal{X} = \{j_1, \dots, j_k\} \subseteq \mathcal{A}$, we write $m \succ_{\alpha} \mathcal{X}$, or equivalently $m \succ_{\alpha} j_1, \dots, j_k$, if $m \succ_{\alpha} j$ holds for every $j \in \mathcal{X}$. Similarly, we write $m \preceq_{\alpha} j$, and we say that m does not win against j in the sense of α , if candidate m is preferred to j by at most $\alpha_j n$ voters.

Thus, we say that candidate m is an α -winner if $m \succ_{\alpha} \mathcal{A}$, i.e. m is preferred to each adversary j by more than $\alpha_j n$ voters. The standard notion of a Condorcet winner corresponds to the special case $\alpha = \left(\frac{1}{2}, \dots, \frac{1}{2}\right)$.

The notion of a *generalized Condorcet winner*, as introduced by Sertel and Sanver [SS04], is recovered by considering a vector α whose coordinates are equal.

The goal of this chapter is to study $\mathbb{P}(m \text{ is } \alpha\text{-winner})$, the probability that candidate m is an α -winner, with a particular focus on its asymptotic behavior as the number of voters n tends to infinity. Note that the probability of the existence of a Condorcet winner can be obtained as the sum of the probabilities for all m candidates.

Our theoretical results will be illustrated using the *Mallows model* [Mal57]. This model is characterized by a reference ranking r_0 over the candidates and a concentration parameter $\rho \in \mathbb{R}_{\geq 0}$. The probability of a ranking r is given by

$$p_r := \gamma e^{-\rho d(r, r_0)},$$

where γ is a normalization constant ensuring $\sum_r p_r = 1$, and $d(r, r_0)$ denotes the Kendall-tau distance [Ken38], which counts the minimum number of adjacent swaps needed to transform r into r_0 . The Mallows model is commonly used in the field of *epistemic democracy* [Bra+16, Chapters 8 and 10], where noisy evaluations from voters are collected with the aim of uncovering a hidden truth. The reference ranking models the hidden truth, and the concentration parameter ρ indicates the skill level of the voters. When $\rho = 0$, all rankings are equally probable, recovering the classical *Impartial Culture* model. We denote by $\mathcal{M}_{m \text{ last}}$ (resp. $\mathcal{M}_{m \text{ first}}$) a Mallows culture with parameter ρ , where the reference ranking r_0 places candidate m last (resp. first). We can assume, without loss of generality, that the reference ranking is $(1, \dots, m)$ (resp. $(m, \dots, 1)$).

Finally, as in the rest of the thesis, we typeset vectors in boldface, e.g. $\mathbf{x} := (x_1, \dots, x_{m-1})$. If \mathcal{X} is a set of indices, we define $\mathbf{x}_{\mathcal{X}} := (x_j)_{j \in \mathcal{X}}$. We define $\log(\mathbf{x}) := (\log(x_1), \dots, \log(x_{m-1}))$, and similarly for the exponential. We write $\mathbf{u} \leq \mathbf{v}$ if $u_j \leq v_j$ for every coordinate j , with similar notation for strict inequalities. We denote $\mathbf{0} := (0, \dots, 0)$ and $\mathbf{1} := (1, \dots, 1)$, where the vector's size is understood from context. The diagonal matrix with diagonal elements u_1, u_2, \dots is denoted $\text{diag}(\mathbf{u})$. Finally, for convenience, we set $\beta := \mathbf{1} - \alpha$, where α is the vector of victory thresholds in the definition of an α -winner.

4.3 From our Combinatorial Problem to its Analytic Formulation

As for fragmented permutations, we first encode the problem of determining the probability that a candidate is an α -winner. However, instead of a univariate infinite series, we now work with a multivariate polynomial. We then show that the coefficient extractions can be achieved through *multivariate Cauchy integrals*. Finally, to analyze their asymptotic behavior for large electorates, we introduce the general principle of the *saddle point method* in the multivariate setting. Its different sub-cases are later developed in Sections 4.5 to 4.8.

4.3.1 Symbolic Method

As discussed in Chapter 2, the first step of the symbolic method is to identify the relevant parameter(s) of the problem. In the case of fragmented permutations, the natural parameter was the number of elements in the permutations. Let us determine the appropriate parameter, restricting attention to the case $m = 3$ for the sake of clarity.

To determine whether candidate 3 is the Condorcet winner, or more generally an α -winner, it suffices to know whether each voter ranks candidate 1 and/or 2 above candidate 3, rather than their full ranking. Consequently, the relevant parameters are the scores of the pairwise comparisons between 1 and 3, and between 2 and 3. Therefore, we introduce two formal variables:

- x_1 indicates when a voter prefers candidate 1 over candidate 3,
- x_2 indicates when a voter prefers candidate 2 over candidate 3.

With the formal variables in hand, let us construct the generating function for a single voter. To represent the probability distribution governing the preferences of a single voter, we introduce the characteristic polynomial $P(x_1, x_2)$, as illustrated in Figure 4.1. In this polynomial, the probability of each ranking is multiplied by the appropriate formal variables, depending on whether candidate 1 and/or 2 is preferred over candidate 3. For instance, in the ranking (1, 2, 3), both 1 and 2 are preferred to 3, so we associate to this ranking the monomial x_1x_2 , whereas in the ranking (1, 3, 2), only 1 is preferred to 3 so we associate the monomial x_1 . When no candidates are preferred to 3, as in 3, 2, 1, we associate 1.

At this stage, the characteristic polynomial may appear to be merely a compact way of representing the probability distribution of a single voter.

p_{123}	p_{132}	p_{213}	p_{231}	p_{312}	p_{321}
1 2 $\boxed{3}$	1 $\boxed{3}$ 2	2 1 $\boxed{3}$	2 $\boxed{3}$ 1	$\boxed{3}$ 1 2	$\boxed{3}$ 2 1
x_1x_2	x_1	x_1x_2	x_2	1	1

$$P(x_1, x_2) := p_{123} \cdot x_1x_2 + p_{132} \cdot x_1 + p_{213} \cdot x_1x_2 + p_{231} \cdot x_2 + p_{312} \cdot 1 + p_{321} \cdot 1$$

Fig. 4.1: Definition of the characteristic polynomial $P(x_1, x_2)$ encoding the probability distribution for a single voter when $m = 3$. The notation p_{123} , for example, is a shorthand for $p_{(1,2,3)}$. The formal variable x_j marks rankings where candidate j is preferred to candidate 3.

Note that the characteristic polynomial can be rewritten as

$$P(x_1, x_2) = p_\emptyset + p_{\{1\}}x_1 + p_{\{2\}}x_2 + p_{\{1,2\}}x_1x_2, \quad (4.1)$$

where, for example, $p_{\{1,2\}} := p_{123} + p_{213}$ is the probability that a voter ranks both candidates 1 and 2 above candidate 3.

Introducing this formal polynomial makes it possible to express the probabilities via multivariate *coefficient extractions*. This concept has already been encountered; however, we recall the unlabeled case here for convenience. The n -th coefficient of a univariate unlabeled formal power series is written as

$$[z^n] \sum_k a_k z^k = a_n.$$

This notation extends naturally to the multivariate case:

$$[z^n] \sum_k a_k z^k = a_n.$$

Therefore, the probability $p_{\{1,2\}} := p_{123} + p_{213}$ can also be expressed as the multivariate coefficient extraction $[x_1^1 x_2^1] P(x_1, x_2)$.

In the general case, *i.e.* for an arbitrary value of m and a subset of adversaries $\mathcal{X} \subseteq \mathcal{A}$, we overload the notation p by defining $p_{\mathcal{X}}$ as the total probability that in a random ranking, the set of adversaries above m is exactly \mathcal{X} . We then encode the probability distribution for a single voter as follows.

Definition 8. The *characteristic polynomial* is defined as

$$P(\mathbf{x}) := \sum_{\mathcal{X} \subseteq \mathcal{A}} \left(p_{\mathcal{X}} \prod_{j \in \mathcal{X}} x_j \right).$$

In essence, the characteristic polynomial is the *generating function* that encodes the vote of a single voter. Also, note that $\sum_r p_r = 1$ implies that $P(\mathbf{1}) = 1$.

Now, returning to the case $m = 3$, let us examine what happens with several voters, beginning with $n = 2$ voters. The core observation, illustrated in Figure 4.2, is that the algebraic operation of developing $P(x_1, x_2)^2$ is isomorphic to the probability tree associated with the preferences of two voters. For example, the probability that, among the two voters, exactly two prefer candidate 1 over candidate 3, and exactly one prefers candidate 2 over candidate 3 is given by the coefficient extraction $[x_1^2 x_2^1]P(x_1, x_2)^2 = 2p_{\{1\}}p_{\{1,2\}}$, which coincides with the computation obtained from the probability tree.

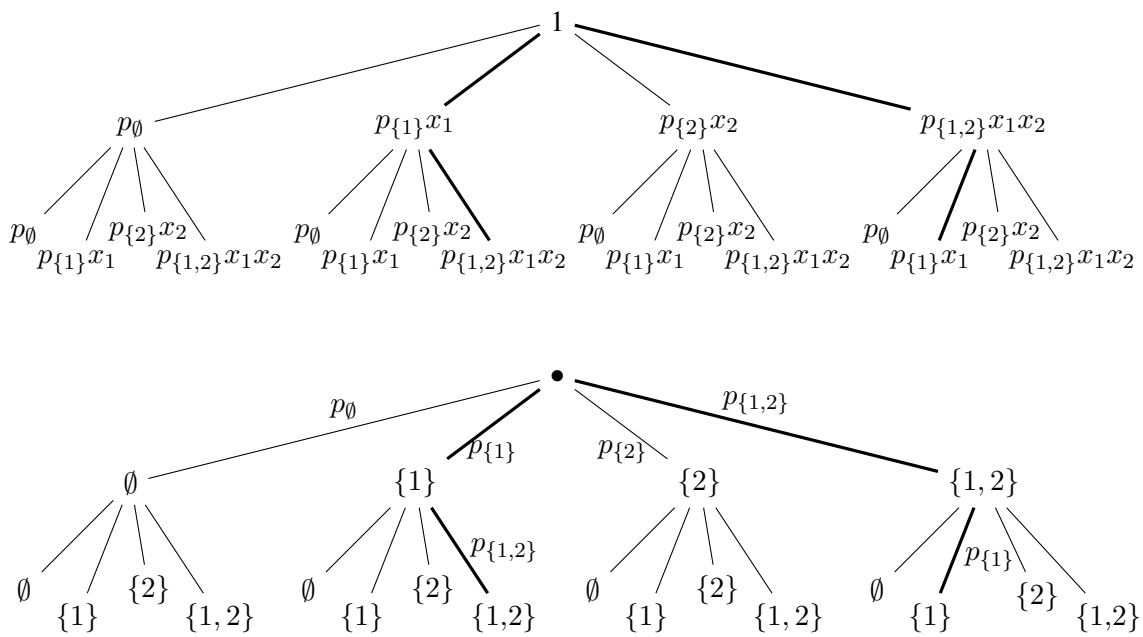


Fig. 4.2: Top: Tree representing the algebraic expansion of $P(x_1, x_2)^2$. Edges correspond to multiplications, and each path from the root to a leaf represents a term in the expansion. The highlighted paths correspond to the terms involving $x_1^2 x_2^1$. **Bottom:** The probability tree for two voters choosing a ranking. Some second-stage probabilities are omitted for readability. The highlighted path corresponds to the probability that, among the two voters, exactly two prefer candidate 1 over candidate 3, and exactly one prefers candidate 2 over candidate 3.

More generally, when m and n are arbitrary, for a vector $\ell \in \mathbb{N}^{m-1}$, the coefficient extraction

$$[\mathbf{x}^\ell]P(\mathbf{x})^n := [x_1^{\ell_1} \cdots x_{m-1}^{\ell_{m-1}}]P(x_1, \dots, x_{m-1})^n$$

corresponds to the probability that each adversary j is preferred to candidate m by exactly ℓ_j voters.

Now, for m to be an α -winner, every adversary j must be preferred to m by less than $(1 - \alpha_j)n = \beta_j n$ voters. Summing over all such cases, the probability that candidate m is an α -winner is given by

$$\mathbb{P}(m \text{ is } \alpha\text{-winner}) = [\mathbf{x}^{<\beta n}]P(\mathbf{x})^n := \sum_{\ell < \beta n} [\mathbf{x}^\ell]P(\mathbf{x})^n.$$

However, the coordinates of βn are not necessarily integers, and we prefer to express the summation bounds in terms of integers:

$$\mathbb{P}(m \text{ is } \alpha\text{-winner}) = [\mathbf{x}^{\leq \lceil \beta n \rceil - 1}]P(\mathbf{x})^n := \sum_{\ell \leq \lceil \beta n \rceil - 1} [\mathbf{x}^\ell]P(\mathbf{x})^n. \quad (4.2)$$

Note that if we define a *weak α -winner* as a candidate preferred to any other candidate j by *at least* $\alpha_j n$ voters, then all the results of this chapter extend to weak α -winners by replacing $\lceil \beta n \rceil - 1$ with $\lfloor \beta n \rfloor$. In particular, our findings for the Condorcet winner naturally extend to the *weak Condorcet winner* (defined analogously) by replacing all occurrences of $\lceil n/2 \rceil - 1$ with $\lfloor n/2 \rfloor$.

4.3.2 Coefficient Extraction via Cauchy Integrals

As explained in Chapter 2, in analytic combinatorics, it is common to extract the coefficient f_ℓ of a series $F(z) = \sum_k f_k z^k$ with a positive radius of convergence by expressing it as a Cauchy integral [FS09, Th. IV.4]:

$$[z^\ell]F(z) = \frac{1}{2i\pi} \oint F(z) \frac{dz}{z^{\ell+1}}. \quad (4.3)$$

The symbol \oint indicates that the integral is taken over a positively oriented loop around 0 in the complex plane. To intuitively understand this equality, one can expand $F(z)$ into its series, interchange sum and integral, and integrate over a circle of small radius centered at 0. The residue theorem then implies that all terms of the form $z^{k-\ell-1}$ result in a zero integral unless $k = \ell$. For a detailed proof, we refer the reader to Chapter 2.

Now, to extract $[z^{\leq L}]F(z) := f_0 + \dots + f_L$ we use the equality

$$[z^{\leq L}]F(z) = [z^L] \frac{F(z)}{1-z}, \quad (4.4)$$

which is obtained by remarking that $\frac{F(z)}{1-z} = \left(\sum_k f_k z^k\right) \left(\sum_k z^k\right) = \sum_k \left(\sum_{\ell=0}^k f_\ell\right) z^k$.

Equations (4.3) and (4.4) extend to the multivariate case. Applying them to Equation (4.2) gives

$$\mathbb{P}(m \text{ is } \alpha\text{-winner}) = \frac{1}{(2i\pi)^{m-1}} \oint \frac{P(\mathbf{x})^n}{\prod_{j=1}^{m-1} (1-x_j)} \frac{d\mathbf{x}}{\prod_{j=1}^{m-1} x_j^{\lceil \beta_j n \rceil}}, \quad (4.5)$$

where $\oint d\mathbf{x}$ is a shorthand for $\oint \cdots \oint dx_1 \dots dx_{m-1}$.

4.3.3 Saddle Point Method

We now investigate the asymptotic behavior of the complex integrals as $n \rightarrow +\infty$. To this end, we apply the *saddle point method*. Since this method has already been presented in detail in Chapter 2, I only recall here its main idea, specialized to our problem.

To build intuition and recall gently about the saddle point method, let us momentarily disregard the ceiling function in Equation (4.5) and observe that

$$\oint \frac{P(\mathbf{x})^n}{\prod_{j=1}^{m-1} (1-x_j)} \frac{d\mathbf{x}}{\prod_{j=1}^{m-1} x_j^{\beta_j n}} = \oint \frac{e^{-n(-\log(P(\mathbf{x})) + \beta^\top \log(\mathbf{x}))}}{\prod_{j=1}^{m-1} (1-x_j)} d\mathbf{x} = \oint A(\mathbf{x}) e^{-n\psi(\log(\mathbf{x}))} d\mathbf{x},$$

for an appropriately defined function $A(\mathbf{x})$ and $\psi(\mathbf{t}) = -\log(P(e^{\mathbf{t}})) + \beta^\top \mathbf{t}$, where $\mathbf{t} = \log(\mathbf{x})$ and β^\top denotes the transpose of β .

To analyze the asymptotic behavior of such integrals, the key idea is to find a contour of integration where, as n approaches infinity, the integral's dominant contribution arises from the neighborhood of a specific point, while the contribution from the rest of the contour becomes negligible in comparison.

For this approach to be valid, the chosen point must satisfy the condition that the gradient of the function ψ vanishes. In the univariate case, since a holomorphic function cannot exhibit local extrema in modulus, the graph of the function's modulus at such a point resembles a saddle. This is why it is called a *saddle point*, even in the general multivariate case.

In our case, the presence of the integer parts $\lceil \beta_j n \rceil$ does not fundamentally alter the method. The necessary adaptations are detailed in Chapter 5. Moreover, Lemma 8 in Section 4.4 ensures the existence of a unique saddle point. The absolute value of $e^{-n\psi(\log(\mathbf{x}))}$ at the saddle point is minimal on the real line and maximal on the integration circle. These considerations lead to define the following objects, which we will use extensively.

Definition 9. The *cumulant generating function* of P is

$$K : \mathbf{t} \in \mathbb{R}^{m-1} \mapsto \log \left(P(e^{\mathbf{t}}) \right).$$

Observing that the function $\psi : \mathbf{t} \mapsto -K(\mathbf{t}) + \beta^\top \mathbf{t}$ is strictly concave on \mathbb{R}^{m-1} (an immediate consequence of Lemma 14 in Chapter 5), we define the *log saddle point* $\boldsymbol{\tau}$ and the *saddle point* $\boldsymbol{\zeta}$ as

$$\boldsymbol{\tau} := \arg \max_{\mathbf{t} \in \mathbb{R}^{m-1}} \left(-K(\mathbf{t}) + \beta^\top \mathbf{t} \right), \quad \boldsymbol{\zeta} = e^{\boldsymbol{\tau}}.$$

Remark that this implies that $\boldsymbol{\zeta}$ minimizes the function $\mathbf{x} \in (\mathbb{R}_{>0})^{m-1} \mapsto \frac{P(\mathbf{x})}{\prod_j x_j^{\beta_j}}$.

To approximate the integral in Equation (4.5), a key ingredient is the Taylor expansion of ψ at its saddle point. In the univariate case, this implied to calculate the second derivative of the function of interest. Here, we will need the Hessian of ψ at $\boldsymbol{\tau}$. Since the term $\beta^\top \mathbf{t}$ is linear, this Hessian simplifies to $\mathcal{H}_K(\boldsymbol{\tau})$, the Hessian of $K(\mathbf{t})$ at $\boldsymbol{\tau}$, which can be computed either directly or as

$$\mathcal{H}_K(\boldsymbol{\tau}) = \text{diag}(\boldsymbol{\zeta}) \frac{\mathcal{H}_P(\boldsymbol{\zeta})}{P(\boldsymbol{\zeta})} \text{diag}(\boldsymbol{\zeta}) + \text{diag}(\boldsymbol{\beta}) - \boldsymbol{\beta} \boldsymbol{\beta}^\top. \quad (4.6)$$

Probabilistic interpretation. There is a nice probabilistic interpretation for the cumulant generating function. For any vector parameter $\mathbf{s} \in \mathbb{R}^{m-1}$, we define the random vector $\mathbf{X}^{(s)}$ as follows. For any subset $\mathcal{X} \subseteq \mathcal{A}$ of adversaries whose indicator vector is denoted $\boldsymbol{\ell} \in \{0,1\}^{m-1}$:

$$\mathbb{P}(\mathbf{X}^{(s)} = \boldsymbol{\ell}) := \frac{[\mathbf{x}^\ell] P(\mathbf{x}) e^{\boldsymbol{\ell}^\top \mathbf{s}}}{P(e^{\mathbf{s}})} = \frac{p_{\mathcal{X}} e^{\boldsymbol{\ell}^\top \mathbf{s}}}{P(e^{\mathbf{s}})}.$$

Note that $\mathbf{X}^{(0)}$ corresponds to the original probability distribution, as encoded by P . For $\mathbf{s} \neq \mathbf{0}$, the distribution of $\mathbf{X}^{(s)}$ represents a perturbation of the original culture.

The *cumulant generating function* of $\mathbf{X}^{(s)}$ is classically defined as

$$K_s : \mathbf{t} \mapsto \log(\mathbb{E}(e^{\mathbf{t}^\top \mathbf{X}^{(s)}})).$$

Remark that $K = K_0$. It is well known that the gradient and Hessian matrix of this function at $\mathbf{t} = \mathbf{0}$ are equal to the mean and covariance matrix of $\mathbf{X}^{(s)}$. Actually, all the information contained in K_s can be retrieved through K thanks to the following observation.

$$K_s(\mathbf{t}) = \log \left(\sum_{\boldsymbol{\ell}} \frac{([\mathbf{x}^\ell] P(\mathbf{x})) e^{\boldsymbol{\ell}^\top \mathbf{s}}}{P(e^{\mathbf{s}})} e^{\boldsymbol{\ell}^\top \mathbf{t}} \right) = \log \left(\sum_{\boldsymbol{\ell}} ([\mathbf{x}^\ell] P(\mathbf{x})) e^{\boldsymbol{\ell}^\top (\mathbf{s} + \mathbf{t})} \right) - \log(P(e^{\mathbf{s}})),$$

which simplifies to

$$K_s(\mathbf{t}) = \log(P(e^{s+\mathbf{t}})) - \log(P(e^s)) = K(\mathbf{s} + \mathbf{t}) - \log(P(e^s)),$$

so the mean and covariance matrix of $\mathbf{X}^{(s)}$ are respectively equal to the gradient and the Hessian of $K(\mathbf{t})$ at $\mathbf{t} = \mathbf{s}$.

Since $\boldsymbol{\tau}$ is the vector where the gradient of $\psi : \mathbf{t} \mapsto -K(\mathbf{t}) + \boldsymbol{\beta}^\top \mathbf{t}$ vanishes, it follows that the gradient of K at $\boldsymbol{\tau}$, *i.e.* the mean of $X^{(\boldsymbol{\tau})}$, is equal to $\boldsymbol{\beta}$. In other words, the distribution of $X^{(\boldsymbol{\tau})}$, *i.e.* the perturbation of the original culture induced by $\boldsymbol{\tau}$, is such that, in expectation, candidate m is precisely at the threshold for being an α -winner.

Criticality. Careful readers of Chapter 2 may recall that I mentioned how, when applying the saddle point method on generating functions arising from classical combinatorics problems, the integrand typically depends on n , and therefore the saddle point also varies with n . In our case, however, this phenomenon does not occur: since the characteristic polynomial is raised to the power n , the growth is exactly balanced by the division by z^n .

Nonetheless, depending on the values of α and \mathbf{p} , the saddle point ζ changes, and we must be cautious with the terms of the form $\frac{1}{1-x_j}$ in Equation (4.5).

- If $\zeta_j < 1$, there is no issue, as we can integrate over a circle of radius ζ_j that loops around 0 without enclosing the singularity at 1. In this case, we say that the coordinate ζ_j is *subcritical*.
- If ζ_j is *critical*, *i.e.*, if $\zeta_j = 1$, then the integration path must be adjusted to slightly bypass the singularity. In this case, we are in a situation similar to that of Chapter 2, where the saddle point depends on n .
- If ζ_j is *supercritical*, *i.e.*, if $\zeta_j > 1$, then a circle of radius ζ_j necessarily encloses a singularity, and the issue remains even with slight deformations of the integration path, requiring the singularity at 1 to be explicitly accounted for in the analysis.

In Section 4.5, we analyze the case where all coordinates are subcritical, while Section 4.6 focuses on the case where all coordinates are critical. Section 4.7 extends the analysis to the mixed case, where each coordinate is either subcritical or critical. To address supercritical coordinates, we introduce the necessary techniques in Section 4.8, which makes it possible to tackle the most general setting. Finally, Section 4.9 provides higher-order terms in the asymptotic expansion of the previous formulas. Together, these sections establish the theoretical asymptotic behavior of an α -winner in the GIC, followed by an application to the Condorcet winner in a particular culture.

4.4 Satisfying the Conditions of the Asymptotic Theorems in Chapter 5

The next three sections are devoted to applying results from multivariate complex analysis in order to derive asymptotics. Specifically, we rely on Theorems 17 to 19, which are stated and proved in Chapter 5, and which apply to generating functions that meet a number of structural conditions. To illustrate both the required conditions and the type of conclusions these theorems provide, we present here Theorem 17 in detail, without the proof. While the conditions to satisfy are identical for all three theorems, the specific asymptotic formulas they yield differ.

Theorem 17 [Multivariate large powers theorem] Consider a power series $B(z_1, \dots, z_d)$ that satisfies the following conditions.

- **Nonnegative coefficients:** all coefficients are nonnegative real numbers.
- **Full rank:** the rank of $B(z)$, defined as the dimension of the real vector space generated by the differences of exponent vectors of nonzero coefficients, equals the number of variables.
- **Aperiodicity:** $B(z)$ is aperiodic, i.e. its exponent vectors do not lie in a proper arithmetic sublattice of \mathbb{Z}^d .
- **Convergence near the origin:** $B(z)$ converges in some neighborhood Ω of $\mathbf{0}$.

Consider $K(\mathbf{t}) = \log(B(e^{\mathbf{t}}))$ its cumulant generating function, and let $\boldsymbol{\lambda}$ denote a vector of positive rational numbers such that the function $K(\mathbf{t}) - \boldsymbol{\lambda}^\top \mathbf{t}$ does not tend to its infimum on the boundary of $\Psi = \{\mathbf{t} \in \mathbb{R}^d, e^{\mathbf{t}} \in \Omega\}$. Let $\boldsymbol{\tau}$ denote the unique point of Ψ where $K(\mathbf{t}) - \boldsymbol{\lambda}^\top \mathbf{t}$ reaches its minimum and \mathcal{H} denote the Hessian matrix of $K(\mathbf{t})$ at $\boldsymbol{\tau}$. Set $\boldsymbol{\zeta} = e^{\boldsymbol{\tau}}$. Consider a power series $A(z)$ analytic on the torus of radius $\boldsymbol{\zeta}$ and satisfying $A(\boldsymbol{\zeta}) \neq 0$. Then, consider $\delta < 1/2$ and a sequence of vectors of real values

$$\boldsymbol{\kappa}_n = \boldsymbol{\lambda}n + \mathcal{O}(n^\delta).$$

We have

$$[\mathbf{z}^{\boldsymbol{\kappa}_n}]A(\mathbf{z})B(\mathbf{z})^n \sim \frac{A(\boldsymbol{\zeta})}{\sqrt{(2\pi n)^d \det(\mathcal{H})}} \frac{B(\boldsymbol{\zeta})^n}{\boldsymbol{\zeta}^{\boldsymbol{\kappa}_n}}.$$

Therefore, to apply these results to our case, we must show that the characteristic polynomial $P(\mathbf{x}) := \sum_{\mathcal{X} \subseteq \mathcal{A}} (p_{\mathcal{X}} \prod_{j \in \mathcal{X}} x_j)$ satisfies several assumptions, like the ones of Theorem 17. In addition, the cumulant generating function of $P(\mathbf{x})$ must not vanish at infinity, in other words, we require the saddle point to have all coordinates finite.

Let us then demonstrate that all the conditions are satisfied. The characteristic polynomial $P(\mathbf{x})$ encodes a probability distribution over the preference rankings of a voter. This guarantees its convergence on a neighborhood Ω of 0. Since we assume that the distribution of voter preferences over rankings is generic, *i.e.* every ranking r has a positive probability, it follows that all coefficients of the characteristic polynomial $P(\mathbf{x})$ are positive. The following result establishes that this assumption ensures that the remaining conditions required for the application of our asymptotic theorems are satisfied.

Lemma 8. *Consider a polynomial $P(x_1, \dots, x_d)$ where for any $\mathcal{X} \subseteq \{1, \dots, d\}$, the monomial $\prod_{j \in \mathcal{X}} x_j$ has a positive coefficient, while any other monomial has a zero coefficient. As usual, define its cumulant generating function as*

$$K : \mathbf{t} \mapsto \log(P(e^{\mathbf{t}})).$$

Then $P(\mathbf{x})$ has full rank and is aperiodic. Furthermore, for any $\boldsymbol{\lambda} \in (0, 1)^d$ and $\boldsymbol{\ell}$ on the boundary of $[-\infty, +\infty]^d$, the function $t \mapsto K(\mathbf{t}) - \boldsymbol{\lambda}^\top \mathbf{t}$ tends to infinity as \mathbf{t} tends to $\boldsymbol{\ell}$. Thus, there exists a unique vector $\boldsymbol{\tau} \in \mathbb{R}^d$ where the gradient of K vanishes.

Proof. By assumption, the support of $P(\mathbf{x})$ contains $\mathbf{0}$ and \mathbf{e}_j for each $j \in \{1, \dots, d\}$. Proposition 1 of Chapter 5 then implies that $P(\mathbf{x})$ has full rank and is aperiodic. The proof is given in detail in Chapter 5; we do not reproduce it here as it requires additional notation, but intuitively, the presence of $\mathbf{0}$ and all unit vectors in the support ensures that the polynomial spreads across every coordinate direction, which enforces full rank. At the same time, these elements prevent the exponents from being restricted to a rigid arithmetic pattern, thereby guaranteeing aperiodicity.

Now, consider \mathbf{t} converging to $\boldsymbol{\ell}$. Let \mathcal{X} denote the sets of indices j such that $\ell_j = +\infty$. Let $p_{\mathcal{X}}$ denote the coefficient of $P(\mathbf{x})$ corresponding to the monomial of exponents in \mathcal{X} . Let $\mathbf{1}_{\mathcal{X}}$ denote the vector of dimension d with value 1 at indices from \mathcal{X} , and 0 otherwise. Then

$$P(e^{\mathbf{t}}) \geq p_{\mathcal{X}} \exp(\mathbf{1}_{\mathcal{X}}^\top \mathbf{t}),$$

so

$$K(\mathbf{t}) - \boldsymbol{\lambda}^\top \mathbf{t} \geq \log(p_{\mathcal{X}}) + \mathbf{1}_{\mathcal{X}}^\top \mathbf{t} - \boldsymbol{\lambda}^\top \mathbf{t} \geq \log(p_{\mathcal{X}}) + (\mathbf{1}_{\mathcal{X}} - \boldsymbol{\lambda})^\top \mathbf{t}.$$

The vector $\mathbf{1}_{\mathcal{X}} - \boldsymbol{\lambda}$ has positive coefficients at the indices where \mathbf{t} tends to $+\infty$, and negative coefficients at the indices where \mathbf{t} tends to a constant or $-\infty$. Since $\boldsymbol{\ell}$ is on the boundary of $[-\infty, +\infty]^d$, it has at least one element equal to $-\infty$ or $+\infty$, so $(\mathbf{1}_{\mathcal{X}} - \boldsymbol{\lambda})^\top \mathbf{t}$ tends to infinity. This implies that $K(\mathbf{t}) - \boldsymbol{\lambda}^\top \mathbf{t}$ converges to $+\infty$ as well. Now, as K is strictly convex (see Lemma 14 in

Chapter 5 for the proof), if its minimum is not reached on the boundary of its domain of definition, it must be reached in its interior and is unique. Thus, there exists a unique point $\tau \in \mathbb{R}^d$ where the gradient of K vanishes. \square

In conclusion, our problem satisfies all the structural conditions required to apply the theorems from Chapter 5 and derive asymptotics. This is precisely the purpose of the next three sections.

4.5 Subcritical Case

In this section, we study the case where all coordinates of the saddle point are subcritical. We then illustrate this scenario with \mathcal{M}_3 last, a Mallows culture where the reference ranking is $(1, 2, 3)$, making candidate 3 particularly unlikely to be the Condorcet winner.

4.5.1 Theoretical Result in the Subcritical Case

Since $P(x)$ meets all the hypotheses of Theorem 17, we can now apply it. Recall that $P(x)$ is introduced in Definition 8, and that τ , ζ , and $\mathcal{H}_K(\tau)$ are introduced in Definition 9.

Theorem 4. *Assume that all coordinates of the saddle point are subcritical, i.e. $\zeta_j < 1$ for every adversary $j \in \mathcal{A}$. Then:*

$$\mathbb{P}(m \text{ is } \alpha\text{-winner}) \underset{n \rightarrow +\infty}{\sim} \frac{P(\zeta)^n}{\prod_{j \in \mathcal{A}} ((1 - \zeta_j) \zeta_j^{\lceil \beta_j n \rceil - 1})} \frac{1}{\sqrt{(2\pi n)^{m-1} \det(\mathcal{H}_K(\tau))}}.$$

Note that using Landau's notations, this yields

$$\mathbb{P}(m \text{ is } \alpha\text{-winner}) = \mathcal{O}\left(\left(\frac{P(\zeta)}{\prod_{j \in \mathcal{A}} \zeta_j^{\beta_j}}\right)^n \cdot \frac{1}{\sqrt{n^{m-1}}}\right), \quad \text{as } n \rightarrow +\infty.$$

To interpret this asymptotic behavior, recall that ζ is the unique global minimizer of $x \mapsto \frac{P(x)}{\prod_j x_j^{\beta_j}}$.

Hence, $\frac{P(\zeta)}{\prod_j \zeta_j^{\beta_j}} < \frac{P(\mathbf{1})}{\prod_j 1^{\beta_j}} = 1$. Thus, Theorem 4 indicates that, in the subcritical case, $\mathbb{P}(m \text{ is } \alpha\text{-winner})$ tends to 0 exponentially fast.

4.5.2 Application: Mallows Culture $\mathcal{M}_{3 \text{ last}}$

In the model $\mathcal{M}_{3 \text{ last}}$, defined in Section 4.2, candidate 3 is ranked last among the three options in the reference ranking (1,2,3), which can be interpreted as representing the hidden truth. We now analyze the probability of the *a priori* undesirable event in which candidate 3 nevertheless emerges as the Condorcet winner.

To find this probability, the first step is to determine the saddle point. We provide the general formula for an arbitrary number of candidates m .

Lemma 9. *Under the Mallows culture $\mathcal{M}_{m \text{ last}}$, the log saddle point τ is given by*

$$\tau = \left(\frac{-m\rho}{2}, \frac{(-m+2)\rho}{2}, \dots, \frac{(m-4)\rho}{2} \right),$$

where ρ is the concentration parameter of the culture, defined in Section 4.2.

Proof. To prove this result, we leverage the probabilistic interpretation of the saddle point (Section 4.3.3). In the distribution induced by the saddle point, candidate m is, in expectation, precisely at the threshold for being an α -winner.

In the Condorcet case, this means that for each candidate $j \neq m$, the probability of all rankings in which j appears above m must equal the probability of all rankings in which j appears below m , after reweighting each probability according to ζ . Concretely, for each candidate $j \neq m$

$$\sum_{r, j \succ_r m} p_r \prod_{k \succ_r m} \zeta_k = \sum_{r, m \succ_r j} p_r \prod_{k \succ_r m} \zeta_k,$$

where $j \succ_r \ell$ means that candidate j appears above candidate ℓ in the ranking r .

For the Mallows culture $\mathcal{M}_{m \text{ last}}$, the equilibrium condition can be formulated in terms of ranking pairs. Specifically, due to the structure of the Mallows distribution, a ranking of the form (σ, m, σ') balances with the ranking obtained by swapping σ and σ' , i.e. (σ', m, σ) .

For instance, when $m = 5$, the ranking (1, 3, 2, 5, 4) must be in equilibrium with (4, 5, 1, 3, 2), yielding

$$P_{(1,3,2,5,4)} \zeta_1 \zeta_2 \zeta_3 = P_{(4,5,1,3,2)} \zeta_4.$$

Therefore, for an arbitrary m , the saddle point satisfies $m!/2$ equilibrium equations. Indeed, for each suitable pair of rankings r and r' , the saddle point satisfies

$$p_r \prod_{k \succ_r m} \zeta_k = p_{r'} \prod_{k \succ_{r'} m} \zeta_k.$$

By considering the equation

$$p_{\{j\}} \zeta_j = p_{\mathcal{A} \setminus \{j\}} \prod_{k \neq j} \zeta_k$$

for each adversary j , we obtain a system of $m - 1$ independent equations with $m - 1$ unknowns (except when $m = 3$, where the system reduces to a single equation). Solving this system, while ensuring that all coordinates of the saddle point are nonzero, yields the desired result. In the case of $m = 3$, the system simplifies to $p_{\{1\}} \zeta_1 = p_{\{2\}} \zeta_2$. Adding the equation $p_{\{1,2\}} \zeta_1 \zeta_2 = p_\emptyset$ provides two independent equations, allowing for a unique solution. \square

In the case of three candidates, Lemma 9 yields $\tau = (\frac{-3\rho}{2}, \frac{-\rho}{2})$, ensuring that both coordinates of the saddle point are subcritical and allowing the application of Theorem 4. Standard algebraic calculations then lead to the following result.

Theorem 5. *Under the Mallows culture \mathcal{M}_3 last, the probability that candidate 3 is the Condorcet winner has asymptotic behavior*

$$\mathbb{P}(3 \text{ is CW}) \underset{n \rightarrow +\infty}{\sim} \frac{P(\zeta)^n}{\prod_{j \in \mathcal{A}} ((1 - \zeta_j) \zeta_j^{\lceil n/2 \rceil - 1})} \frac{1}{2\pi n \sqrt{\det(\mathcal{H}_K(\tau))}},$$

where $\zeta = (e^{-3\rho/2}, e^{-\rho/2})$, $P(\zeta) = 2 \frac{e^{-2\rho}(1+e^{-\rho/2}+e^{-\rho})}{(1+e^{-\rho})(1+e^{-\rho}+e^{-2\rho})}$ and $\det(\mathcal{H}_K(\tau)) = \frac{1}{4} \frac{e^{-\rho/2}(1+e^{-\rho})}{(1+e^{-\rho/2}+e^{-\rho})^2}$.

As we saw in Section 4.5.1, this probability tends exponentially fast to 0. This can be viewed as a generalization of Condorcet's Jury Theorem [Con85], which gives the same result for 2 candidates.

As mentioned in Section 4.3.1, the counterpart of Theorem 5 for the weak Condorcet winner is obtained by replacing $\lceil n/2 \rceil - 1$ with $\lfloor n/2 \rfloor$. Notably, for even n , the former simplifies to $n/2 - 1$, while the latter becomes $n/2$. Thus, although the limit remains 0 in both cases, the rates of convergence for even n differ by a factor $\prod_{j \in \mathcal{A}} \zeta_j$.

Numerical simulations To validate our results numerically, we developed the Python package `Actinvoting` [Dur25]. This package notably compares our theoretical results with Monte Carlo simulations, which estimate probabilities by generating a large number of random profiles. Ranking samples in the Mallows model are obtained using the algorithms of Doignon et al. [DPR04] and Lu and Boutilier [LB14].

Figure 4.3 illustrates the case of $\mathcal{M}_{3 \text{ last}}$, showing the probability that candidate 3 is the Condorcet winner as a function of the number of voters n . We provide three estimations: the theoretical approximation given by Theorem 5, the exact result obtained via coefficient extraction from Equation (4.2), and a Monte Carlo estimate based on 10,000 profiles per value of n . The concentration parameter is set to $\rho = \log(2)$. This choice balances two opposing constraints observed in practice. Lower concentration values require larger n for the theoretical approximation to be accurate, making direct comparison with exact results difficult, as their computational cost grows exponentially in n due to the algebraic expansion of $P(\mathbf{x})^n$. Conversely, higher concentration values cause probabilities to decay rapidly, rendering Monte Carlo estimation unreliable.

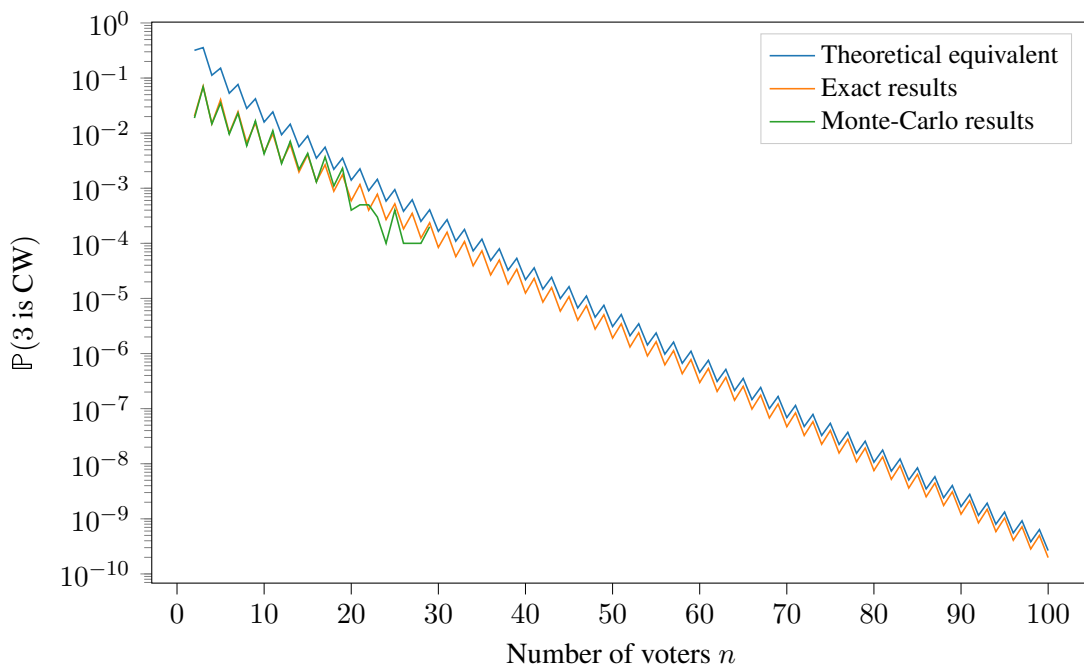


Fig. 4.3: Probability that candidate 3 is the Condorcet winner as a function of n in a culture $\mathcal{M}_{3 \text{ last}}$ with $\rho = \log(2)$, shown on a semilog scale. The theoretical equivalent is based on Theorem 5, while exact results rely on Equation (4.2). Monte Carlo simulations use 10,000 profiles per point, yielding an error of order 10^{-2} . For $n \geq 30$, they return zero, which is not visible due to the logarithmic scale.

The computational costs of these methods vary significantly. For instance¹, at $n = 100$, Monte Carlo simulations take 46 s, the exact computation 17 s, and the theoretical approximation only 338 μ s. As previously said, the exact computation cost grows exponentially with n . Monte Carlo simulations are also expensive, as their error scales as $\mathcal{O}(1/\sqrt{N})$, with N the sample size. In contrast, the theoretical approximation has complexity polynomial in $\log(n)$, making it essentially constant time in practice.

A striking feature of Figure 4.3 is the non-monotonicity of the curves, arising from parity effects in the number of voters n . For small n , the exact and Monte Carlo curves are nearly indistinguishable, while the theoretical approximation slightly overestimates the probability. When probabilities drop below the order of magnitude of the Monte Carlo error, $1/\sqrt{N} = 10^{-2}$, the Monte Carlo results become unstable, and below $1/N = 10^{-4}$ they are undetectable. As expected, for large n , the theoretical approximation converges to the exact result while being much faster computationally.

4.6 Critical Case

In this section, we study the case where all coordinates of the saddle point are critical, which is exemplified by the notion of Condorcet winner in the Impartial Culture.

4.6.1 Theoretical Result in the Critical Case

When $\zeta = \mathbf{1}$, it means that the expected proportion of voters who prefer candidate m to candidate j is exactly α_j , as explained in the probabilistic interpretation of Section 4.3.3. In other words, in expectation, candidate m is precisely at the threshold of being an α -winner.

As soon as one coordinate ζ_j is equal to 1, the term $\frac{1}{1-x_j}$ in Equation (4.5) prevents integration along a circle of radius ζ_j . Theorem 18 in Chapter 5 is based on the idea that this difficulty can be circumvented by choosing an integration path that slightly deforms around the singularity at 1. Applying this result to the particular case where all coordinates are critical, we obtain the following theorem.

Theorem 6. *Assume that all coordinates of the saddle point are critical, i.e. $\zeta_j = 1$ for every adversary $j \in \mathcal{A}$. Then:*

$$\lim_{n \rightarrow +\infty} \mathbb{P}(m \text{ is } \alpha\text{-winner}) = \frac{1}{\sqrt{(2\pi)^{m-1} \det(\mathcal{H}_K(\boldsymbol{\tau}))}} \int_{(0,+\infty)^{m-1}} e^{-\mathbf{u}^\top \mathcal{H}_K(\boldsymbol{\tau})^{-1} \mathbf{u}/2} d\mathbf{u}.$$

¹All of our numerical simulations were performed on an 11th Gen Intel Core i9-11980HK processor (8 cores, 16 logical processors) with 64 GB of RAM.

In this case, Equation (4.6) simplifies to $\mathcal{H}_K(\boldsymbol{\tau}) = \mathcal{H}_P(\mathbf{1}) + \text{diag}(\boldsymbol{\beta}) - \boldsymbol{\beta}\boldsymbol{\beta}^\top$ because we have $\boldsymbol{\zeta} = \mathbf{1} = P(\mathbf{1})$.

Note that this integral is a relatively simple Gaussian integral of dimension $m - 1$, which can be efficiently evaluated numerically. Thus, this theorem provides an explicit and practically computable value for the limit.

For the critical case, Theorem 6 generalizes to α -winners the result of Niemi and Weisberg [NW68] and Krishnamoorthy and Raghavachari [KR05, Equation (12)] on Condorcet winners. While we derive this result via analytic combinatorics, their approach relies on a standard probabilistic method: the Gaussian approximation. This method allows for interpreting the vector \mathbf{u} as the standardized outcome of each pairwise comparison between candidate m and their adversaries. Integrating over the positive orthant corresponds to requiring that each comparison outcome exceeds its expectation $\alpha_j n$.

The two approaches differ in the matrix involved. In their case, it is the correlation matrix of the pairwise comparisons against each adversary, denoted R_m by both Niemi and Weisberg [NW68] and Krishnamoorthy and Raghavachari [KR05]. Let us now recall its definition and establish the relation $R_m = 4D\mathcal{H}_K(\boldsymbol{\tau})D$, where D is a diagonal matrix, which is the identity matrix when considering the notion of Condorcet winner.

For each $j \in \mathcal{A}$, let X_j denote the random variable taking value 1 if candidate j is ranked higher than candidate m , and 0 otherwise. Set $\mathbf{Y} = 2\mathbf{X} - \mathbf{1}$. We consider the covariance matrix $\mathcal{H}_K(\boldsymbol{\tau})$ of \mathbf{X} , whose coefficient (j, k) is

$$\mathbb{E}(X_j X_k) - \mathbb{E}(X_j)\mathbb{E}(X_k),$$

while Niemi and Weisberg [NW68] and Krishnamoorthy and Raghavachari [KR05] consider the correlation matrix R_m of \mathbf{Y} , whose coefficient (j, k) is

$$\frac{\mathbb{E}(Y_j Y_k) - \mathbb{E}(Y_j)\mathbb{E}(Y_k)}{\sqrt{(1 - \mathbb{E}(Y_j)^2)(1 - \mathbb{E}(Y_k)^2)}}.$$

Observe that

$$\begin{aligned} \mathbb{E}(Y_j Y_k) - \mathbb{E}(Y_j)\mathbb{E}(Y_k) &= \mathbb{E}((2X_j - 1)(2X_k - 1)) - \mathbb{E}(2X_j - 1)\mathbb{E}(2X_k - 1) \\ &= 4(\mathbb{E}(X_j X_k) - \mathbb{E}(X_j)\mathbb{E}(X_k)). \end{aligned}$$

Let D denote the invertible diagonal matrix whose j th element is $(1 - \mathbb{E}(Y_j)^2)^{-1/2}$, then

$$R_m = 4D\mathcal{H}_K(\boldsymbol{\tau})D.$$

Thus, the change of variable $\mathbf{u} = 2D\mathbf{v}$ links the two formulas

$$\frac{1}{\sqrt{\det(R_m)}} \int_{(0,+\infty)^{m-1}} e^{-\mathbf{v}^\top R_m^{-1} \mathbf{v}/2} d\mathbf{v} = \frac{1}{\sqrt{\det(\mathcal{H}_K(\boldsymbol{\tau}))}} \int_{(0,+\infty)^{m-1}} e^{-\mathbf{u}^\top \mathcal{H}_K(\boldsymbol{\tau})^{-1} \mathbf{u}/2} d\mathbf{u}.$$

Therefore, we see that this rescaling has no impact. However, our analytic combinatorics approach makes it easier to deal with non-critical cases and extends naturally to the computation of an asymptotic expansion (see Section 4.9). For instance, in the subcritical case, their formula only establishes that the limit is zero, whereas Theorem 4 additionally provides the convergence rate.

4.6.2 Application: Impartial Culture

Computing the limiting probability for a candidate to be the Condorcet winner in Impartial Culture corresponds to applying Theorem 6 with $\beta = 1/2$. We obtain the following result.

Theorem 7. *Under Impartial Culture with m candidates, the probability that candidate m is the Condorcet winner has the following limit:*

$$\lim_{n \rightarrow +\infty} \mathbb{P}(m \text{ is CW}) = \frac{1}{\sqrt{(2\pi)^{m-1} \det(\mathcal{H}_K(\mathbf{0}))}} \int_{(0,+\infty)^{m-1}} e^{-\mathbf{u}^\top \mathcal{H}_K(\mathbf{0})^{-1} \mathbf{u}/2} d\mathbf{u}$$

where, letting I denote the identity matrix and J the matrix where all elements are 1s, both of dimension $(m-1) \times (m-1)$,

$$\mathcal{H}_K(\mathbf{0}) = \frac{1}{6} \left(I + \frac{1}{2} J \right), \quad \mathcal{H}_K(\mathbf{0})^{-1} = 6 \left(I - \frac{1}{m+1} J \right), \quad \det(\mathcal{H}_K(\mathbf{0})) = \frac{m+1}{2 \cdot 6^{m-1}}. \quad (4.7)$$

Proof. For each $j \in \mathcal{A}$, let X_j denote the random variable taking value 1 if candidate j is ranked higher than candidate m , and 0 otherwise. The probability generating function of the vector \mathbf{X} is $P(\mathbf{x})$ and its cumulant generating function $\mathbf{t} \mapsto \mathbb{E}(e^{\mathbf{t}^\top \mathbf{X}})$ is $K : \mathbf{t} \mapsto \log(P(e^{\mathbf{t}}))$. Thus, under Impartial Culture We have $\mathbb{E}(\mathbf{X}) = \mathbf{1}/2$.

The Hessian of K at $\mathbf{0}$ is the covariance matrix of \mathbf{X} , whose coefficient (j, k) is equal to

$$\mathcal{H}_K(\mathbf{0})_{j,k} = \mathbb{E}((X_j - 1/2)(X_k - 1/2)) = \mathbb{E}(X_j X_k) - \frac{1}{4}.$$

For $j = k$, we use that $X_j^2 = X_j$ since X_j is an indicator variable, hence $\mathbb{E}(X_j^2) = \mathbb{E}(X_j) = 1/2$. We deduce $\mathcal{H}_K(\mathbf{0})_{j,j} = 1/4$.

For $j \neq k$, note that in a random ranking r , the probability that each candidate from a given subset \mathcal{Y} of adversaries beats candidate m is $\frac{1}{|\mathcal{Y}|+1}$. Indeed, the ranking induced by r on $\mathcal{Y} \cup \{m\}$ is uniformly distributed, so the probability that m is in the last position is $\frac{1}{|\mathcal{Y}|+1}$. This implies that for $j \neq k$

$$\mathbb{E}(X_j X_k) = \frac{1}{2+1} = \frac{1}{3},$$

yielding

$$\mathcal{H}_K(\mathbf{0})_{j,k} = \frac{1}{3} - \frac{1}{4} = \frac{1}{12}.$$

Letting I denote the identity matrix of dimension $m-1$ and J the matrix containing only 1s, we deduce

$$\mathcal{H}_K(\mathbf{0}) = \frac{1}{12}J + \frac{1}{6}I.$$

The expressions of its inverse and determinant from Equation (4.7) follow from straightforward calculation. \square

We showed that this result is the same as Equation (10) from Niemi and Weisberg [NW68] and Equation (12) from Krishnamoorthy and Raghavachari [KR05]. For completeness, we recall a simpler univariate integral formulation introduced by Ruben [Rub54] to study moments of Gaussian distributions, and linked to the voting setting first implicitly by Niemi and Weisberg [NW68], then explicitly by May [May71] (Equation (7)).

Theorem 8 (May [May71]). *Under Impartial Culture,*

$$\lim_{n \rightarrow +\infty} \mathbb{P}(m \text{ is CW}) = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{+\infty} e^{-u^2} (1 - \Phi(u))^{m-1} du \quad (4.8)$$

where $\Phi(u) = \frac{1}{2} \left(1 + \operatorname{erf}\left(\frac{u}{\sqrt{2}}\right)\right)$ and erf denotes the error function: $\operatorname{erf}(u) = \frac{2}{\sqrt{\pi}} \int_0^u e^{-v^2} dv$.

Numerical simulations Figure 4.4 illustrates our results for the Impartial Culture with three candidates. Computational times vary significantly: for $n = 100$, Monte Carlo simulations take 5 seconds (faster than for the Mallows model due to quicker random profile generation in IC), the exact computation takes 22 seconds, and the theoretical approximation requires only 358 μs . As before, the Monte Carlo and exact result curves exhibit sawtooth patterns due to parity effects in the number of voters n and remain nearly indistinguishable. In contrast, the theoretical approximation,

which is just a numerical limit here, is simply represented as a horizontal line. Notably, for odd n , the exact values closely match the limit, whereas for even n , they are significantly lower, although they converge to the same value. In Section 4.9, we present the next term in the asymptotic expansion of the probability, which improves the fit between the theoretical approximation and the exact curve, and explains why the limit approximation is already very accurate for an odd number of voters n .

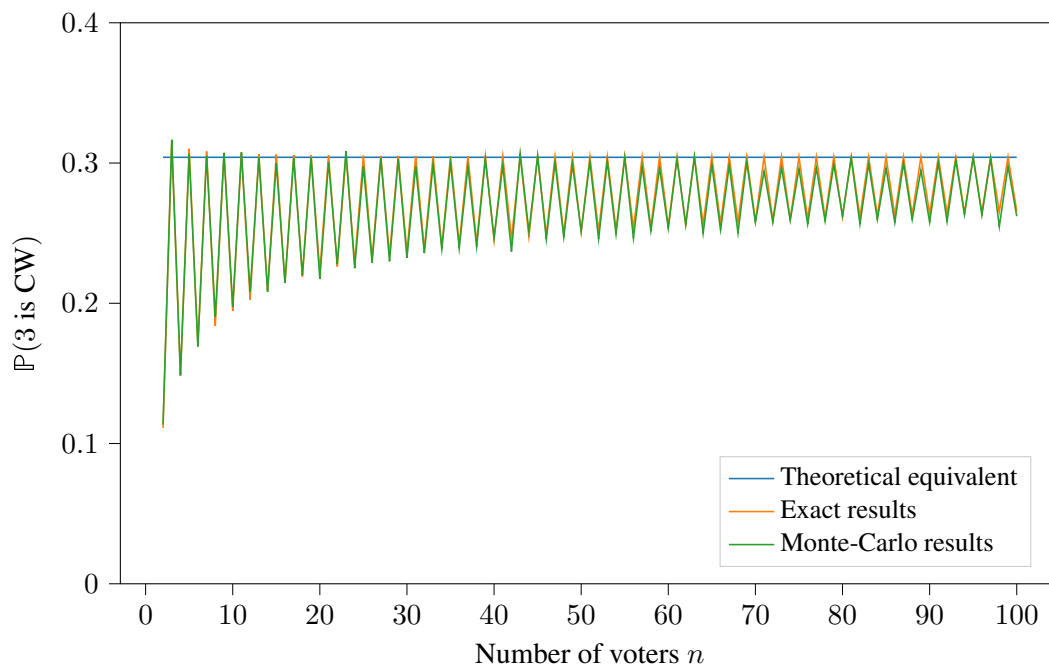


Fig. 4.4: Probability that a given candidate is the Condorcet winner as a function of n in the Impartial Culture with three candidates. The theoretical equivalent is based on Theorem 7, while exact results rely on Equation (4.2). Monte Carlo simulations use 10,000 profiles per point, yielding an error of order 10^{-2} . When the exact results curve is not visible, it is overlapped by the Monte Carlo results curve.

4.7 Mixed Case: Subcritical and Critical

We study the case where some of the coordinates are subcritical and some are critical, combining the ideas developed in Sections 4.5 and 4.6.

4.7.1 Theoretical Result in the Mixed Case

As in the critical case, we apply Theorem 18 from Chapter 5, now in its more general form.

Theorem 9. *Let $\mathcal{S} := \{j \in \mathcal{A}, \zeta_j < 1\}$ and $\mathcal{C} := \{j \in \mathcal{A}, \zeta_j = 1\}$ respectively denote the subset of subcritical and critical coordinates of the saddle point. Assume $\mathcal{A} = \mathcal{S} \cup \mathcal{C}$. Then:*

$$\mathbb{P}(m \text{ is } \alpha\text{-winner}) \underset{n \rightarrow +\infty}{\sim} \frac{P(\zeta)^n}{\prod_{j \in \mathcal{S}} ((1 - \zeta_j) \zeta_j^{\lceil \beta_j n \rceil - 1})} \times \frac{1}{\sqrt{(2\pi)^{m-1} n^{|\mathcal{S}|} \det(\mathcal{H}_K(\tau))}} \\ \times \int_{(0, +\infty)^{|\mathcal{C}|}} e^{-\mathbf{u}^\top M \mathbf{u} / 2} d\mathbf{u}.$$

where M is the submatrix of $\mathcal{H}_K(\tau)^{-1}$ that corresponds to the rows and columns of \mathcal{C} .

As in the subcritical case, this result establishes that the probability converges exponentially fast to zero whenever at least one coordinate is subcritical. The Gaussian integral associated with the critical coordinates is just a multiplicative constant in this asymptotic behavior.

4.7.2 Application: Mallows Culture $\mathcal{M}_{4 \text{ last}}$

In Section 4.5.2, we established that for the Mallows culture $\mathcal{M}_{3 \text{ last}}$, all coordinates of the saddle point are subcritical. In the case of 4 candidates, *i.e.* for $\mathcal{M}_{4 \text{ last}}$, Lemma 9 yields $\zeta = (e^{-2\rho}, e^{-\rho}, 1)$. The first two coordinates are subcritical, while the last one is critical. We then apply Theorem 9. Since the Gaussian integral is univariate in that case, we compute it explicitly.

Theorem 10. *Under the Mallows culture $\mathcal{M}_{4 \text{ last}}$, denoting M_{33} is the bottom right coefficient of $\mathcal{H}_K(\tau)^{-1}$, the probability that candidate 4 is the Condorcet winner has asymptotic behavior*

$$\mathbb{P}(4 \text{ is CW}) \underset{n \rightarrow +\infty}{\sim} \frac{P(\zeta)^n}{\prod_{j \in \mathcal{S}} ((1 - \zeta_j) \zeta_j^{\lceil n/2 \rceil - 1})} \frac{1}{4\pi n \sqrt{\det(\mathcal{H}_K(\tau))} M_{33}},$$

In this expression, computing $P(\zeta)$ by hand remains relatively straightforward and yields the elegant result: $P(\zeta) = 8\gamma e^{-3\rho} (1 + e^{-\rho}) \left(\frac{1}{2} + e^{-\rho}\right)^2$. However, the expressions of $\mathcal{H}_K(\tau)$ and its inverse being large and not particularly relevant when explicitated, we omit them. More generally, as m increases, the computation of $P(\zeta)$ and $\mathcal{H}_K(\tau)$ is best left to the computer. The package `Actinvoiting` [Dur25] provides symbolic algorithms to efficiently compute these quantities.

Numerical simulations Figure 4.5 shows our results for $\mathcal{M}_{4 \text{ last}}$. Exact computation requires evaluating the polynomial $P(\mathbf{x})$, which has 2^{m-1} terms corresponding to all subsets of adversaries, followed by expanding $P(\mathbf{x})^n$, resulting in a computational cost of at least $2^{(m-1)n}$. For $m = 4$ and $n = 25$, this already takes over 7 minutes, making larger n impractical. So we did not perform this calculation for larger values of n . In contrast, Monte Carlo simulations remain manageable, taking 24 seconds for $n = 25$. However, as usual, their limitation lies in the precision of the calculation: in particular, when the exact probabilities are small, Monte Carlo simulations often return an empirical probability of zero. Finally, the theoretical approximation fits the exact curve well while remaining extremely fast, taking only 437 μs for $n = 25$. This illustrates that our asymptotic approach not only accelerates computations but also makes them feasible where exact methods become intractable.

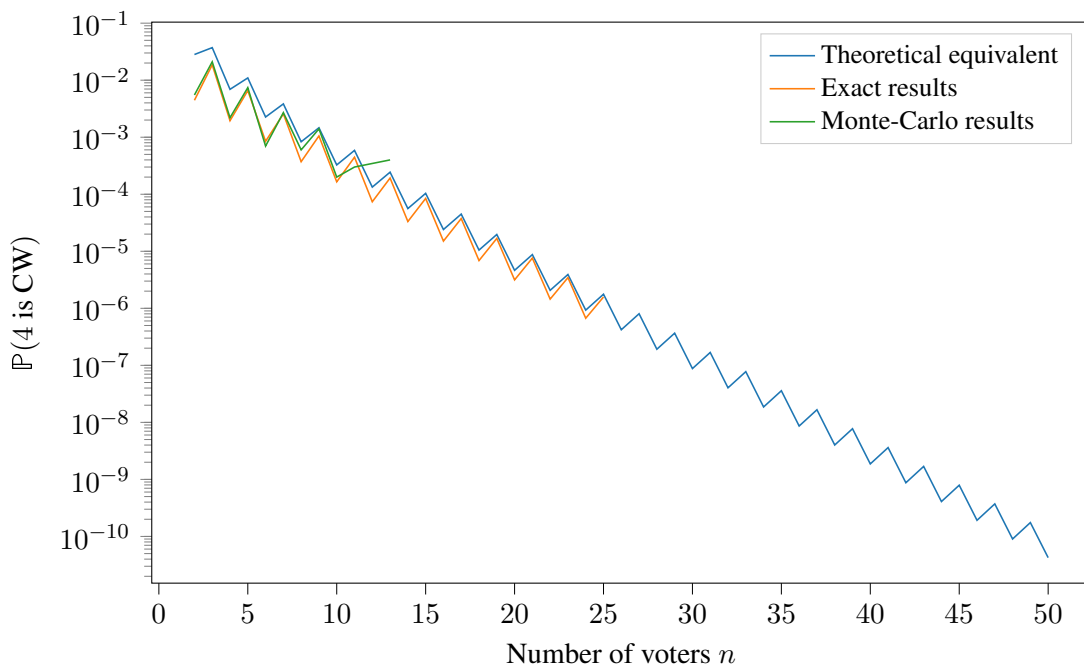


Fig. 4.5: Probability that candidate 4 is the Condorcet winner as a function of n in a culture $\mathcal{M}_{4 \text{ last}}$ with $\rho = \log(2)$, shown on a semilog scale. The theoretical equivalent is based on Theorem 10, while exact results rely on Equation (4.2). Monte Carlo simulations use 10,000 profiles per point, yielding an error of order 10^{-2} . For $n \geq 14$, they return zero, which is not visible due to the logarithmic scale. Exact results are not computed for $n \geq 26$ due to prohibitive runtime (e.g., 7 minutes for $n = 25$).

4.8 Dealing With Supercriticality

To compute $\mathbb{P}(m \succ_{\alpha} \mathcal{A})$, our general approach relies on calculating the characteristic polynomial $P(\mathbf{x})$ and using the saddle point method. In Sections 4.5 to 4.7, the singularity at 1 arising from the terms $\frac{1}{1-x_j}$ in Equation (4.5) was not an issue, as each coordinate ζ_j of the saddle point was either subcritical or critical. Specifically, when ζ_j is subcritical, the corresponding integration contour is simply a circle that passes through ζ_j , and hence does not encircle 1. When ζ_j is critical, the integration contour for that coordinate can be chosen to slightly bypass the singularity at 1.

We now consider the case where at least one coordinate ζ_j is supercritical, *i.e.* $\zeta_j > 1$. In this scenario, the integration contour, which must pass sufficiently close to ζ_j , inevitably encloses the singularity at 1. In general, the contribution from this singularity is non-negligible and can be determined via a residue calculation. However, rather than relying solely on calculus, we provide an equivalent, more intuitive interpretation based on the complement rule.

4.8.1 Theoretical Result for Supercriticality

Assume that the coordinate ζ_j is supercritical. The complement rule gives

$$\mathbb{P}(m \succ_{\alpha} \mathcal{A}) = \mathbb{P}(m \succ_{\alpha} \mathcal{A} \setminus \{j\}) - \mathbb{P}(m \succ_{\alpha} \mathcal{A} \setminus \{j\} \wedge m \preceq_{\alpha} j). \quad (4.9)$$

In the first term on the right-hand side, the event $\{m \succ_{\alpha} \mathcal{A} \setminus \{j\}\}$ means that m is an α -winner after removing adversary j , reducing the competition to $m - 2$ adversaries. As we will elaborate, the associated culture arises by setting $x_j = 1$ in $P(\mathbf{x})$, allowing the application of the saddle point method in a lower-dimensional space.

The second term is the probability that m defeats all candidates in $\mathcal{A} \setminus \{j\}$ but not j in the sense of α . As we will see, this probability is expressed by using a modified characteristic polynomial and a new saddle point derived from ζ , which is modified solely by inverting its j -th coordinate, thereby making it subcritical.

At this stage, the saddle point of the new characteristic polynomial may still have supercritical coordinates. Thus, the process might need to be iterated, using the following generalization of Equation (4.9). Let \mathcal{X} and \mathcal{Y} be disjoint sets of adversaries, and let $j \in \mathcal{X}$. Then,

$$\mathbb{P}(m \succ_{\alpha} \mathcal{X} \wedge m \preceq_{\alpha} \mathcal{Y}) = \mathbb{P}(m \succ_{\alpha} \mathcal{X} \setminus \{j\} \wedge m \preceq_{\alpha} \mathcal{Y}) - \mathbb{P}(m \succ_{\alpha} \mathcal{X} \setminus \{j\} \wedge m \preceq_{\alpha} \mathcal{Y} \cup \{j\}). \quad (4.10)$$

Once again, the first term reduces the dimensionality of the problem, while the second term enables the inversion of a saddle point coordinate, which will be used in practice to transform a supercritical coordinate into a subcritical one. By iterating this process, we systematically reduce the problem to terms of the form $\mathbb{P}(m \succ_{\alpha} \mathcal{X} \wedge m \preccurlyeq_{\alpha} \mathcal{Y})$, for well-chosen sets \mathcal{X} and \mathcal{Y} .

To compute such terms, we define a new characteristic polynomial $P_{\mathcal{Y}}^{\mathcal{X}}(\mathbf{x}_{\mathcal{X}}, \mathbf{y}_{\mathcal{Y}})$, with variables $\mathbf{x}_{\mathcal{X}} = (x_j)_{j \in \mathcal{X}}$ and $\mathbf{y}_{\mathcal{Y}} = (y_j)_{j \in \mathcal{Y}}$, derived from the original polynomial $P(\mathbf{x})$ through the following transformations. For each adversary $j \in \mathcal{X}$, the formal variable x_j keeps its interpretation and does not lead to a modification in the polynomial. For each adversary $j \in \mathcal{Y}$, instead of using x_j , which encodes when j is preferred to m , we introduce a new formal variable y_j that encodes when m is preferred to j . This variable will appear in all monomials that do not contain x_j in $P(\mathbf{x})$. Finally, for each adversary $j \in \mathcal{A} \setminus (\mathcal{X} \cup \mathcal{Y})$, we disregard the outcome of the pairwise comparison, hence we eliminate the corresponding variable by setting $x_j = 1$. This corresponds to the algebraic operation:

$$P_{\mathcal{Y}}^{\mathcal{X}}(\mathbf{x}_{\mathcal{X}}, \mathbf{y}_{\mathcal{Y}}) = \left(\prod_{j \in \mathcal{Y}} y_j \right) P\left(\mathbf{x}_{\mathcal{X}}, \frac{\mathbf{1}}{\mathbf{y}_{\mathcal{Y}}}, \mathbf{1}_{\mathcal{A} \setminus (\mathcal{X} \cup \mathcal{Y})}\right),$$

where $\frac{\mathbf{1}}{\mathbf{y}_{\mathcal{Y}}} := (y_j^{-1})_{j \in \mathcal{Y}}$. As an example, consider $m = 3$, $\mathcal{X} = \{1\}$, and $\mathcal{Y} = \{2\}$. From the expression of P in Equation (4.1), we deduce:

$$P_{\{2\}}^{\{1\}}(x_1, y_2) = p_{\emptyset} y_2 + p_{\{1\}} x_1 y_2 + p_{\{2\}} + p_{\{1,2\}} x_1.$$

Note that $P_{\mathcal{Y}}^{\mathcal{X}}(\mathbf{x}_{\mathcal{X}}, \mathbf{y}_{\mathcal{Y}})$ generalizes $P(\mathbf{x})$, since $P(\mathbf{x}) = P_{\emptyset}^{\mathcal{A}}(\mathbf{x}_{\mathcal{A}}, \mathbf{y}_{\emptyset})$. As in Equation (4.2), the probability of interest is expressed as a coefficient extraction:

$$\mathbb{P}(m \succ_{\alpha} \mathcal{X} \wedge m \preccurlyeq_{\alpha} \mathcal{Y}) = \left[\prod_{j \in \mathcal{X}} x_j^{<\beta_j n} \prod_{j \in \mathcal{Y}} y_j^{\leq \alpha_j n} \right] P_{\mathcal{Y}}^{\mathcal{X}}(\mathbf{x}_{\mathcal{X}}, \mathbf{y}_{\mathcal{Y}}).$$

Given two disjoint sets of adversaries \mathcal{X} and \mathcal{Y} , Definition 9 generalizes to the cumulant generating function $\tilde{K}(\mathbf{t}_{\mathcal{X} \cup \mathcal{Y}}) = \log(P_{\mathcal{Y}}^{\mathcal{X}}(\exp(\mathbf{t}_{\mathcal{X} \cup \mathcal{Y}})))$, the log saddle point $\tilde{\boldsymbol{\tau}} = \arg \max_{\mathbf{t}_{\mathcal{X} \cup \mathcal{Y}} \in \mathbb{R}^{|\mathcal{X} \cup \mathcal{Y}|}} (-K(\mathbf{t}_{\mathcal{X} \cup \mathcal{Y}}) + (\boldsymbol{\beta}_{\mathcal{X}}, \boldsymbol{\alpha}_{\mathcal{Y}})^{\top} \mathbf{t}_{\mathcal{X} \cup \mathcal{Y}})$, and the saddle point $\tilde{\boldsymbol{\zeta}} = e^{\tilde{\boldsymbol{\tau}}}$.

As previously mentioned, transferring an adversary from \mathcal{X} to \mathcal{Y} results in the inversion of the corresponding coordinate in the saddle point. The following lemma formalizes this process.

Lemma 10. *Let \mathcal{X} and \mathcal{Y} be two disjoint sets of adversaries. Let $\tilde{\boldsymbol{\zeta}}$ be the saddle point associated with $\mathbb{P}(m \succ_{\alpha} \mathcal{X} \wedge m \preccurlyeq_{\alpha} \mathcal{Y})$. Let $j \in \mathcal{X}$, $\mathcal{X}' = \mathcal{X} \setminus \{j\}$, and $\mathcal{Y}' = \mathcal{Y} \cup \{j\}$. Then the saddle point associated with $\mathbb{P}(m \succ_{\alpha} \mathcal{X}' \wedge m \preccurlyeq_{\alpha} \mathcal{Y}')$ has its k -th coordinate equal to $\tilde{\zeta}_k$ if $k \neq j$ and $\frac{1}{\tilde{\zeta}_j}$ if $k = j$.*

Proof. Transferring a candidate from \mathcal{X} to \mathcal{Y} corresponds to the following algebraic operation:

$$P_{\mathcal{Y}'}^{\mathcal{X}'}(\mathbf{x}_{\mathcal{X}'}, \mathbf{y}_{\mathcal{Y}'}) = y_j P_{\mathcal{Y}}^{\mathcal{X}}(\mathbf{x}_{\mathcal{X}'}, \frac{1}{y_j}, \mathbf{y}_{\mathcal{Y}}), \quad (4.11)$$

where y_j encodes when m is preferred to j , whereas x_j encoded the opposite.

Now, let (β, α) denote the threshold vector associated with $\mathbb{P}(m \succ_{\alpha} \mathcal{X} \wedge m \preccurlyeq_{\alpha} \mathcal{Y})$, and $(\hat{\beta}, \hat{\alpha})$ the threshold vector associated with $\mathbb{P}(m \succ_{\alpha} \mathcal{X}' \wedge m \preccurlyeq_{\alpha} \mathcal{Y}')$. Remark that $\hat{\beta}$ is the vector β where the coordinate β_j has been removed, and $\hat{\alpha}$ is the vector α where the coordinate α_j has been added. Finally, let us denote $\hat{\zeta}$ the saddle point associated with $\mathbb{P}(m \succ_{\alpha} \mathcal{X}' \wedge m \preccurlyeq_{\alpha} \mathcal{Y}')$. By definition of $\hat{\zeta}$, extending the remark in Section 4.3.3, we have

$$\hat{\zeta} = \arg \min_{(\mathbb{R}_{>0})^{|\mathcal{X}' \cup \mathcal{Y}'|}} \frac{P_{\mathcal{Y}'}^{\mathcal{X}'}(\mathbf{x}_{\mathcal{X}'}, \mathbf{y}_{\mathcal{Y}'})}{\mathbf{x}_{\mathcal{X}'}^{\hat{\beta}} \mathbf{y}_{\mathcal{Y}'}^{\hat{\alpha}}}.$$

So injecting Equation (4.11) we get

$$\hat{\zeta} = \arg \min_{(\mathbb{R}_{>0})^{|\mathcal{X}' \cup \mathcal{Y}'|}} \frac{P_{\mathcal{Y}}^{\mathcal{X}}(\mathbf{x}_{\mathcal{X}'}, \frac{1}{y_j}, \mathbf{y}_{\mathcal{Y}'})}{\mathbf{x}_{\mathcal{X}'}^{\hat{\beta}} y_j^{-\beta_j} \mathbf{y}_{\mathcal{Y}}^{\alpha}}.$$

Since the minimum of a function over a positive orthant remains unchanged under the inversion of a coordinate, setting $x_j = \frac{1}{y_j}$ yields

$$\min_{(\mathbb{R}_{>0})^{|\mathcal{X}' \cup \mathcal{Y}'|}} \frac{P_{\mathcal{Y}}^{\mathcal{X}}(\mathbf{x}_{\mathcal{X}'}, \frac{1}{y_j}, \mathbf{y}_{\mathcal{Y}'})}{\mathbf{x}_{\mathcal{X}'}^{\hat{\beta}} y_j^{-\beta_j} \mathbf{y}_{\mathcal{Y}}^{\alpha}} = \min_{(\mathbb{R}_{>0})^{|\mathcal{X} \cup \mathcal{Y}|}} \frac{P_{\mathcal{Y}}^{\mathcal{X}}(\mathbf{x}_{\mathcal{X}}, \mathbf{y}_{\mathcal{Y}})}{\mathbf{x}_{\mathcal{X}}^{\beta} \mathbf{y}_{\mathcal{Y}}^{\alpha}},$$

and more importantly, the transformation $x_j = \frac{1}{y_j}$ yields

$$\hat{\zeta} = (\tilde{\zeta}_1, \dots, \tilde{\zeta}_{j-1}, \frac{1}{\tilde{\zeta}_j}, \tilde{\zeta}_{j+1}, \dots, \tilde{\zeta}_{m-1}). \quad \square$$

This lemma, combined with the iterated application of Equation (4.10), allows expressing the probability $\mathbb{P}(m \text{ is } \alpha\text{-winner})$ as a sum of terms of the form $\mathbb{P}(m \succ_{\alpha} \mathcal{X} \wedge m \preccurlyeq_{\alpha} \mathcal{Y})$, where all the coordinates of all saddle points are either subcritical or critical. The following theorem provides the asymptotic behavior of such terms. It directly follows from Theorem 18 in Chapter 5.

Theorem 11. *Let \mathcal{X} and \mathcal{Y} be disjoint sets of adversaries. Let $\tilde{P}(\mathbf{x}_{\mathcal{X}}, \mathbf{y}_{\mathcal{Y}}) := P_{\mathcal{Y}}^{\mathcal{X}}(\mathbf{x}_{\mathcal{X}}, \mathbf{y}_{\mathcal{Y}})$ and \tilde{K} its cumulant generating function. Let $\tilde{\tau}$ and $\tilde{\zeta}$ respectively be the log saddle point and saddle point*

associated with $\mathbb{P}(m \succ_{\alpha} \mathcal{X} \wedge m \preccurlyeq_{\alpha} \mathcal{Y})$. We assume that (1) $\mathcal{X} = \mathcal{S} \cup \mathcal{C}$, where $\tilde{\zeta}_j < 1, \forall j \in \mathcal{S}$ and $\tilde{\zeta}_j = 1, \forall j \in \mathcal{C}$, and (2) $\tilde{\zeta}_j < 1, \forall j \in \mathcal{Y}$. Then

$$\mathbb{P}(m \succ_{\alpha} \mathcal{X} \wedge m \preccurlyeq_{\alpha} \mathcal{Y}) \underset{n \rightarrow +\infty}{\sim} \frac{\tilde{P}(\tilde{\zeta})^n}{\prod_{j \in \mathcal{S}} \tilde{\zeta}_j^{[\beta_j n] - 1} \prod_{j \in \mathcal{Y}} \tilde{\zeta}_j^{[\alpha_j n]}} \frac{1}{\prod_{j \in \mathcal{S} \cup \mathcal{Y}} (1 - \tilde{\zeta}_j)} \frac{1}{\sqrt{(2\pi)^{|\mathcal{X} \cup \mathcal{Y}|} n^{|\mathcal{S} \cup \mathcal{Y}|} \det(\mathcal{H}_{\tilde{K}}(\tilde{\tau}))}} \int_{(0, +\infty)^{|\mathcal{C}|}} e^{-\mathbf{u}^{\top} \tilde{M} \mathbf{u} / 2} d\mathbf{u},$$

where \tilde{M} is the submatrix of $\mathcal{H}_{\tilde{K}}(\tilde{\tau})^{-1}$ that corresponds to the rows and columns of \mathcal{C} .

In summary, to handle supercriticality, the process consists in applying Equation (4.10) to one of the supercritical coordinates of the original saddle point, and then iterating this step on each term whose saddle point still contains supercritical coordinates, until every term satisfies the assumptions of Theorem 11. The procedure is guaranteed to terminate, as each iteration either reduces the dimension by one or decreases the number of supercritical coordinates. Once Theorem 11 is used to compute the asymptotics of each term, some may be negligible compared to others, leading to a simplified sum with fewer terms, as illustrated in the following application.

4.8.2 Application: Mallows Culture $\mathcal{M}_{3 \text{ first}}$

Analogously to Lemma 9 for the culture $\mathcal{M}_{m \text{ last}}$, Lemma 11 provides the expression of the saddle point for $\mathcal{M}_{m \text{ first}}$. The proof is similar and is provided in Section 4.5.2.

Lemma 11. *Under the culture $\mathcal{M}_{m \text{ first}}$, the log saddle point τ is given by*

$$\tau = \left(\frac{m\rho}{2}, \frac{(m-2)\rho}{2}, \dots, \frac{(-m+4)\rho}{2} \right),$$

where ρ is the concentration parameter of the culture, defined in Section 4.2.

In the case of three candidates, Lemma 11 gives $\zeta = (e^{3\rho/2}, e^{\rho/2})$, hence both coordinates are supercritical. Consequently, we apply the procedure outlined in Section 4.8.1. In this specific case, let us show that this approach effectively reduces to applying the inclusion-exclusion principle and identifying the dominant term in the sum.

First, it is needed to iteratively apply Equation (4.11) on the supercritical coordinates, until every term is subcritical, which gives the following equality

$$1 - \mathbb{P}(3 \text{ is CW}) = (\mathbb{P}(3 \preceq_{\frac{1}{2}} \{1\}) + \mathbb{P}(3 \preceq_{\frac{1}{2}} \{2\})) - \mathbb{P}(3 \preceq_{\frac{1}{2}} \{1,2\}).$$

Subcriticality. Note that the characteristic polynomial associated with $\mathbb{P}(3 \preceq_{\frac{1}{2}} \{1,2\})$, namely $P_{\{1,2\}}^0(y_1 y_2)$, is actually the characteristic polynomial of the probability $\mathbb{P}(3 \text{ is CW})$ under the Mallows model $\mathcal{M}_{3 \text{ last}}$. Therefore, we already know that the saddle point for this term is subcritical.

The characteristic polynomials $P_{\{1\}}^{\{2\}}(y_1, 1)$ and $P_{\{2\}}^{\{1\}}(1, y_2)$ associated respectively with $\mathbb{P}(3 \preceq_{\alpha} \{1\})$ and $\mathbb{P}(3 \preceq_{\alpha} \{2\})$ are both univariate and of the form $\gamma(q + py)$. Denoting ξ a saddle point associated with such a polynomial, a straightforward calculation yields $\xi = \frac{q}{p}$. The verification of subcriticality follows directly, by simply writing explicitly

$$P_{\{1\}}^{\{2\}}(y_1, 1) = \gamma \left((2e^{-2\rho} + e^{-3\rho}) + (1 + 2e^{-\rho})y_1 \right),$$

and

$$P_{\{2\}}^{\{1\}}(1, y_2) = \gamma \left((e^{-\rho} + e^{-2\rho} + e^{-3\rho}) + (1 + e^{-\rho} + e^{-2\rho})y_2 \right).$$

Dominating term. The first two terms can be associated with a Mallows culture with two candidates and a concentration parameter $\tilde{\rho} = \log(\frac{q}{p})$. In the case of $\mathbb{P}(3 \preceq_{\alpha} \{2\})$, $\tilde{\rho} = \rho$. In the case of $\mathbb{P}(3 \preceq_{\alpha} \{1\})$, $\tilde{\rho}$ is equal to ρ multiplied by a quantity strictly less than 1. Therefore, this Mallows is less concentrated than the first one. Thus, asymptotically, $\mathbb{P}(3 \preceq_{\alpha} \{2\})$ will dominate $\mathbb{P}(3 \preceq_{\alpha} \{1\})$.

In particular, as $\mathbb{P}(3 \preceq_{\alpha} \{1,2\}) \leq \mathbb{P}(3 \preceq_{\alpha} \{1\})$, the third term of the sum is also dominated by $\mathbb{P}(3 \preceq_{\alpha} \{2\})$. Thus,

$$1 - \mathbb{P}(3 \text{ is CW}) \underset{n \rightarrow +\infty}{\sim} \mathbb{P}(3 \preceq_{\alpha} \{2\}). \quad (4.12)$$

Unsurprisingly, the dominant term that may prevent candidate 3 from being the Condorcet winner corresponds to the event of not winning the pairwise comparison against the strongest adversary, candidate 2.

Theorem 12. *Under $\mathcal{M}_{3 \text{ first}}$, the probability that candidate 3 is the Condorcet winner has asymptotic behavior*

$$1 - \mathbb{P}(3 \text{ is CW}) \underset{n \rightarrow +\infty}{\sim} \sqrt{\frac{2}{\pi n}} \frac{e^{-\rho \lceil n/2 \rceil}}{1 - e^{-\rho}} \left(\frac{2}{1 + e^{-\rho}} \right)^n.$$

Proof. Using Theorem 11 and Equation (4.12), we get

$$1 - \mathbb{P}(3 \text{ is CW}) \underset{n \rightarrow +\infty}{\sim} \frac{P_{\{2\}}^{\{1\}}(1, \xi)^n}{(1 - \xi)\xi^{\lfloor n/2 \rfloor}} \frac{1}{\sqrt{2\pi n^{\frac{1}{4}}}},$$

where $\xi = e^{-\rho}$, thus $P_{\{2\}}^{\{1\}}(1, \xi) = 2\gamma e^{-\rho}(1 + e^{-\rho} + e^{-2\rho})$, with $\gamma = \frac{1}{1+2e^{-\rho}+2e^{-2\rho}+e^{-3\rho}}$.

Straightforward calculations yield the final formula. □

Numerical simulations Figure 4.6 illustrates these results. Since the probability that candidate 3 is the Condorcet winner tends towards 1, we now represent the complement probability on a semi-logarithmic scale to visualize the difference with the limit. The computation times are similar to those in Figure 4.3 as they involve the same culture and parameters. As in the previous cases, Monte Carlo simulations become unreliable when probabilities are small. We observe good agreement between the theoretical approximation, the exact results, and the Monte Carlo estimate when the latter is relevant.

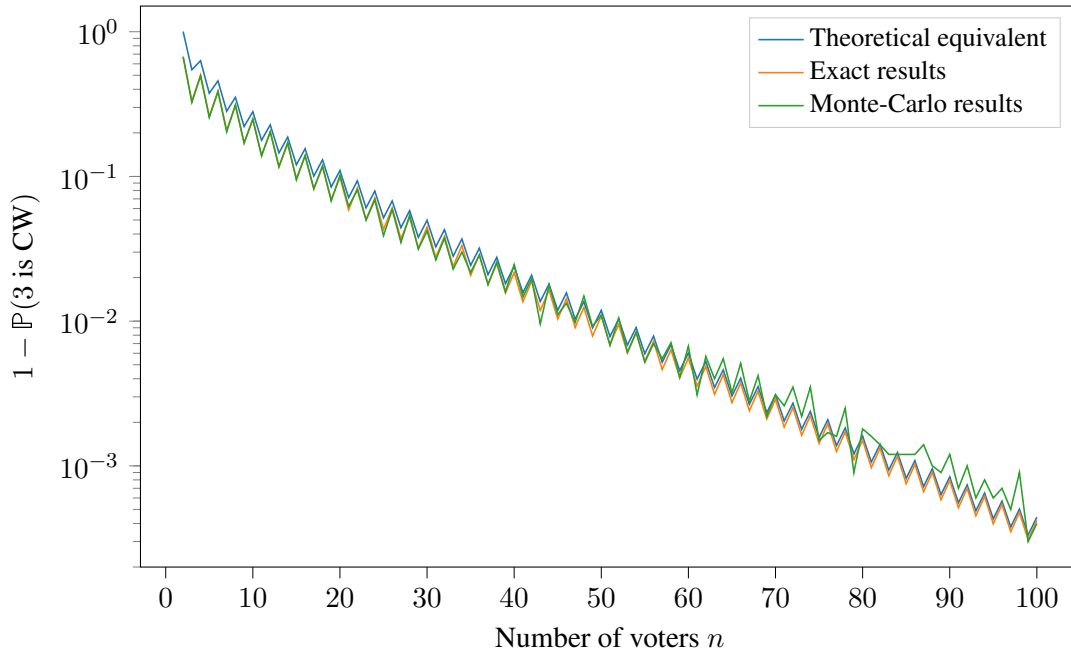


Fig. 4.6: Probability that candidate 3 fails to be the Condorcet winner as a function of n in a culture $\mathcal{M}_{3 \text{ first}}$ with $\rho = \log(2)$, shown on a semilog scale. The theoretical equivalent is based on Theorem 12, while exact results rely on Equation (4.2). Monte Carlo simulations use 10,000 profiles per point, yielding an error of order 10^{-2} . When the exact results curve is not visible, it is overlapped by the Monte Carlo results curve.

4.9 Asymptotic Expansion

We now explain how to improve our asymptotic results. Both Theorem 4 for the subcritical case and Theorem 6 for the critical case are encompassed by Theorem 9, which addresses the *mixed case*. As detailed in Section 4.8.1, the supercritical case also reduces to the mixed case. Therefore, we focus on this scenario, involving a set \mathcal{S} of subcritical coordinates and a set \mathcal{C} of critical coordinates. Essentially, Theorem 9 provides a constant a_0 such that

$$\mathbb{P}(m \text{ is CW}) \underset{n \rightarrow +\infty}{\sim} \frac{P(\boldsymbol{\zeta})^n}{\prod_{j \in \mathcal{S}} \zeta_j^{\lceil \beta_j n \rceil - 1}} \frac{1}{n^{|\mathcal{S}|/2}} a_0.$$

Theorem 19 in Chapter 5 extends this result, establishing the existence of coefficients $(a_{k,n})_{k,n \geq 1}$ such that, for any $r \geq 0$, we have

$$\mathbb{P}(m \text{ is CW}) = \frac{P(\boldsymbol{\zeta})^n}{\prod_{j \in \mathcal{S}} \zeta_j^{\lceil \beta_j n \rceil - 1}} \frac{1}{n^{|\mathcal{S}|/2}} \left(a_0 + a_{1,n} n^{-1/2} + \cdots + a_{r-1,n} n^{-(r-1)/2} + \mathcal{O}(n^{-r/2}) \right).$$

In general, the coefficients $a_{k,n}$ may depend on n , but only through the bounded vector $\lceil \beta n \rceil - \mathbf{1} - \beta n$. For example, in the case of the Condorcet winner, they depend only on the parity of n .

Such an approximation is called an *asymptotic expansion*. In the subcritical case, it is well known [FS09; PW13, Section VIII.3][Chapter 5] and proven in Theorem 19, that $a_{k,n} = 0$ for odd k , leading to an expansion in powers of n^{-1} instead of $n^{-1/2}$. However, this does not hold in general, as the following result illustrates.

For the Impartial Culture, Theorem 7 gives the limiting probability but not the convergence rate, which we now address by computing $a_{1,n}$. This explicit computation results in the application of Lemma 16 in Chapter 5.

Theorem 13. *Under Impartial Culture with m candidates, the probability that candidate m is the Condorcet winner has asymptotic behavior*

$$\mathbb{P}(m \text{ is CW}) = a_0 + a_{1,n} n^{-1/2} + \mathcal{O}(n^{-1}),$$

where a_0 and $a_{1,n}$ are given by

$$a_0 = \frac{1}{\sqrt{(2\pi)^{m-1} \det(\mathcal{H}_K(\mathbf{0}))}} \int_{(0,+\infty)^{m-1}} e^{-\mathbf{u}^\top \mathcal{H}_K(\mathbf{0})^{-1} \mathbf{u}/2} d\mathbf{u},$$

$$a_{1,n} = \begin{cases} 0 & \text{if } n \text{ is odd,} \\ -\frac{6}{m+1} \frac{1}{\sqrt{(2\pi)^{m-1} \det(\mathcal{H}_K(\mathbf{0}))}} \int_{(0,+\infty)^{m-1}} \mathbf{1}^\top \mathbf{u} e^{-\mathbf{u}^\top \mathcal{H}_K(\mathbf{0})^{-1} \mathbf{u}/2} d\mathbf{u}. & \text{if } n \text{ is even,} \end{cases}$$

and $\mathcal{H}_K(\mathbf{0})$, its inverse and determinant are given in Equation (4.7).

Proof. A useful property of the cumulant generating function is that the third term of its Taylor expansion $\phi_3 := \mathbf{t} \mapsto [y^3] \log(P(e^{t\mathbf{y}}))$ (using the notation from Lemma 16 in Chapter 5) is equal to the third central moment of \mathbf{X} . Thus

$$\phi_3(\mathbf{t}) = \sum_{j,k,\ell \in \{1,\dots,m-1\}} \mathbb{E}((X_j - 1/2)(X_k - 1/2)(X_\ell - 1/2)) t_j t_k t_\ell.$$

We develop

$$\begin{aligned} \mathbb{E}\left(\left(X_j - \frac{1}{2}\right)\left(X_k - \frac{1}{2}\right)\left(X_\ell - \frac{1}{2}\right)\right) &= \mathbb{E}(X_j X_k X_\ell) \\ &\quad - \frac{1}{2} \left(\mathbb{E}(X_j X_k) + \mathbb{E}(X_j X_\ell) + \mathbb{E}(X_k X_\ell) \right) \\ &\quad + \frac{3}{4} \mathbb{E}(X_j) - \frac{1}{8}. \end{aligned}$$

As we saw earlier, we have $\mathbb{E}(X_j X_k X_\ell) = \frac{1}{4}$. In all cases, whether j, k and ℓ are distinct or some of them are equal, we find the third central moment to be 0 and deduce $\phi_3(\mathbf{t}) = 0$.

The term $a_{1,n}$ is then obtained by application of Lemma 16 with $c = d = m - 1$, $\mathbf{b} = -\mathbf{1}/2$, $A(\mathbf{z}) = 1$ so $\nabla_A(\boldsymbol{\zeta}) = \mathbf{0}$, $\mathcal{H} = \mathcal{H}_K(\mathbf{0})$, $\phi_3^*(\mathbf{u}) = 0$ because $\phi_3(\mathbf{t}) = 0$, and $\boldsymbol{\kappa}_n - \boldsymbol{\lambda}n = (\lceil n/2 \rceil - 1 - n/2)\mathbf{1}$. \square

Numerical simulations In Figure 4.4, we observed that the approximation by the limit value was highly accurate for odd values of n . Theorem 13 provides an explanation for this: the term of the asymptotic expansion in $n^{-1/2}$ vanishes in this case. In contrast, for even n , the term is non-zero and negative, which is consistent with the previous observation that the exact values are significantly lower than the limit in these cases. To illustrate this, Figure 4.7 shows the same curves as Figure 4.4 for the Monte Carlo simulations and exact values. However, this time, the theoretical approximation

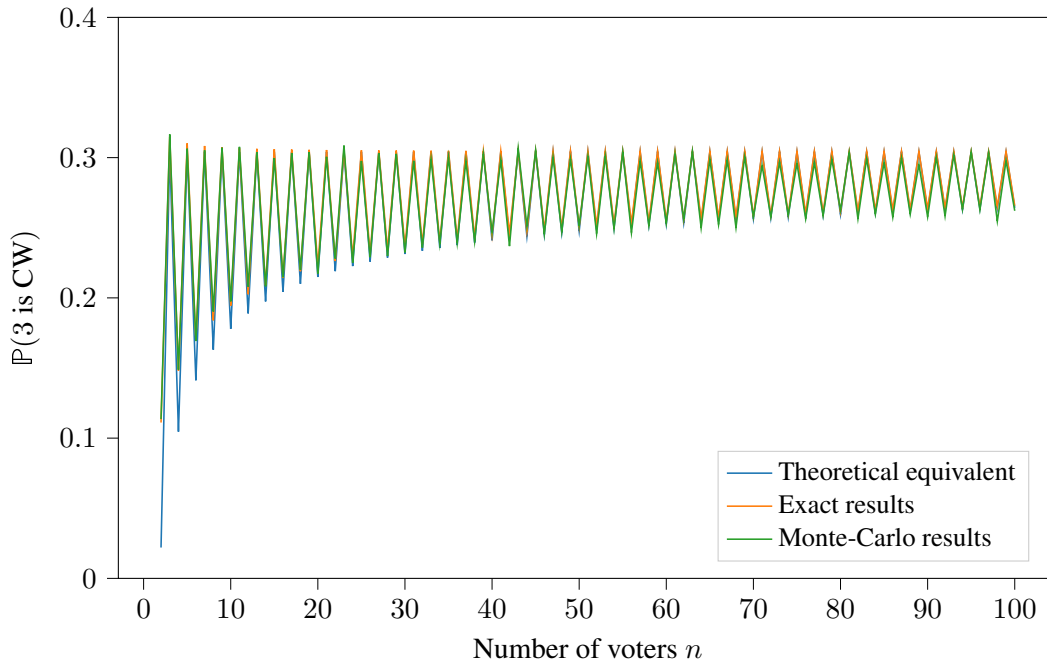


Fig. 4.7: Probability that a given candidate is the Condorcet winner as a function of n in the Impartial Culture with three candidates. The theoretical equivalent is based on Theorem 13, while exact results rely on Equation (4.2). Monte Carlo simulations use 10,000 profiles per point, yielding an error of order 10^{-2} . Not all curves being visible mean that they overlap.

includes the term in $n^{-1/2}$. As before, this approximation is computationally inexpensive: for instance, it takes only $55 \mu\text{s}$ for $n = 100$. Now, the three curves become nearly indistinguishable visually: the addition of this term in $n^{-1/2}$ significantly improves the accuracy of the approximation, the error term being $\mathcal{O}(n^{-1})$.

4.10 Conclusion

Summary of the Contributions In this chapter, we have presented a method for calculating the probability that a candidate is an α -winner under the GIC model, assuming each ranking has a strictly positive probability. Our approach, based on analytic combinatorics, first computed the characteristic polynomial, the cumulant generating function, and the saddle point. The asymptotic analysis then split into cases based on whether each coordinate of the saddle point is subcritical, critical, or supercritical. In all cases, we derived the limiting probability and rate of convergence. Additionally, our method computed higher-order terms in the asymptotic expansion. We applied

this approach to the Impartial Culture and Mallows models, providing explicit formulas supported by numerical simulations.

For Impartial Culture, the probability that there exists a Condorcet winner is of the form

$$\mathbb{P}(m \text{ is CW}) = a_0 + a_{1,n}n^{-1/2} + \mathcal{O}(n^{-1}),$$

where a_0 and $a_{1,n}$ are given by

$$a_0 = \frac{1}{\sqrt{(2\pi)^{m-1} \det(\mathcal{H}_K(\mathbf{0}))}} \int_{(0,+\infty)^{m-1}} e^{-\mathbf{u}\mathcal{H}_K(\mathbf{0})^{-1}\mathbf{u}/2} d\mathbf{u},$$

$$a_{1,n} = \begin{cases} 0 & \text{if } n \text{ is odd,} \\ -\frac{6}{m+1} \frac{1}{\sqrt{(2\pi)^{m-1} \det(\mathcal{H}_K(\mathbf{0}))}} \int_{(0,+\infty)^{m-1}} \mathbf{1}^\top \mathbf{u} e^{-\mathbf{u}\mathcal{H}_K(\mathbf{0})^{-1}\mathbf{u}/2} d\mathbf{u}. & \text{if } n \text{ is even,} \end{cases}$$

and $\mathcal{H}_K(\mathbf{0})$, its inverse and determinant are given in Equation (4.7).

In the case of the Mallows model, we studied two main variants: one in which the candidate of interest occupies the first position in the reference ranking, and another where it appears last. The latter variant was examined in two settings, with three and four candidates, corresponding respectively to the models $\mathcal{M}_{3 \text{ last}}$ and $\mathcal{M}_{4 \text{ last}}$. For both cases, since the reference ranking can be interpreted as representing the underlying ground truth, we showed that the probability of the *a priori* undesirable event in which the candidate ranked last nevertheless emerges as the Condorcet winner tends exponentially fast to 0. Conversely, under the model $\mathcal{M}_{3 \text{ first}}$, where candidate 3 occupies the first position in the reference ranking, we showed that the probability that candidate 3 is the Condorcet winner tends exponentially fast to 1.

Future Work A natural direction for future work is to investigate cases where the saddle point has null or infinite coordinates, which may arise in non-generic cultures. Additionally, it would be valuable to perform a finer analysis of the complexity of the algorithm presented in Section 4.8 for handling supercritical coordinates, particularly to identify conditions under which it runs in polynomial time. Another avenue is to examine the limiting behavior of the probability as the number of candidates m tends to infinity. Finally, it would be insightful to extend our method to other events, such as the transitivity of the majority relation, different kinds of monotonicity failures, or the manipulability, *i.e.* the susceptibility to strategic voting.

Multivariate Analytic Combinatorics

5.1 Introduction

In the previous chapter, we were interested in the probability that a candidate is an α -winner. Our objective was to determine its asymptotic behavior as the number of voters n tends to infinity. To this end, we encoded the problem in a combinatorial framework by constructing the generating function associated with one voter:

$$P(\mathbf{x}) := \sum_{\mathcal{X} \subseteq \mathcal{A}} \left(p_{\mathcal{X}} \prod_{j \in \mathcal{X}} x_j \right).$$

This encoding allowed us to express the desired probability as a coefficient extraction problem

$$\mathbb{P}(m \text{ is } \alpha\text{-winner}) = \frac{1}{(2i\pi)^{m-1}} \oint \frac{P(\mathbf{x})^n}{\prod_{j=1}^{m-1} (1-x_j)} \frac{d\mathbf{x}}{\prod_{j=1}^{m-1} x_j^{\lceil \beta_j n \rceil}}. \quad (5.1)$$

In order to analyze this integral asymptotically, the next step is to apply the multivariate saddle point method. Accordingly, we defined the cumulant generating function associated with P

$$K : \mathbf{t} \in \mathbb{R}^{m-1} \mapsto \log \left(P(e^{\mathbf{t}}) \right),$$

and the *log saddle point* $\boldsymbol{\tau}$ and the *saddle point* $\boldsymbol{\zeta}$ as

$$\boldsymbol{\tau} := \arg \max_{\mathbf{t} \in \mathbb{R}^{m-1}} \left(-K(\mathbf{t}) + \boldsymbol{\beta}^{\top} \mathbf{t} \right), \quad \boldsymbol{\zeta} = e^{\boldsymbol{\tau}}.$$

From this point on, the asymptotic analysis requires some care, as it is not always standard. Notably, depending on the culture and on the vector $\boldsymbol{\alpha}$, we have seen that the saddle point may coincide with the singularity at 1, which is for instance the case when looking at the Condorcet winner under Impartial Culture. In such cases, the asymptotic analysis requires a refined treatment. This chapter is therefore devoted to developing the necessary multivariate analytic theorems to deal with all possible situations, combining both existing results and new contributions obtained in this work.

Our analytical results are developed at a more general level than required by our application. This abstraction both clarifies the methodological framework and enables future extensions to other settings.

Note that reading this chapter is not strictly necessary for the general understanding of the thesis, as I have already presented the main ideas behind the theorems developed here in Chapter 2 and Chapter 4. Thus, this chapter is primarily intended for readers who wish to dive into analytic combinatorics and apply similar techniques in their research, or who are curious and motivated to understand the machinery of the results presented previously. As this chapter is the most technical part of the manuscript, we assume that the reader has consulted Chapter 2 or is familiar with the field.

5.1.1 Related Work and Contributions

In univariate analytic combinatorics, several standard theorems and formulas are already available for asymptotic analysis. For instance, one of the central results, the *large power theorem* [FS09, Section VIII.8], provides sufficient conditions to derive the asymptotics of coefficient extractions of the form $[z^{\lambda n}] A(z) B(z)^n$ as n tends to infinity. A natural question is whether similar theorems exist in the multivariate setting, extending this result to

$$[z^{\lambda n}] A(\mathbf{z}) B(\mathbf{z})^n.$$

This extension is precisely formulated by the multivariate Laplace method presented in Pemantle and Wilson [PW13, Theorem 5.1.2].

Theorem 14 (Multivariate Laplace Method.). *Consider a compact neighborhood $C \subset \mathbb{R}^d$ of $\mathbf{0}$ and two functions $A(\mathbf{x})$ and $\phi(\mathbf{x})$ analytic on C . Suppose that the real part of $\phi(\mathbf{x})$ is strictly positive except at the origin, that its gradient vanishes at the origin and that its Hessian matrix \mathcal{H} is non-singular there. Assume $A(\mathbf{0}) \neq 0$, then*

$$\oint A(\mathbf{x}) e^{-n\phi(\mathbf{x})} d\mathbf{x} \sim \left(\frac{2\pi}{n}\right)^{d/2} \frac{A(\mathbf{0})}{\sqrt{\det(\mathcal{H})}}$$

and the choice of sign of the square root is defined by taking the product of the principal square roots of the eigenvalues of \mathcal{H} .

Variants of this theorem exist for the case where $A(\mathbf{0}) = 0$ and where the Hessian of $\phi(\mathbf{x})$ is singular at $\mathbf{0}$. In addition, if the assumptions of the theorem are satisfied, except $\phi(\mathbf{0}) \neq 0$, then we rewrite

$$\oint A(\mathbf{x})e^{-n\phi(\mathbf{x})}d\mathbf{x} = e^{-n\phi(\mathbf{0})} \oint A(\mathbf{x})e^{-n(\phi(\mathbf{x})-\phi(\mathbf{0}))}d\mathbf{x}.$$

and the theorem is applicable with $\phi(\mathbf{x})$ replaced by $\phi(\mathbf{x}) - \phi(\mathbf{0})$. Finally, if the point where the gradient of $\phi(\mathbf{x})$ vanishes is some other point than $\mathbf{0}$, as long as the Hessian of ϕ is non-singular at this point and A does not vanish there, the theorem is applicable after a translation sending this point to $\mathbf{0}$.

However, it is challenging to verify that a given generating function satisfies its assumptions (e.g., localization and uniqueness of the saddle point). So, our first contribution is to introduce new general results providing sufficient conditions for a generating function to meet the assumptions of Theorem 14. Our results extend the univariate case [FS09, Section VIII.8] to the multivariate setting. We consider the definitions of rank and periodicity for multivariate generating functions as some of our most significant contributions, along with Lemma 15, which extends the Daffodil Lemma [FS09, Lemma IV.1] to the multivariate case.

This allowed us to apply Theorem 14 in the subcritical case of Chapter 4, or more precisely, a slightly broader version of it (Theorem 17), which takes care of the ceiling function in the coefficient extraction.

When some of the coordinates of the saddle point are critical, it is not applicable anymore (saddle point and singularities coalesce). We adapt its proof to our needs (introducing, in particular, a different integration path), guided by general principles ([De 14]) and other cases of coalescence (e.g. [Ban+01, Theorem 11]). This adaptation, Theorem 18, is our second contribution and we hope it will help in the future with other cases of coalescence of saddle point and singularities.

Finally, analytic techniques are known for providing very precise asymptotic results, the so called "*asymptotic expansions*". For example, asymptotic expansions for multivariate Laplace integrals are provided by Pemantle and Wilson [PW13, Theorem 5.4.8].

Theorem 15 (Asymptotic Expansion Formula). *Consider a compact neighborhood $C \subset \mathbb{R}^d$ of $\mathbf{0}$ and two functions $A(\mathbf{x})$ and $\phi(\mathbf{x})$ analytic on C . Suppose that the real part of $\phi(\mathbf{x})$ is strictly positive except at the origin, that its gradient vanishes at the origin and that its Hessian matrix \mathcal{H} is*

non-singular there. Assume $A(\mathbf{0}) \neq 0$. Then, there exist constants a_k depending on the derivatives of A and ϕ at $\mathbf{0}$ such that, for each $r \geq 1$, the following asymptotic expansion holds:

$$\oint A(\mathbf{z}) e^{-n\phi(\mathbf{z})} d\mathbf{z} = \left(\frac{2\pi}{n}\right)^{d/2} \frac{A(\mathbf{0})}{\sqrt{\det(\mathcal{H})}} \left(1 + \sum_{k=1}^{r-1} a_k n^{-k/2} + \mathcal{O}(n^{-r/2})\right). \quad (5.2)$$

Asymptotic expansions therefore provide a refinement of the expressions obtained from the previous theorem. Indeed, Theorem 14 is a restricted version of Theorem 15, corresponding to retaining only the first-order term of the latter.

One may wonder why it is relevant to compute higher-order terms in the expansion. In Chapter 4, for instance, the rate of convergence in the critical case could not be determined using only the first term; it was necessary to consider the second-order term in the asymptotic expansion to obtain this rate.

However, for the same reasons mentioned above, the critical and mixed cases require an adaptation of the proof to derive these expansions. While obtaining an asymptotic expansion in our setting is not unexpected, it still requires careful modifications and constitutes our third contribution. This adaptation was carried out by Élie de Panafieu.

5.1.2 Summary of our Contributions and Roadmap

To summarize, our contributions are as follows.

- We formulate general sufficient conditions to reduce the multivariate coefficient extraction of large powers to a Laplace integral. This extends to the multivariate setting results known on the univariate case (see [FS09, Section VIII.8]). We first establish a few lemmas in Section 5.2 ensuring that the assumptions of the Laplace method are satisfied, then derive the multivariate large powers theorem in Section 5.3. It is used in Chapter 4 for the subcritical case from Section 4.5.
- We consider in Section 5.4 the case where the saddle point meets a singularity, which is applied both in Section 4.6 and Section 4.7 of Chapter 4.
- In Section 5.5, Élie de Panafieu performs the computation of an asymptotic expansion, improving the precision of our results. It is used in Section 4.9 of Chapter 4.

5.1.3 Preliminaries

As always, we use bold letters to denote vectors, *e.g.* $\mathbf{x} = (x_1, \dots, x_d)$ with $d \in \mathbb{N}_{>0}$. Given two vectors \mathbf{x} and \mathbf{y} of the same dimension d and a scalar z , we define the notations

$$\begin{aligned}\mathbf{x}^{\mathbf{y}} &:= x_1^{y_1} \dots x_d^{y_d}, \\ \mathbf{x}z^{\mathbf{y}} &:= (x_1 z^{y_1}, \dots, x_d z^{y_d}), \\ \log(\mathbf{x}) &:= (\log(x_1), \dots, \log(x_d)).\end{aligned}$$

In particular, we write $\zeta e^{i\theta}$ for the vector $(\zeta_1 e^{i\theta_1}, \dots, \zeta_d e^{i\theta_d})$.

The *cumulant generating function* associated to the generating function $B(\mathbf{z})$ then writes as

$$K(\mathbf{t}) := \log(B(e^{\mathbf{t}})) = \log(B(e^{t_1}, \dots, e^{t_d})).$$

As in Chapter 2, we consider analytic functions. Here is the definition in the multivariate case.

Definition 10 (Multivariate analytic function). Let $U \subset \mathbb{C}^d$ be an open set and let $F : U \rightarrow \mathbb{C}$ be a complex-valued function of d complex variables. We say that F is *analytic* at a point $\boldsymbol{\omega} \in U$ if there exists a convergent power series expansion

$$F(\mathbf{z}) = \sum_{\mathbf{n} \in \mathbb{N}^d} a_{\mathbf{n}} (\mathbf{z} - \boldsymbol{\omega})^{\mathbf{n}},$$

valid in some neighborhood of $\boldsymbol{\omega}$, where we use multi-index notation

$$\mathbf{z} = (z_1, \dots, z_d), \quad \mathbf{n} = (n_1, \dots, n_d), \quad (\mathbf{z} - \boldsymbol{\omega})^{\mathbf{n}} = \prod_{j=1}^d (z_j - \omega_j)^{n_j}.$$

If F is analytic at every point of U , then F is analytic on U .

In practice, we often assume the function of interest to be analytic on a *polydisc*, which is a Cartesian product of discs, and integrate it on a *torus*, *i.e.* a Cartesian product of circles.

To perform approximations as n tends to infinity, we need to consider the gradient of a function $B(\mathbf{z})$ at \mathbf{x} , which is denoted here by $\nabla_B(\mathbf{x})$.

Finally, Section 5.2 relies on the following notions. The *support* of a multivariate power series $B(\mathbf{z})$ is the set

$$S_B := \{\mathbf{n}, [\mathbf{z}^{\mathbf{n}}]B(\mathbf{z}) \neq 0\}$$

and we define

$$\Delta(S_B) := \{\mathbf{n} - \mathbf{m}, \mathbf{n}, \mathbf{m} \in S_B\}.$$

We write $r + q\mathbb{Z}_{\geq 0}$ for the set $\{r + q\ell, \ell \in \mathbb{Z}_{\geq 0}\}$.

5.2 Analytic Conditions for Applying the Saddle Point Method

This section develops a multivariate version of the *Daffodil Lemma* [FS09, Lemma IV.1], providing sufficient conditions for the existence and uniqueness of the saddle point. To this end, we first present general results on multivariate formal power series, introducing the notions of *rank* and *period*. For a broader treatment of the Daffodil lemma, we refer the reader to Flajolet and Sedgewick [FS09] for the univariate case and to Melczer [Mel21] for the multivariate case.

5.2.1 Rank and Periodicity of a Multivariate Generating Function

Lemma 12. *The kernel of any integer matrix has a basis composed of integer vectors.*

Proof. Let M denote a matrix and let its Smith normal form be

$$M = UDV,$$

where U and V are unimodular integer matrices and D is an integer diagonal matrix. The dimension of the kernel of M is equal to the number of zero elements on the diagonal of D . Consider such an element of index j and let $\mathbf{e}^{(j)}$, denote the vector with a 1 in position j and 0's everywhere else. Set $\mathbf{p}^{(j)} = V^{-1}\mathbf{e}^{(j)}$. Since V is unimodular, $\mathbf{p}^{(j)}$ has integer coefficients and is nonzero. We have

$$M\mathbf{p}^{(j)} = UDV\mathbf{p}^{(j)} = UDe^{(j)} = \mathbf{0},$$

so $\mathbf{p}^{(j)}$ belongs to the kernel of M . The invertibility of V ensures that the vectors $\mathbf{p}^{(j)}$ are linearly independent, hence they form a basis of the kernel. \square

Definition 11. We define the *rank* of a formal power series $B(\mathbf{z})$ as the dimension of the real vector space generated by $\Delta(S_B)$. We say that $B(\mathbf{z})$ has *full rank* if its rank is equal to its number of variables.

In particular, a formal power series on at least one variable and that has full rank cannot be a monomial.

Lemma 13. *If the formal power series $B(z_1, \dots, z_d)$ has rank k , then there exist $d - k$ integer vectors $\mathbf{p}^{(1)}, \dots, \mathbf{p}^{(d-k)}$ generating a real vector space of dimension $d - k$ and integers r_1, \dots, r_{d-k} such that for all $j \in [1, d - k]$*

$$B(\mathbf{z}y^{\mathbf{p}^{(j)}}) = y^{r_j} B(\mathbf{z}).$$

Reciprocally, if such vectors $\mathbf{p}^{(j)}$ and integers r_j exist, then $B(\mathbf{z})$ has rank at most k .

Proof. If $B(z_1, \dots, z_d)$ has rank k , by Lemma 12, the vector space orthogonal to $\Delta(S_B)$ has a basis of $d - k$ integer vectors $(\mathbf{p}^{(1)}, \dots, \mathbf{p}^{(d-k)})$. Fix \mathbf{m} in the support S_B of $B(\mathbf{z})$ and define $r_j = \mathbf{m}^T \mathbf{p}^{(j)}$, then for all \mathbf{n} in S_B , we have

$$\mathbf{n}^T \mathbf{p}^{(j)} = r_j, \quad \text{so} \quad B(\mathbf{z}y^{\mathbf{p}^{(j)}}) = y^{r_j} B(\mathbf{z}).$$

Reciprocally, if there are $d - k$ linearly independent integer vectors $(\mathbf{p}^{(1)}, \dots, \mathbf{p}^{(d-k)})$ and integers r_1, \dots, r_{d-k} such that

$$B(\mathbf{z}y^{\mathbf{p}^{(j)}}) = y^{r_j} B(\mathbf{z}),$$

then each $\mathbf{p}^{(j)}$ is in the vector space orthogonal to $\Delta(S_B)$, so this vector space has dimension at most k . □

In this thesis, we will consider a polynomial P having full rank. It is however worth noticing that coefficient extraction in a power series that does not have full rank can be simplified as follows. Say we are interested in the coefficient extraction $[\mathbf{z}^{\mathbf{n}}]B(\mathbf{z})$ of a power series $B(\mathbf{z})$ that does not have full rank. Let \mathbf{p} and r denote a nonzero integer vector and an integer such that

$$B(\mathbf{z}y^{\mathbf{p}}) = y^r B(\mathbf{z}).$$

Since \mathbf{p} is nonzero, there exists j such that $p_j \neq 0$. Then for any $\mathbf{n} \in S_B$, we have

$$n_j = \frac{r - \sum_{\ell \neq j} p_\ell n_\ell}{p_j},$$

so n_j is uniquely determined by $(n_\ell)_{\ell \neq j}$ and

$$[\mathbf{z}^n]B(\mathbf{z}) = \begin{cases} 0 & \text{if } \mathbf{n}^T \mathbf{p} \neq r, \\ [z_1^{n_1} \cdots z_{j-1}^{n_{j-1}} z_{j+1}^{n_{j+1}} \cdots z_d^{n_d}]B(z_1, \dots, z_{j-1}, 1, z_{j+1}, \dots, z_d) & \text{otherwise.} \end{cases}$$

Thus, coefficient extraction in a power series that does not have full rank reduces to a coefficient extraction in a power series with fewer variables.

Consider a univariate probability generating function $B(z) = \sum_n b_n z^n$, meaning $b_n \geq 0$ for all n and $B(1) = 1$. Its *cumulant generating function* $K(t)$ is defined as

$$K(t) = \log(B(e^t)).$$

A classic result of probability theory states this function is convex on its domain of definition (direct application of Höler's inequality, see *e.g.* [DZ09]). Our next lemma is a multivariate variant of this result. The proof is adapted from [FS09, Note IV.46].

Lemma 14 (Convexity of the cumulant generating function). *Consider a series $B(\mathbf{z})$ with nonnegative coefficients, full rank, and analytic on a neighborhood Ω of $\mathbf{0}$. Then its cumulant generating function*

$$K : \mathbf{t} \mapsto \log(B(e^{\mathbf{t}}))$$

is strictly convex on $\{\mathbf{t} \in \mathbb{R}^d, e^{\mathbf{t}} \in \Omega\}$.

Proof. Let \mathbf{X} denote the vector of random variables parametrized by $\mathbf{x} \in \mathbb{R}_{>0}^d \cap \Omega$ with distribution

$$\mathbb{P}(\mathbf{X} = \mathbf{n}) = \frac{[\mathbf{z}^n]B(\mathbf{z})\mathbf{x}^{\mathbf{n}}}{B(\mathbf{x})}.$$

By construction, those probabilities sum to 1. This is the so-called *Boltzmann distribution* associated to the generating function $B(\mathbf{z})$ (see [Duc+04]). We now follow the classical proof of convexity of the cumulant generating function of a multivariate random variable, then prove that the assumption that $B(\mathbf{z})$ has full rank implies strict convexity.

The moment generating function of \mathbf{X} is

$$\mathbb{E}(e^{\mathbf{t}^T \mathbf{X}}) = \frac{B(\mathbf{x} e^{\mathbf{t}})}{B(\mathbf{x})}.$$

Observe that the cumulant generating function of \mathbf{X} is equal to

$$\log(\mathbb{E}(e^{t^T \mathbf{X}})) = \log(B(\mathbf{x}e^t)) - \log(B(\mathbf{x})) = K(\mathbf{t} + \log(\mathbf{x})) - \log(B(\mathbf{x})).$$

The vector of the means is

$$\mathbb{E}(\mathbf{X}) = \frac{\text{diag}(\mathbf{x})\nabla_B(\mathbf{x})}{B(\mathbf{x})}.$$

It is equal to the gradient of $K(\mathbf{t})$ at $\mathbf{t} = \log(\mathbf{x})$. The covariance matrix is

$$\text{Var}(\mathbf{X}) = \mathbb{E}((\mathbf{X} - \mathbb{E}(\mathbf{X}))^T(\mathbf{X} - \mathbb{E}(\mathbf{X}))) = \mathbb{E}(\mathbf{X}\mathbf{X}^T) - \mathbb{E}(\mathbf{X})\mathbb{E}(\mathbf{X})^T.$$

It is equal to the Hessian of $K(\mathbf{t})$ at $\mathbf{t} = \log(\mathbf{x})$.

For any vector $\mathbf{u} \in \mathbb{R}^d$,

$$\mathbf{u}^T \text{Var}(\mathbf{X})\mathbf{u} = \mathbb{E}(((\mathbf{X} - \mathbb{E}(\mathbf{X}))^T \mathbf{u})^2),$$

so $\text{Var}(\mathbf{X})$ is positive semi-definite. Consider a vector \mathbf{u} such that

$$0 = \mathbf{u}^T \text{Var}(\mathbf{X})\mathbf{u} = \mathbb{E}(((\mathbf{X} - \mathbb{E}(\mathbf{X}))^T \mathbf{u})^2).$$

This implies $(\mathbf{X} - \mathbb{E}(\mathbf{X}))^T \mathbf{u} = \mathbf{0}$ so, for any \mathbf{n}, \mathbf{m} in the support of $B(\mathbf{x})$,

$$(\mathbf{n} - \mathbf{m})^T \mathbf{u} = 0.$$

Our assumption that $B(\mathbf{z})$ has full rank implies $\mathbf{u} = \mathbf{0}$, which proves that $\text{Var}(\mathbf{X})$ is positive definite. Since $\text{Var}(\mathbf{X})$ is equal to the Hessian of $K(\mathbf{t})$ at $\log(\mathbf{x})$, we deduce that $K(\mathbf{t})$ is strictly convex on $\{\mathbf{t} \in \mathbb{R}^d, e^t \in \Omega\}$. \square

We recall below the definition of q -periodicity from Flajolet et al. [FSZ91, Section 4.2] for univariate formal power series.

Definition 12. A univariate generating function $B(z)$ is said to be q -periodic if there exist an integer r and a formal power series $C(z)$ such that

$$B(z) = z^r C(z^q).$$

Equivalently, the support S_B of $B(z)$ satisfies

$$S_B \subseteq r + q\mathbb{Z}_{\geq 0}.$$

The series $B(\mathbf{z})$ is said to be *aperiodic* if it is not q -periodic for any $q \geq 2$.

Let us now extend this definition to multivariate formal power series.

Definition 13. A multivariate generating function $B(z_1, \dots, z_d)$ is said to be q -periodic if there exist nonnegative integers p_1, \dots, p_d , not all divisible by q , such that $B(y^{p_1}, \dots, y^{p_d})$ is a q -periodic univariate formal power series. An equivalent formulation on the support S_B of $B(\mathbf{z})$ is that there exists a nonnegative integer r such that

$$\{\mathbf{p}^T \mathbf{n}, \mathbf{n} \in S_B\} \subseteq r + q\mathbb{Z}_{\geq 0}. \quad (5.3)$$

Yet another equivalent formulation is that there exists a formal power series $E(\mathbf{z}, y)$, integers p_1, \dots, p_d not all divisible by q and an integer r such that

$$B(z_1 y^{p_1}, \dots, z_d y^{p_d}) = y^r E(\mathbf{z}, y^q).$$

We say that $B(\mathbf{z})$ is *aperiodic* if it is not q -periodic for any $q \geq 2$.

Note that our definition of aperiodicity is related but distinct from the one from [Mel21, Definition 5.9]. Our next result presents a simple sufficient condition for a formal multivariate power series to have full rank and be aperiodic.

Proposition 1. Let \mathbf{e}_j denote the vector with a 1 at position j , and 0s everywhere else. Consider a formal power series $B(z_1, \dots, z_d)$ whose support contains $\mathbf{0}, \mathbf{e}_1, \dots, \mathbf{e}_d$. Then $B(\mathbf{z})$ has full rank and is aperiodic.

Proof. Since the support of $B(\mathbf{z})$ contains $\mathbf{0}$, we have $S_B \subseteq \Delta(S_B)$. Thus, $\Delta(S_B)$ contains d independent vectors $\mathbf{e}_1, \dots, \mathbf{e}_d$, so it has rank d and $B(\mathbf{z})$ has full rank. Consider an integer vector \mathbf{p} and integers r and $q \geq 2$. Assume

$$\{\mathbf{p}^T \mathbf{n}, \mathbf{n} \in S_B\} \subseteq r + q\mathbb{Z}_{\geq 0}.$$

For $\mathbf{n} = \mathbf{0}$, we deduce that q divides r . For $\mathbf{n} = \mathbf{e}_j$, we deduce that q divides $p_j - r$, so p_j is a multiple of q . Thus, \mathbf{p} is necessarily a multiple of q and $B(\mathbf{z})$ cannot be q -periodic, so $B(\mathbf{z})$ is aperiodic. \square

With this background in place, we are now ready to state and prove a multivariate extension of the Daffodil Lemma [FS09, Lemma IV.1].

5.2.2 The Daffodil Lemma

The Daffodil lemma states that the absolute value of a univariate generating function with nonnegative coefficients reaches its maximum at a unique point on any disk contained in its domain of convergence if and only if it is aperiodic. Our following lemma extends this result to the multivariate setting. This is reminiscent of [Mel21, Proposition 5.5] that provides sufficient conditions for the *minimal point* (a central concept of singularity analysis, defined there) of rational functions to have positive real coordinates.

Lemma 15 (Multivariate daffodil lemma). *Consider a multivariate power series $B(z_1, \dots, z_d)$ with nonnegative coefficients and a vector $\zeta \in \mathbb{R}_{>0}^d$ such that the domain of convergence of $B(\mathbf{z})$ contains the closed polydisc of radius ζ*

$$D(\zeta) := \{\mathbf{z}, |z_1| \leq \zeta_1 \wedge \dots \wedge |z_d| \leq \zeta_d\}.$$

Then for all $\mathbf{z} \in D(\zeta)$, we have $|B(\mathbf{z})| \leq B(\zeta)$.

Assume furthermore that $B(\mathbf{z})$ has full rank. Then $|B(\mathbf{z})|$ has a unique maximum on $D(\zeta)$ if and only if it is aperiodic.

Proof. Using the positivity of the coefficients of $B(\mathbf{z})$, triangular inequality yields for any $\mathbf{x} \in D(\zeta)$

$$|B(\mathbf{x})| = \left| \sum_{\mathbf{n}} b_{\mathbf{n}} \mathbf{x}^{\mathbf{n}} \right| \leq \sum_{\mathbf{n}} b_{\mathbf{n}} |\mathbf{x}|^{\mathbf{n}} \leq \sum_{\mathbf{n}} b_{\mathbf{n}} \zeta^{\mathbf{n}} = B(\zeta).$$

Assume $|B(\mathbf{x})| = B(\zeta)$, then $(|x_1|, \dots, |x_d|) = \zeta$, so there exists a vector $\mathbf{t} \in [0, 1)^d$ such that for all j , we have $x_j = \zeta_j e^{2i\pi t_j}$. The case $\mathbf{t} = \mathbf{0}$ corresponds to $\mathbf{x} = \zeta$. Let us assume first $\mathbf{t} \neq \mathbf{0}$, and prove that there exists $q \geq 1$ such that $B(\mathbf{z})$ is q -periodic. The equality $|B(\mathbf{x})| = B(\zeta)$ implies

$$\left| \sum_{\mathbf{n}} b_{\mathbf{n}} \zeta^{\mathbf{n}} e^{2i\pi \mathbf{n}^T \mathbf{t}} \right| = \sum_{\mathbf{n}} b_{\mathbf{n}} \zeta^{\mathbf{n}},$$

so by the triangular inequality, for all \mathbf{n} in the support S_B of $B(\mathbf{z})$, the complex numbers $e^{2i\pi \mathbf{n}^T \mathbf{t}}$ are aligned. This is equivalent with the existence of $t_0 \in [0, 1)$ such that for all $\mathbf{n} \in S$, we have

$$\mathbf{n}^T \mathbf{t} = t_0 \pmod{1}.$$

Then for all $\mathbf{n} \in \Delta(S_B)$, we have

$$\mathbf{n}^T \mathbf{t} = 0 \pmod{1}.$$

Thus, for all $\mathbf{n} \in \Delta(S_B)$, there exist integers $k_{\mathbf{n}}$ such that

$$\mathbf{n}^T \mathbf{t} = k_{\mathbf{n}}.$$

Let $\Delta(S_B)'$ denote the set of vectors \mathbf{n} from $\Delta(S_B)$ with an additional element $-k_{\mathbf{n}}$ at the bottom, and let \mathbf{t}' denote the vector \mathbf{t} with an additional element 1 at the bottom, then the previous equality is equivalent with

$$\mathbf{n}^T \mathbf{t}' = 0$$

for all $\mathbf{n} \in \Delta(S_B)'$. Since $B(\mathbf{z})$ has full rank, $\Delta(S_B)$ generates a vector space of dimension d , so the vector space orthogonal to $\Delta(S_B)$ has dimension at most 1. It contains a nonzero vector, so the dimension is exactly 1. By Lemma 12, there exists a nonzero integer vector \mathbf{u} orthogonal to $\Delta(S_B)'$. We have $t'_{d+1} = 1$ and \mathbf{t}' proportional to the integer vector \mathbf{u} , so \mathbf{t} is a vector of rational values. Recall that $\mathbf{t} \in [0,1]^d$ and is nonzero. Thus, there exist $q \in \mathbb{Z}_{\geq 2}$ and $\mathbf{p} \in \mathbb{Z}_{\geq 0}^d$ with q that does not divide all p_j , such that $\mathbf{t} = \mathbf{p}/q$, so for all $\mathbf{n} \in \Delta(S_B)$,

$$\mathbf{n}^T \mathbf{p} = q k_{\mathbf{n}}.$$

Fix $r = \min_{\mathbf{n} \in S_B} (\mathbf{n}^T \mathbf{p})$, then for all $\mathbf{n} \in S_B$,

$$\mathbf{n}^T \mathbf{p} - r = q k_{\mathbf{n}}.$$

Thus, q divides $\mathbf{n}^T \mathbf{p} - r$. This implies the existence of a formal univariate power series $C(y)$ such that

$$B(y^{p_1}, \dots, y^{p_d}) = y^r C(y^q), \quad (5.4)$$

so $B(\mathbf{z})$ is q -periodic.

Reciprocally, if $B(\mathbf{z})$ is q -periodic for some $q \geq 2$, there exist nonnegative integers r, p_1, \dots, p_d , with q not dividing all p_j , and a formal power series $C(y)$ such that Equation (5.4) holds. Then for all $\mathbf{n} \in S_B$, the integer q divides $\mathbf{n}^T \mathbf{p} - r$, so $e^{2i\pi \mathbf{n}^T \mathbf{p}/q} = e^{2i\pi r/q}$. For any $\zeta \in \mathbb{R}_{>0}^d$ such that the domain of convergence of $B(\mathbf{z})$ contains the polydisc of radius ζ , setting $\mathbf{x} = \zeta e^{2i\pi \mathbf{p}/q}$, we have

$$|B(\mathbf{x})| = \left| \sum_{\mathbf{n} \in S_B} b_{\mathbf{n}} \zeta^{\mathbf{n}} e^{2i\pi \mathbf{p}^T \mathbf{n}/q} \right| = \left| \sum_{\mathbf{n} \in S_B} b_{\mathbf{n}} \zeta^{\mathbf{n}} e^{2i\pi r} \right| = \left| \sum_{\mathbf{n} \in S_B} b_{\mathbf{n}} \zeta^{\mathbf{n}} \right| = B(\zeta),$$

where the last equality comes from the fact that the coefficients of $B(\mathbf{z})$ are nonnegative. Thus, on the torus of radius ζ , at the point \mathbf{x} , which is distinct from ζ , we have $|B(\mathbf{x})| = B(\zeta)$. \square

In this thesis, we only consider aperiodic polynomials. Nevertheless, it is worth briefly discussing the periodic case, as it reveals useful simplifications in coefficient extraction. Indeed, coefficient extraction in a periodic power series can be simplified as follows. Suppose we are interested in the coefficient extraction $[z^n]B(z)$. If $B(z)$ does not have full rank, we saw after Lemma 13 how to simplify the coefficient extraction. Assume now $B(z)$ has full rank, is analytic at $\mathbf{0}$, and is q -periodic for some $q \geq 2$. Then there exist nonnegative integers r, p_1, \dots, p_d , with q not dividing all p_j , and a power series $E(z, y)$ such that

$$B(\mathbf{z}y^{\mathbf{p}}) = y^r E(\mathbf{z}, y^q).$$

We have

$$[z^n]B(\mathbf{z}) = [z^n y^{\mathbf{p}^T \mathbf{n}}]B(z_1 y^{p_1}, \dots, z_d y^{p_d}) = [z^n y^{\mathbf{p}^T \mathbf{n}}]y^r E(\mathbf{z}, y^q).$$

Let j be such that $p_j \neq 0$ (which must exist, otherwise q would divide all p_i). Let $\mathbf{n}_{\setminus j}$ denote the vector $(n_1, \dots, n_{j-1}, n_{j+1}, \dots, n_d)$ and similarly for $\mathbf{z}_{\setminus j}$. Then $B(\mathbf{z})$ having full rank implies

$$[z^n y^{\mathbf{p}^T \mathbf{n}}]B(\mathbf{z}y^{\mathbf{p}}) = [z_{\setminus j}^{\mathbf{n}_{\setminus j}} y^{\mathbf{p}^T \mathbf{n}}]B(z_1 y^{p_1}, \dots, z_{j-1} y^{p_{j-1}}, y^{p_j}, z_{j+1} y^{p_{j+1}}, \dots, z_d y^{p_d}).$$

Define

$$D(\mathbf{z}_{\setminus j}, y) := E(z_1, \dots, z_{j-1}, 1, z_{j+1}, \dots, z_d, y),$$

then

$$[z^n]B(\mathbf{z}) = \begin{cases} 0 & \text{if } q \text{ does not divide } \mathbf{p}^T \mathbf{n} - r, \\ [z_{\setminus j}^{\mathbf{n}_{\setminus j}} y^{(\mathbf{p}^T \mathbf{n} - r)/q}]D(\mathbf{z}_{\setminus j}, y) & \text{otherwise.} \end{cases}$$

Thus, when $|B(\mathbf{z})|$ reaches its maximum on a polydisc at several points, the coefficient extraction $[z^n]B(\mathbf{z})$ can be simplified. This shows that the location of the maximum modulus of $B(\mathbf{z})$ on a torus is essentially independent of the choice of torus, as long as the torus is centered at the origin and lies within the domain of convergence. The following corollary formalizes this property.

Corollary 3. *Consider a power series $B(\mathbf{z})$ and two complex torii T_1, T_2 centered at the origin and contained in the domain of convergence of $B(\mathbf{z})$. If $|B(\mathbf{z})|$ reaches its maximum on T_1 at a unique point, then it reaches its maximum on T_2 at a unique point as well.*

To conclude, we provided a formal framework of sufficient conditions for the existence and uniqueness of the saddle point. Consequently, to apply the multivariate Laplace theorem, it suffices to verify that the function of interest satisfies the conditions of Lemma 15, without explicitly computing the saddle point. This considerably simplifies the analysis: checking these conditions typically require only a few steps, as illustrates Section 4.4 of Chapter 4.

With these conditions in place, we can now state the main theorems explicitly, beginning with the case where the saddle point does not coincide with the singularity of the generating function, namely the large powers framework.

5.3 Large Powers

When the saddle point does not coincide with the singularity, we are in the *large powers* framework, so called because the generating function is raised to a large power. In this setting, exemplified by the subcritical case of Section 4.5 in Chapter 4, the contour of integration can pass through the saddle point without intersecting the singularity, and Theorem 14 provides the key asymptotic result.

Therefore, in this section, we restate Theorem 14 under the sufficient conditions established above. The result is first stated for coefficient extraction at index λn with a fixed rational vector λ , and then extended to the more general case of indices given by sequences of real vectors of the form $\kappa_n = \lambda n + \mathcal{O}(n^\delta)$, fitting the framework of Chapter 4.

Theorem 16 (Multivariate large powers theorem). *Consider a power series $B(z_1, \dots, z_d)$ with nonnegative coefficients, full rank (Definition 11), aperiodic (Definition 13) and convergent on a neighborhood Ω of 0. Let $K(\mathbf{t}) = \log(B(e^{\mathbf{t}}))$ denote its cumulant generating function. Let λ denote a vector of positive rational numbers such that the function $K(\mathbf{t}) - \lambda^T \mathbf{t}$ does not tend to its infimum on the boundary of $\Psi = \{\mathbf{t} \in \mathbb{R}^d, e^{\mathbf{t}} \in \Omega\}$. Let τ denote the unique point of Ψ where $K(\mathbf{t}) - \lambda^T \mathbf{t}$ reaches its minimum and \mathcal{H} denote the Hessian matrix of $K(\mathbf{t})$ at τ . Set $\zeta = e^\tau$. Consider a power series $A(\mathbf{z})$ analytic on the torus of radius ζ and satisfying $A(\zeta) \neq 0$. Then, considering large values of n such that λn is a vector of integers,*

$$[z^{\lambda n}]A(\mathbf{z})B(\mathbf{z})^n \sim \frac{A(\zeta)}{\sqrt{(2\pi n)^d \det(\mathcal{H})}} \frac{B(\zeta)^n}{\zeta^{\lambda n}}.$$

Proof. Lemma 14 ensures that $K(\mathbf{t})$ is strictly convex, so $K(\mathbf{t}) - \lambda^T \mathbf{t}$ is strictly convex as well. Since it does not reach its minimum on the boundary of Ψ , there exists a unique point τ in Ψ where $K(\mathbf{t}) - \lambda^T \mathbf{t}$ reaches its minimum. We write the coefficient extraction as a Cauchy integral.

$$[z^{\lambda n}]A(\mathbf{z})B(\mathbf{z})^n = \frac{1}{(2i\pi)^d} \oint A(\mathbf{z})B(\mathbf{z})^n \frac{d\mathbf{z}}{z^{\lambda n + \mathbf{1}}}.$$

The analyticity of $B(z)$ at ζ and its nonnegative coefficients ensure that $B(z)$ is analytic on the torus of radius ζ . We choose for the integration domain the torus of radius ζ

$$[z^{\lambda n}]A(z)B(z)^n = \frac{1}{(2\pi)^d} \int_{\theta \in [-\pi, \pi]^d} A(\zeta e^{i\theta}) B(\zeta e^{i\theta})^n \frac{d\theta}{\zeta^{\lambda n} e^{in\lambda^T \theta}}.$$

Introducing

$$\phi(\theta) := i\lambda^T \theta - \log \left(\frac{B(\zeta e^{i\theta})}{B(\zeta)} \right),$$

we have

$$[z^{\lambda n}]A(z)B(z)^n = \frac{B(\zeta)^n}{\zeta^{\lambda n}} \frac{1}{(2\pi)^d} \int_{\theta \in [-\pi, \pi]^d} A(\zeta e^{i\theta}) e^{-n\phi(\theta)} d\theta.$$

Since $|B(\zeta e^{i\theta})|$ reaches its maximal value $B(\zeta)$ only at $\theta = 0$, the real part of $\phi(\theta)$ is positive except at $\theta = 0$. The Hessian matrix of $\phi(\theta)$ at $\theta = 0$ is equal to the Hessian matrix \mathcal{H} of $K(\mathbf{t})$ at $\mathbf{t} = \log(\zeta)$, and is nonsingular according to Lemma 14. The result of the theorem follows by application of the Laplace method, recalled in Theorem 14. \square

As mentioned above, applications often require greater flexibility than provided by the previous result, notably in Chapter 4. Instead of extracting the coefficient at index λn for some fixed rational vector λ , we may consider indices $\kappa_n \sim \lambda n$. This allows us to handle cases where λ contains irrational values or, as in Chapter 4, when the index involves a ceiling function. To obtain this result, we follow the proof of Theorem 14, with slight modifications.

Theorem 17 (Multivariate large powers theorem, more general setting). *Consider a power series $B(z_1, \dots, z_d)$ with nonnegative coefficients, full rank, aperiodic and convergent on a neighborhood Ω of 0. Let $K(\mathbf{t}) = \log(B(e^{\mathbf{t}}))$ denote its cumulant generating function. Let λ denote a vector of positive rational numbers such that the function $K(\mathbf{t}) - \lambda^T \mathbf{t}$ does not tend to its infimum on the boundary of $\Psi = \{\mathbf{t} \in \mathbb{R}^d, e^{\mathbf{t}} \in \Omega\}$. Let τ denote the unique point of Ψ where $K(\mathbf{t}) - \lambda^T \mathbf{t}$ reaches its minimum and \mathcal{H} denote the Hessian matrix of $K(\mathbf{t})$ at τ . Set $\zeta = e^\tau$. Consider a power series $A(z)$ analytic on the torus of radius ζ and satisfying $A(\zeta) \neq 0$. Consider $\delta < 1/2$ and a sequence of vectors of real values*

$$\kappa_n = \lambda n + \mathcal{O}(n^\delta).$$

We have

$$[z^{\kappa_n}]A(z)B(z)^n \sim \frac{A(\zeta)}{\sqrt{(2\pi n)^d \det(\mathcal{H})}} \frac{B(\zeta)^n}{\zeta^{\kappa_n}}.$$

Proof. We write again the coefficient extraction as a Cauchy integral on the torus of radius ζ

$$[z^{\kappa_n}]A(z)B(z)^n = \frac{1}{(2\pi)^d} \int_{(-\pi, \pi]^d} A(\zeta e^{i\theta}) B(\zeta e^{i\theta})^n \frac{d\theta}{\zeta^{\kappa_n} e^{i\kappa_n^T \theta}}$$

and introduce

$$\phi(\theta) := i\lambda^T \theta - \log \left(\frac{B(\zeta e^{i\theta})}{B(\zeta)} \right) \quad (5.5)$$

to rewrite it as

$$[z^{\kappa_n}]A(z)B(z)^n = \frac{B(\zeta)^n}{\zeta^{\kappa_n}} \frac{1}{(2\pi)^d} \int_{(-\pi, \pi]^d} A(\zeta e^{i\theta}) e^{-n\phi(\theta)} \frac{d\theta}{e^{i(\kappa_n - \lambda n)^T \theta}}. \quad (5.6)$$

Central part. We introduce a positive sequence ε_n converging to 0 and specified later. The central part of the integral is defined as

$$C_n := \frac{1}{(2\pi)^d} \int_{(-\varepsilon_n, \varepsilon_n)^d} A(\zeta e^{i\theta}) e^{-n\phi(\theta)} \frac{d\theta}{e^{i(\kappa_n - \lambda n)^T \theta}}.$$

Since $\phi(\mathbf{0}) = \mathbf{0}$ and, by definition of ζ , the gradient of $\phi(\theta)$ also vanishes at $\mathbf{0}$, the Taylor expansion of $\phi(\theta)$ is

$$\phi(\theta) = -\frac{1}{2} \theta^T \mathcal{H} \theta + \mathcal{O}(\|\theta\|^3). \quad (5.7)$$

For $\theta \in (-\varepsilon_n, \varepsilon_n)$, we have

$$n\phi(\theta) = -\frac{n}{2} \theta^T \mathcal{H} \theta + \mathcal{O}(n\varepsilon_n^3).$$

We will choose ε_n to ensure $n\varepsilon_n^3 \rightarrow 0$ so that the term in the \mathcal{O} is negligible. At the same time, we want $n^\delta \varepsilon_n \rightarrow 0$, so $e^{i(\kappa_n - \lambda n)^T \theta} \sim 1$ uniformly for $\theta \in (-\varepsilon_n, \varepsilon_n)$.

The change of variable $\mathbf{t} = \sqrt{n}\theta$ is applied

$$C_n = \frac{n^{-d/2}}{(2\pi)^d} \int_{(-\sqrt{n}\varepsilon_n, \sqrt{n}\varepsilon_n)^d} A(\zeta e^{in^{-1/2}\mathbf{t}}) e^{-\mathbf{t}^T \mathcal{H} \mathbf{t} / 2 + \mathcal{O}(n\varepsilon_n^3)} d\mathbf{t}.$$

We have $A(\zeta e^{in^{-1/2}\mathbf{t}}) \rightarrow A(\zeta)$ and $n\varepsilon_n^3 \rightarrow 0$, so

$$C_n \sim A(\zeta) \frac{n^{-d/2}}{(2\pi)^d} \int_{(-\sqrt{n}\varepsilon_n, \sqrt{n}\varepsilon_n)^d} e^{-\mathbf{t}^T \mathcal{H} \mathbf{t} / 2} d\mathbf{t}.$$

We recognise a Cauchy integral and choose ε_n such that $\sqrt{n}\varepsilon_n \rightarrow +\infty$, so

$$C_n \sim A(\zeta) \frac{n^{-d/2}}{(2\pi)^d} \int_{(-\infty, +\infty)^d} e^{-t^T \mathcal{H} t / 2} dt \sim \frac{A(\zeta)}{\sqrt{(2\pi n)^d \det(\mathcal{H})}}.$$

To satisfy the three constraints on ε_n , we define it as follows.

$$\varepsilon_n = n^{-\beta} \text{ with } \beta \in (\max(\delta, 1/3), 1/2).$$

Tail. The tail corresponds to $(-\pi, \pi]^d \setminus (-\varepsilon_n, \varepsilon_n)^d$. The tail part of the integral (5.6) is bounded by

$$\int_{(-\pi, \pi]^d \setminus (-\varepsilon_n, \varepsilon_n)^d} |A(\zeta e^{i\theta}) e^{-n\phi(\theta)}| d\theta \leq \left(\sup_{\theta \in (-\pi, \pi)^d} |A(\zeta e^{i\theta})| \right) \int_{(-\pi, \pi]^d \setminus (-\varepsilon_n, \varepsilon_n)^d} e^{-n \operatorname{Re}(\phi(\theta))} d\theta.$$

We will prove in the next paragraph that for all large enough n , the minimum of $\operatorname{Re}(\phi(\theta))$ on the tail is reached at some points from $[-\varepsilon_n, \varepsilon_n]^d \setminus (-\varepsilon_n, \varepsilon_n)^d$. Thus, the tail is bounded by

$$\mathcal{O}(1) \sup_{\theta \in [-\varepsilon_n, \varepsilon_n]^d \setminus (-\varepsilon_n, \varepsilon_n)^d} e^{-n \operatorname{Re}(\phi(\theta))}.$$

We replace $\phi(\theta)$ with its Taylor expansion

$$-n \operatorname{Re}(\phi(\theta)) = -\frac{n}{2} \theta^T \mathcal{H} \theta + \mathcal{O}(n\varepsilon_n^3).$$

Our choice of ε_n ensures the \mathcal{O} tends to 0. We divide θ by ε_n to bound the tail by

$$\mathcal{O}(1) \sup_{\theta \in [-\varepsilon_n, \varepsilon_n]^d \setminus (-\varepsilon_n, \varepsilon_n)^d} \exp \left(-\frac{n\varepsilon_n^2}{2} (\varepsilon_n^{-1} \theta)^T \mathcal{H} (\varepsilon_n^{-1} \theta) \right).$$

The vector $\varepsilon_n^{-1} \theta$ has at least one coefficient equal to -1 or 1 , so it stays bounded away from $\mathbf{0}$. Since \mathcal{H} is positive definite, there exist a positive constant c such that $(\varepsilon_n^{-1} \theta)^T \mathcal{H} (\varepsilon_n^{-1} \theta) \geq c$, so the tail is bounded by

$$\mathcal{O}(1) \exp \left(-c \frac{n\varepsilon_n^2}{2} \right).$$

Since $n\varepsilon_n^2 \rightarrow +\infty$, this converges exponentially fast to 0, so the tail is negligible compared to the central part of the integral.

Now we need to locate the minimum of $\operatorname{Re}(\phi(\theta))$ on the tail. By definition of $\phi(\theta)$, we have

$$\operatorname{Re}(\phi(\theta)) = \operatorname{Re} \left(i\lambda^T \theta - \log \left(\frac{B(\zeta e^{i\theta})}{B(\zeta)} \right) \right) = \log(B(\zeta)) - \log(|B(\zeta e^{i\theta})|).$$

Thus, the minima of $\operatorname{Re}(\phi(\boldsymbol{\theta}))$ correspond to maxima of $|B(\zeta e^{i\boldsymbol{\theta}})|$. By assumption, on the polydisc of radius ζ , the maximum of $|B(\boldsymbol{z})|$ is reached only at $\boldsymbol{z} = \zeta$, so the minimum of $\operatorname{Re}(\phi(\boldsymbol{\theta}))$ is reached only at $\boldsymbol{\theta} = \mathbf{0}$. As all points of the central part tend to ζ with n , this minimum tends to $\operatorname{Re}(\phi(\mathbf{0})) = 0$. By contradiction, let us assume that there is an infinite set of indices n for which there exists a point in $(-\pi, \pi)^d \setminus [-\varepsilon_n, \varepsilon_n]^d$ where $\operatorname{Re}(\phi(\boldsymbol{\theta}))$ reaches its minimum on the tail $(-\pi, \pi)^d \setminus (-\varepsilon_n, \varepsilon_n)^d$. We extract a converging subsequence \boldsymbol{u}_n . The limit \boldsymbol{v} must be $\mathbf{0}$, otherwise $\lim_n \operatorname{Re}(\phi(\boldsymbol{u}_n)) = \operatorname{Re}(\phi(\boldsymbol{v})) > 0$, which contradicts the fact that the minimum on the tail tends to 0. The gradient of $\operatorname{Re}(\phi(\boldsymbol{\theta}))$ vanishes at each $\boldsymbol{\theta} = \boldsymbol{u}_n$. From Equation (5.7), we deduce that the gradient of $\operatorname{Re}(\phi(\boldsymbol{\theta}))$ at $\boldsymbol{\theta} = \boldsymbol{u}_n$ is proportional to $\mathcal{H}\boldsymbol{u}_n + \mathcal{O}(\|\boldsymbol{u}_n\|^2)$. Let q_n denote the minimum of the absolute value of all coefficients of \boldsymbol{u}_n , then for all n

$$\mathcal{H} \frac{\boldsymbol{u}_n}{q_n} + \mathcal{O}(\|\boldsymbol{u}_n\|) = \mathbf{0}.$$

Left-multiplying by \mathcal{H}^{-1} , we deduce that \boldsymbol{u}_n/q_n converges to $\mathbf{0}$, which is impossible as at least one of its coefficients is equal to 1. This concludes the proof that for all large enough n , the minimum of $\operatorname{Re}(\phi(\boldsymbol{\theta}))$ on $(-\pi, \pi]^d \setminus (-\varepsilon_n, \varepsilon_n)^d$ belongs to $[-\varepsilon_n, \varepsilon_n]^d \setminus (-\varepsilon_n, \varepsilon_n)^d$. \square

5.4 Combining Large Powers and Singularity Analysis

When some coordinates of the saddle point coincide with singularities, the theorem stated in the previous section must be adapted. This situation arises, for instance, in the subcritical case discussed in Section 4.6 and the mixed case in Section 4.7 of Chapter 4, where the contour of integration cannot pass through the saddle point without intersecting a singularity.

The following result generalizes Theorem 17 to the case where the saddle point coincides with singularities. It provides the analytic foundation for Theorems 6, 9 and 11 and can be seen as a multivariate combination of the large powers theorem [FS09, Theorem VIII.8] and singularity analysis [FS09, Section VI]. The presence of large powers simplifies the singularity analysis and avoids the more intricate complications of full multivariate singularity analysis [PW13]. A more precise asymptotic expansion of this result will be derived in Section 5.5.

Theorem 18. *Consider a power series $B(z_1, \dots, z_d)$ with nonnegative coefficients, full rank, aperiodic and convergent on a neighborhood Ω of 0. Let $K(\mathbf{t}) = \log(B(e^{\mathbf{t}}))$ denote its cumulant generating function. Let $\boldsymbol{\lambda}$ denote a vector of positive real values such that the function $K(\mathbf{t}) - \boldsymbol{\lambda}^T \mathbf{t}$ does not tend to its infimum on the boundary of $\Psi = \{\mathbf{t} \in \mathbb{R}^d, e^{\mathbf{t}} \in \Omega\}$. Let $\boldsymbol{\tau}$ denote the unique point of Ψ where $K(\mathbf{t}) - \boldsymbol{\lambda}^T \mathbf{t}$ reaches its minimum. Set $\zeta = e^{\boldsymbol{\tau}}$.*

We assume $\zeta_j \leq 1$ for all j and denote by \mathcal{S} the set of indices such that $\zeta_j < 1$, and by \mathcal{C} the corresponding set for $\zeta_j = 1$. Note that by assumption, we have $\mathcal{S} \cup \mathcal{C} = \{1, \dots, d\}$. Let \mathcal{H} denote the Hessian matrix of $K(\mathbf{t})$ at $\boldsymbol{\tau}$ and \mathcal{M} the submatrix of \mathcal{H}^{-1} corresponding to the rows and column in \mathcal{C} . Consider a power series $A(\mathbf{z})$ analytic on the torus of radius ζ and satisfying $A(\zeta) \neq 0$. Consider a real value $\delta < 1/2$ and a sequence of vectors of real values

$$\boldsymbol{\kappa}_n = \boldsymbol{\lambda}n + \mathcal{O}(n^\delta).$$

Then

$$[z^{\boldsymbol{\kappa}_n}] \frac{A(\mathbf{z})B(\mathbf{z})^n}{\prod_{j \in \mathcal{S} \cup \mathcal{C}} (1 - z_j)} \sim \frac{B(\zeta)^n}{\zeta^{\boldsymbol{\kappa}_n}} \frac{A(\zeta)}{\prod_{j \in \mathcal{S}} (1 - \zeta_j)} \frac{1}{\sqrt{(2\pi)^d n^{|\mathcal{S}|} \det(\mathcal{H})}} \int_{(0, +\infty)^{|\mathcal{C}|}} e^{-\mathbf{u}^T \mathcal{M} \mathbf{u}/2} d\mathbf{u}.$$

Remark. Note that this theorem actually encompasses Theorem 17, which corresponds to the case where \mathcal{C} is empty. We chose to present both versions because it is instructive to consider the large powers theorem on its own, particularly to see it rewritten under the conditions developed in Section 5.2. Similarly, the case where all coordinates are critical, as in Section 4.6 of Chapter 4, corresponds to $\mathcal{S} = \emptyset$.

Proof. We write the coefficient extraction as a Cauchy integral and apply again the change of variable $z = \zeta e^{i\theta}$. However, for $j \in \mathcal{C}$, as $\zeta_j = 1$ and $z_j = 1$ is a singularity of the integrand, we cannot let θ_j go through 0. So in that case, fixing $\varepsilon_n = n^{-\beta}$ with $\beta \in (\max(\delta, 1/3), 1/2)$, as in the proof of theorem 17, we define the path of integration for θ_j as $(-\pi, -\varepsilon_n)$, then a half-circle centered at 0, starting at $-\varepsilon_n$, going through $i\varepsilon_n$ and stopping at ε_n , then $(\varepsilon_n, \pi]$. For $j \in \mathcal{S}$, we have $\zeta_j < 1$ so we choose the integration path $\theta_j \in (-\pi, \pi]$. The function $\phi(\mathbf{t})$ is defined as in (5.5) and eq. (5.6) becomes

$$[z^{\boldsymbol{\kappa}_n}] \frac{A(\mathbf{z})B(\mathbf{z})^n}{\prod_{j \in \mathcal{S} \cup \mathcal{C}} (1 - z_j)} = \frac{B(\zeta)^n}{\zeta^{\boldsymbol{\kappa}_n}} \frac{1}{(2\pi)^d} \int \frac{A(\zeta e^{i\theta}) e^{-n\phi(\theta)}}{\prod_{j \in \mathcal{S} \cup \mathcal{C}} (1 - \zeta_j e^{i\theta_j})} \frac{d\theta}{e^{i(\boldsymbol{\kappa}_n - \boldsymbol{\lambda}n)^T \boldsymbol{\theta}}}.$$

Central part. It corresponds to the following part of the domain of integration, denoted by γ_n . For $j \in \mathcal{C}$, we choose θ_j on the half-circle described above. For $j \in \mathcal{S}$, we choose $\theta_j \in (-\varepsilon_n, \varepsilon_n)$. On this domain, we apply a Taylor expansion of the integrand. Note that for $j \in \mathcal{C}$, as $\zeta_j = 1$, we have $1 - \zeta_j e^{i\theta_j} \sim -i\theta_j$, while for $j \in \mathcal{S}$, we have $1 - \zeta_j e^{i\theta_j} \sim 1 - \zeta_j$.

$$\frac{1}{(2\pi)^d} \int_{\gamma_n} \frac{A(\zeta e^{i\theta}) e^{-n\phi(\theta)}}{\prod_{j \in \mathcal{S} \cup \mathcal{C}} (1 - \zeta_j e^{i\theta_j})} \frac{d\theta}{e^{i(\boldsymbol{\kappa}_n - \boldsymbol{\lambda}n)^T \boldsymbol{\theta}}} \sim \frac{1}{(2\pi)^d} \int_{\gamma_n} \frac{A(\zeta) e^{-n\boldsymbol{\theta}^T \mathcal{H} \boldsymbol{\theta}/2 + \mathcal{O}(n\|\boldsymbol{\theta}\|^3)}}{\prod_{j \in \mathcal{C}} (-i\theta_j) \prod_{j \in \mathcal{S}} (1 - \zeta_j)} d\theta. \quad (5.8)$$

Note that our choice of ε_n ensures that $\mathcal{O}(n\|\boldsymbol{\theta}\|^3)$ tends to 0. The change of variable $\mathbf{t} = \sqrt{n}\boldsymbol{\theta}$ is applied. The integral is now equivalent with

$$\frac{A(\zeta)}{\prod_{j \in \mathcal{S}} (1 - \zeta_j)} \frac{i^{|\mathcal{C}|}}{(2\pi)^d n^{|\mathcal{S}|/2}} \int \frac{e^{-\mathbf{t}^T \mathcal{H} \mathbf{t} / 2}}{\prod_{j \in \mathcal{C}} t_j} d\mathbf{t},$$

where the domain of integration is the following. For $j \in \mathcal{C}$, t_j is on the line from $-\sqrt{n}\varepsilon_n$ to i , then on the line from i to $\sqrt{n}\varepsilon_n$. For $j \in \mathcal{S}$, $t_j \in (\sqrt{n} - \varepsilon_n, \sqrt{n}\varepsilon_n)$. The tails of this integral are exponentially small, so we add them to simplify the expression. The central asymptotics becomes

$$\frac{A(\zeta)}{\prod_{j \in \mathcal{S}} (1 - \zeta_j)} \frac{i^{|\mathcal{C}|}}{(2\pi)^d n^{|\mathcal{S}|/2}} \int_{(i-\infty, i+\infty)^{|\mathcal{C}|} \times (-\infty, +\infty)^{|\mathcal{S}|}} \frac{e^{-\mathbf{t}^T \mathcal{H} \mathbf{t} / 2}}{\prod_{j \in \mathcal{C}} t_j} d\mathbf{t}.$$

To further simplify the expression, we inject the following identity, valid for any t with positive imaginary part:

$$\frac{1}{t} = -i \int_0^{+\infty} e^{itu} du.$$

Applying this identity for $t = t_j$ and $j \in \mathcal{C}$ yields

$$\frac{A(\zeta)}{(2\pi)^d n^{|\mathcal{S}|/2} \prod_{j \in \mathcal{S}} (1 - \zeta_j)} \int_{(i-\infty, i+\infty)^{|\mathcal{C}|} \times (-\infty, +\infty)^{|\mathcal{S}|}} \int_{(0, +\infty)^{|\mathcal{C}|}} \exp\left(-\frac{1}{2} \mathbf{t}^T \mathcal{H} \mathbf{t} + i \sum_{j \in \mathcal{C}} u_j t_j\right) du d\mathbf{t}.$$

Let $(\mathbf{u}_{\mathcal{C}}, \mathbf{0}_{\mathcal{S}})$ denote the vector of dimension d with u_j at any index $j \in \mathcal{C}$ and 0 at any index $j \in \mathcal{S}$. The matrix \mathcal{H} is definite positive, so its inverse and square root are well defined. We interchange the integrals and apply the change of variable $\mathbf{s} = \sqrt{\mathcal{H}} \mathbf{t}$

$$\frac{A(\zeta)}{\prod_{j \in \mathcal{S}} (1 - \zeta_j)} \frac{1}{(2\pi)^d \sqrt{n^{|\mathcal{S}|} \det(\mathcal{H})}} \times \int_{(0, +\infty)^{|\mathcal{C}|}} \int_{(i-\infty, i+\infty)^{|\mathcal{C}|} \times (-\infty, +\infty)^{|\mathcal{S}|}} \exp\left(-\frac{1}{2} \mathbf{s}^T \mathbf{s} + i(\mathbf{u}_{\mathcal{C}}, \mathbf{0}_{\mathcal{S}})^T \mathcal{H}^{-1/2} \mathbf{s}\right) ds du.$$

The argument of the exponential is factorized

$$\frac{A(\zeta)}{\prod_{j \in \mathcal{S}} (1 - \zeta_j)} \frac{1}{(2\pi)^d \sqrt{n^{|\mathcal{S}|} \det(\mathcal{H})}} \times \iint \exp\left(-\frac{1}{2} \left(\mathbf{s} - i\mathcal{H}^{-1/2}(\mathbf{u}_{\mathcal{C}}, \mathbf{0}_{\mathcal{S}})\right)^T \left(\mathbf{s} - i\mathcal{H}^{-1/2}(\mathbf{u}_{\mathcal{C}}, \mathbf{0}_{\mathcal{S}})\right) - \frac{1}{2} (\mathbf{u}_{\mathcal{C}}, \mathbf{0}_{\mathcal{S}})^T \mathcal{H}^{-1} (\mathbf{u}_{\mathcal{C}}, \mathbf{0}_{\mathcal{S}})\right) ds du.$$

The integrals with respect to each s_j are now separated, each a Gaussian integral with value $\sqrt{2\pi}$. We deduce that the central part has asymptotics

$$\frac{A(\zeta)}{\prod_{j \in \mathcal{S}} (1 - \zeta_j)} \frac{1}{\sqrt{(2\pi)^{d_n} |\mathcal{S}| \det(\mathcal{H})}} \int_{(0, +\infty)^{|\mathcal{C}|}} \exp\left(-\frac{1}{2}(\mathbf{u}_{\mathcal{C}}, \mathbf{0}_{\mathcal{S}})^T \mathcal{H}^{-1}(\mathbf{u}_{\mathcal{C}}, \mathbf{0}_{\mathcal{S}})\right) d\mathbf{u}.$$

Tail. The proof that the tail is exponentially small compared to the central part is the same as in the proof of Theorem 17. \square

5.5 Asymptotic Expansion

In the previous chapter, we observed that applying Theorem 18 to the critical case only yields the limiting value, not the rate of convergence. This is expected, since in this case the limit is nonzero, and the first-order approximation is therefore equal to this limit. In situations of this kind, it is of interest to determine the next-order term. More generally, one may seek asymptotic expansions of arbitrary order. So, in this section, we explain how to obtain more precise asymptotic results, as applied in Section 4.9 of Chapter 4.

As mentioned before, our treatment of the subcritical case, in Theorem 17, is an application of a technique known as *large powers*, relying on the *Laplace method*. It is well known in the field (see e.g. [PW13]) that the corresponding asymptotic expansion consists of terms in increasing powers of $1/n$. The critical and mixed cases from Theorem 18 combine properties from large powers and *singularity analysis*. In univariate singularity analysis [FS09, Section VI], the asymptotic expansion is also found to be in powers of $1/n$. Somewhat surprisingly, our next result shows that in the critical and mixed cases, the asymptotic expansion is in powers of $1/\sqrt{n}$ instead.

Theorem 19. *Consider a power series $B(z_1, \dots, z_d)$ with nonnegative coefficients, full rank, aperiodic and convergent on a neighborhood Ω of 0. Let $K(\mathbf{t}) = \log(B(e^{\mathbf{t}}))$ denote its cumulant generating function. Let λ denote a vector of positive real values such that the function $K(\mathbf{t}) - \lambda^T \mathbf{t}$ does not tend to its infimum on the boundary of $\Psi = \{\mathbf{t} \in \mathbb{R}^d, e^{\mathbf{t}} \in \Omega\}$. Let τ denote the unique point of Ψ where $K(\mathbf{t}) - \lambda^T \mathbf{t}$ reaches its minimum. Set $\zeta = e^{\tau}$.*

We assume $\zeta_j \leq 1$ for all j and denote by \mathcal{S} the set of indices such that $\zeta_j < 1$, and by \mathcal{C} the corresponding set for $\zeta_j = 1$. Note that by assumption, we have $\mathcal{S} \cup \mathcal{C} = \{1, \dots, d\}$. Let \mathcal{H} denote the Hessian matrix of $K(\mathbf{t})$ at τ and \mathcal{M} the submatrix of \mathcal{H}^{-1} corresponding to the rows

and column in \mathcal{C} . Consider a power series $A(z)$ analytic on the torus of radius ζ and satisfying $A(\zeta) \neq 0$. Consider a real value $\delta < 1/2$ and a sequence of vectors of real values

$$\kappa_n = \lambda n + \mathcal{O}(n^\delta).$$

Then for any $r \geq 1$, there exist functions $a_{k,n}$ of $\kappa_n - \lambda n$ (which are bounded with respect to n) such that

$$[z^{\kappa_n}] \frac{A(z)B(z)^n}{\prod_{j \in \text{SUC}} (1 - z_j)} = \frac{B(\zeta)^n}{\zeta^{\kappa_n}} \frac{1}{n^{|\mathcal{S}|/2}} \left(\sum_{k=0}^{r-1} a_{k,n} n^{-k/2} + \mathcal{O}(n^{-r/2}) \right). \quad (5.9)$$

An algorithm computing them is provided in the proof. In particular, $a_0 := a_{0,n}$ is independent of n and is equal to

$$a_0 = \frac{1}{\sqrt{(2\pi)^d \det(\mathcal{H})}} \int_{(0,+\infty)^{|\mathcal{C}|}} e^{-\mathbf{u}^T \mathcal{M} \mathbf{u}} d\mathbf{u}$$

and we will express $a_{1,n}$ in Lemma 16. Furthermore, in the particular case $\mathcal{C} = \emptyset$, then $a_{k,n} = 0$ for all odd k .

Proof. We fix some $r \geq 1$. Let us say that a sequence u_n tends superpolynomially fast to 0 if for any integer k , the sequence $n^k u_n$ tends to 0. For example, $e^{-\sqrt{n}}$ tends superpolynomially fast to 0. We proved that the tails tends superpolynomially fast to 0, so its contribution to Equation (5.9) is negligible. We apply the change of variable $\mathbf{t} = \sqrt{n}\boldsymbol{\theta}$ in the left-hand side of Equation (5.8) and deduce

$$\begin{aligned} [z^{\kappa_n}] \frac{A(z)B(z)^n}{\prod_{j \in \text{SUC}} (1 - z_j)} &= \frac{B(\zeta)^n}{\zeta^{\kappa_n}} \frac{1}{(2\pi)^d n^{d/2}} \\ &\times \int \frac{A(\zeta e^{it/\sqrt{n}}) e^{-n\phi(t/\sqrt{n})}}{\prod_{j \in \text{SUC}} (1 - \zeta_j e^{it_j/\sqrt{n}})} \frac{d\mathbf{t}}{e^{i(\kappa_n - \lambda n)^T \mathbf{t}/\sqrt{n}}} \left(1 + \mathcal{O}(n^{-r/2}) \right) \end{aligned}$$

where the domain of integration is the following. For $j \in \mathcal{C}$, t_j is on the line from $-\sqrt{n}\varepsilon_n$ to i , then on the line from i to $\sqrt{n}\varepsilon_n$. For $j \in \mathcal{S}$, $t_j \in (-\sqrt{n}\varepsilon_n, \sqrt{n}\varepsilon_n)$. Let us set

$$F(\mathbf{t}, y) := \frac{A(\zeta e^{i\mathbf{t}y}) \exp\left(\frac{1}{2} \mathbf{t}^T \mathcal{H} \mathbf{t} - \phi(\mathbf{t}y)y^{-2}\right)}{\left(\prod_{j \in \mathcal{C}} \frac{1 - e^{it_j y}}{-it_j y}\right) \prod_{j \in \mathcal{S}} (1 - \zeta_j e^{it_j y})} \frac{1}{e^{i(\kappa_n - \lambda n)^T \mathbf{t}y}}. \quad (5.10)$$

Recalling that $\zeta_j = 1$ for $j \in \mathcal{C}$ and injecting $F(\mathbf{t}, y)$ in our previous expression leads to

$$[z^{\kappa_n}] \frac{A(z)B(z)^n}{\prod_{j \in \text{SUC}} (1 - z_j)} = \frac{B(\zeta)^n}{\zeta^{\kappa_n}} \frac{i^{|\mathcal{C}|}}{(2\pi)^d n^{|\mathcal{S}|/2}} \int \frac{F(\mathbf{t}, n^{-1/2})}{\prod_{j \in \mathcal{C}} t_j} e^{-\mathbf{t}^T \mathcal{H} \mathbf{t}/2} d\mathbf{t} \left(1 + \mathcal{O}(n^{-r/2}) \right). \quad (5.11)$$

The term $\frac{1-e^{it_j y}}{-it_j y}$ has a Taylor expansion at $y = 0$. Observe also that the valuation of $\frac{1}{2}t^T \mathcal{H} t - \phi(\mathbf{t}y)y^{-2}$ in y is at least 1, and that in the series expansion at $(\mathbf{t}, y) = (\mathbf{0}, 0)$, each monomial where the sum of the exponents of the t_j is p has y at the power $p - 2$. This implies that $F(\mathbf{t}, y)$ has a Taylor expansion at $y = 0$, denoted by

$$F(\mathbf{t}, y) = \sum_{k=0}^{r-1} F_k(\mathbf{t})y^k + \mathcal{O}\left((t_1 \cdots t_d)^{\beta_r} y^r\right)$$

where β_r is some sequence depending on r (we could prove $\beta_r \leq 3r$ but will not need it) and each $F_k(\mathbf{t})$ is a polynomial in \mathbf{t} and in $\kappa_n - \lambda n$, which is a bounded sequence in n . With our domain of integration, we have for any β_r

$$\int \frac{\mathcal{O}\left((t_1 \cdots t_d)^{\beta_r} n^{-r/2}\right)}{\prod_{j \in \mathcal{C}} t_j} e^{-t^T \mathcal{H} t/2} d\mathbf{t} = \mathcal{O}(n^{-r/2})$$

so, injecting in Equation (5.11) the Taylor expansion of $F(\mathbf{t}, y)$ at $y = n^{-1/2}$, we obtain

$$[z^{\kappa_n}] \frac{A(z)B(z)^n}{\prod_{j \in \mathcal{S} \cup \mathcal{C}} (1 - z_j)} = \frac{B(\zeta)^n}{\zeta^{\kappa_n}} \frac{1}{n^{|\mathcal{S}|/2}} \left(\sum_{k=0}^{r-1} \left[\frac{i^{|\mathcal{C}|}}{(2\pi)^d} \int \frac{F_k(\mathbf{t})}{\prod_{j \in \mathcal{C}} t_j} e^{-t^T \mathcal{H} t/2} d\mathbf{t} \right] n^{-k/2} + \mathcal{O}(n^{-r/2}) \right).$$

The tails of the integrals are exponentially small, so we add them to simplify the expression. After deformation, the domain of integration becomes $(i - \infty, i + \infty)^{|\mathcal{C}|} \times (-\infty, +\infty)^{|\mathcal{S}|}$. Defining

$$a_{k,n} := \frac{i^{|\mathcal{C}|}}{(2\pi)^d} \int_{(i-\infty, i+\infty)^{|\mathcal{C}|} \times (-\infty, +\infty)^{|\mathcal{S}|}} \frac{F_k(\mathbf{t})}{\prod_{j \in \mathcal{C}} t_j} e^{-t^T \mathcal{H} t/2} d\mathbf{t},$$

we obtain Equation (5.9).

The expression of a_0 was obtained and simplified in the proof of Theorem 18. Let us now show how to apply the same ideas to the other coefficients. Write for any complex t with positive imaginary part

$$\frac{i}{t} = \int_0^{+\infty} e^{itu} du,$$

then

$$a_{k,n} = \frac{1}{(2\pi)^d} \iint_{(0, +\infty)^{|\mathcal{C}|}} \mathbf{t}^p F_k(\mathbf{t}) e^{-t^T \mathcal{H} t/2 + i(\mathbf{u}_\mathcal{C}, \mathbf{0}_\mathcal{S})^T \mathbf{t}} d\mathbf{u} d\mathbf{t}.$$

We apply the change of variable $\mathbf{s} = \sqrt{\mathcal{H}}\mathbf{t}$ and factorize the argument of the exponential as in the proof of Theorem 18. The coefficient $a_{k,n}$ is then equal to

$$\frac{1}{(2\pi)^d \sqrt{\det(\mathcal{H})}} \iint_{(0,+\infty)^{|\mathcal{C}|}} F_k(\mathcal{H}^{-1/2}\mathbf{s}) e^{-\mathbf{s}^T \mathbf{s}/2 + i(\mathbf{u}_{\mathcal{C}}, \mathbf{0}_{\mathcal{S}})^T \mathcal{H}^{-1/2} \mathbf{s}} d\mathbf{u} d\mathbf{s}$$

Which rewrites as

$$\frac{1}{(2\pi)^d \sqrt{\det(\mathcal{H})}} \iint_{(0,+\infty)^{|\mathcal{C}|}} F_k(\mathcal{H}^{-1/2}\mathbf{s}) e^{-(\mathbf{s} - i\mathcal{H}^{-1/2}(\mathbf{u}_{\mathcal{C}}, \mathbf{0}_{\mathcal{S}}))^T (\mathbf{s} - i\mathcal{H}^{-1/2}(\mathbf{u}_{\mathcal{C}}, \mathbf{0}_{\mathcal{S}}))/2 - \mathbf{u}^T \mathcal{M} \mathbf{u}/2} d\mathbf{u} d\mathbf{s}.$$

We interchange the integrals and shift the vector \mathbf{s} by $i\mathcal{H}^{-1/2}(\mathbf{x}, \mathbf{0})$

$$a_{k,n} = \frac{1}{(2\pi)^d \sqrt{\det(\mathcal{H})}} \int_{(0,+\infty)^{|\mathcal{C}|}} \int F_k(\mathcal{H}^{-1/2}\mathbf{s} + i\mathcal{H}^{-1/2}(\mathbf{u}_{\mathcal{C}}, \mathbf{0}_{\mathcal{S}})) e^{-\mathbf{s}^T \mathbf{s}/2 - \mathbf{u}^T \mathcal{M} \mathbf{u}/2} d\mathbf{s} d\mathbf{u}.$$

Recall that $F_k(\mathbf{t})$ is a polynomial, so there exist polynomials $G_{k,p}(\mathbf{u})$ (which are also polynomials in $\kappa_n - \lambda_n$) and a finite set $S_k \subset \mathbb{Z}_{\geq 0}^d$ such that

$$F_k(\mathcal{H}^{-1/2}\mathbf{s} + i\mathcal{H}^{-1/2}(\mathbf{u}_{\mathcal{C}}, \mathbf{0}_{\mathcal{S}})) = \sum_{\mathbf{p} \in S_k} \mathbf{s}^{\mathbf{p}} G_{k,p}(\mathbf{u}).$$

We deduce

$$a_{k,n} = \frac{1}{(2\pi)^d \sqrt{\det(\mathcal{H})}} \sum_{\mathbf{p} \in S_k} \int_{(0,+\infty)^{|\mathcal{C}|}} \int \mathbf{s}^{\mathbf{p}} G_{k,p}(\mathbf{u}) e^{-\mathbf{s}^T \mathbf{s}/2 - \mathbf{u}^T \mathcal{M} \mathbf{u}/2} d\mathbf{s} d\mathbf{u}.$$

Each integral with respect to s_j , for $j \in \mathcal{S} \cup \mathcal{C}$, is simplified using the formula

$$\int_{(i-\infty, i+\infty)} s^p e^{-s^2/2} ds = \begin{cases} 0 & \text{if } p \text{ is odd,} \\ \sqrt{2\pi} \frac{(2q)!}{2^q q!} & \text{if } p = 2q. \end{cases}$$

We observe that in the expression of $a_{k,n}$, only vectors \mathbf{p} where all coefficients are even may contribute. Let us define

$$S_k^* = \{\mathbf{q}, 2\mathbf{q} \in S_k\},$$

then

$$a_{k,n} = \frac{1}{\sqrt{(2\pi)^d \det(\mathcal{H})}} \sum_{\mathbf{q} \in S_k^*} \left(\prod_{j \in \mathcal{S} \cup \mathcal{C}} \frac{(2q_j)!}{2^{q_j} q_j!} \right) \int_{(0,+\infty)^{|\mathcal{C}|}} G_{k,2\mathbf{q}}(\mathbf{u}) e^{-\mathbf{u}^T \mathcal{M} \mathbf{u}/2} d\mathbf{u}. \quad (5.12)$$

In the particular case $\mathcal{C} = \emptyset$, the function F_k is simply composed with $\mathcal{H}^{-1/2}\mathbf{s}$. If k is odd, by construction, the sum of the exponents in each monomial of $F_k(\mathbf{t})$ is odd. This implies that each monomial of $F_k(\mathcal{H}^{-1/2}\mathbf{s})$ contains some s_j at an odd power, so $a_{k,n} = 0$.

An alternative approach to compute the coefficients $a_{k,n}$ is to apply the change of variable $-\mathbf{t}^T \mathbf{t} / 2 = \phi(\boldsymbol{\theta})$ in the left-hand side of Equation (5.8). We refer the interested reader to [PW13, Chapter 5]. \square

Theorem 19 essentially guarantees the existence of the coefficients, but does not present them in a particularly transparent form. In particular, the proof establishes the existence of polynomials $G_{k,p}(\mathbf{u})$ (depending on $\boldsymbol{\kappa}_n - \boldsymbol{\lambda}_n$) and of a finite set $S_k \subset \mathbb{Z}_{\geq 0}^d$ such that

$$a_{k,n} = \frac{1}{\sqrt{(2\pi)^d \det(\mathcal{H})}} \sum_{\mathbf{q} \in S_k^*} \left(\prod_{j \in \mathcal{S} \cup \mathcal{C}} \frac{(2q_j)!}{2^{q_j} q_j!} \right) \int_{(0,+\infty)^{|\mathcal{C}|}} G_{k,2\mathbf{q}}(\mathbf{u}) e^{-\mathbf{u}^T \mathcal{M} \mathbf{u} / 2} d\mathbf{u},$$

without making the individual terms explicit. Our next result provides a more concrete expression for the coefficient $a_{1,n}$ in the asymptotic expansion from Theorem 19.

Lemma 16. *With ζ defined as in Theorem 19, define the polynomial*

$$\phi_3(\mathbf{t}) := [y^3] \log \left(\frac{B(\zeta e^{t\mathbf{y}})}{B(\zeta)} \right).$$

Observe that the sum of the exponents of each of its monomials is 3. Let $\phi_3^*(\mathbf{u})$ denote the polynomial obtained by replacing each monomial $t_j t_k t_\ell$ (with j, k, ℓ in $\mathcal{S} \cup \mathcal{C}$ not necessarily distinct) with

$$\begin{aligned} & -e_j^T \mathcal{H}^{-1}(\mathbf{u}_{\mathcal{C}}, \mathbf{0}_{\mathcal{S}}) e_k^T \mathcal{H}^{-1}(\mathbf{u}_{\mathcal{C}}, \mathbf{0}_{\mathcal{S}}) e_\ell^T \mathcal{H}^{-1}(\mathbf{u}_{\mathcal{C}}, \mathbf{0}_{\mathcal{S}}) \\ & + \left((\mathcal{H}^{-1})_{k,\ell} \mathbf{e}_j + (\mathcal{H}^{-1})_{j,\ell} \mathbf{e}_k + (\mathcal{H}^{-1})_{j,k} \mathbf{e}_\ell \right)^T \mathcal{H}^{-1}(\mathbf{u}_{\mathcal{C}}, \mathbf{0}_{\mathcal{S}}). \end{aligned} \quad (5.13)$$

Let \mathbf{b} denote the vector of size d with $b_j = -1/2$ for $j \in \mathcal{C}$ and $b_j = \frac{\zeta_j}{1-\zeta_j}$ for $j \in \mathcal{S}$. Then the term $a_{1,n}$ in the asymptotic expansion from Theorem 19 is equal to

$$\begin{aligned} a_{1,n} &= \frac{A(\zeta)}{\prod_{j \in \mathcal{S}} (1 - \zeta_j)} \frac{1}{\sqrt{(2\pi)^d \det(\mathcal{H})}} \\ & \times \int_{(0,+\infty)^{|\mathcal{C}|}} \left(\phi_3^*(\mathbf{u}) - \left[\frac{\nabla A(\zeta)}{A(\zeta)} + \mathbf{b} - (\boldsymbol{\kappa}_n - \boldsymbol{\lambda}_n) \right]^T \mathcal{H}^{-1}(\mathbf{u}_{\mathcal{C}}, \mathbf{0}_{\mathcal{S}}) \right) e^{-\mathbf{u} \mathcal{M} \mathbf{u} / 2} d\mathbf{u}. \end{aligned}$$

Proof. Our first step is to compute $F_1(\mathbf{t}) = [y^1]F(\mathbf{t}, y)$, where $F(\mathbf{t}, y)$ is defined in Equation (5.10). Using the expression of $\phi_3(\mathbf{t})$ from the lemma, we have the following Taylor expansions

$$\begin{aligned} A(\zeta e^{i\mathbf{t}y}) &= A(\zeta) + i\nabla_A(\zeta)\mathbf{t}y + \mathcal{O}(y^2), \\ \exp\left(\frac{1}{2}\mathbf{t}^T\mathcal{H}\mathbf{t} - \phi(\mathbf{t}y)y^{-2}\right) &= 1 - i\phi_3(\mathbf{t})y + \mathcal{O}(y^2), \\ \frac{1}{\frac{1-e^{it_jy}}{-it_jy}} &= 1 - \frac{i}{2}t_jy + \mathcal{O}(y^2), \\ \frac{1}{1-\zeta_j e^{it_jy}} &= \frac{1}{1-\zeta_j} + \frac{\zeta_j}{(1-\zeta_j)^2}it_jy + \mathcal{O}(y^2), \\ \frac{1}{e^{i(\boldsymbol{\kappa}_n - \boldsymbol{\lambda}_n)^T\mathbf{t}y}} &= 1 - i(\boldsymbol{\kappa}_n - \boldsymbol{\lambda}_n)^T\mathbf{t}y + \mathcal{O}(y^2). \end{aligned}$$

Using the vector \mathbf{b} from the current lemma, we obtain the following Taylor expansion for the products

$$\frac{1}{\prod_{j \in \mathcal{C}} \frac{1-e^{it_jy}}{-it_jy}} \frac{1}{\prod_{j \in \mathcal{S}} (1-\zeta_j e^{it_jy})} = 1 + i\mathbf{b}^T\mathbf{t}y + \mathcal{O}(y^2).$$

We deduce

$$F_1(\mathbf{t}) = \frac{-iA(\zeta)}{\prod_{j \in \mathcal{S}} (1-\zeta_j)} \left(\phi_3(\mathbf{t}) - \left[\frac{\nabla A(\zeta)}{A(\zeta)} + \mathbf{b} - (\boldsymbol{\kappa}_n - \boldsymbol{\lambda}_n) \right]^T \mathbf{t} \right).$$

Following the proof of Theorem 19, our next step is to compute

$$\sum_{\mathbf{q} \in \mathcal{S}_1^*} \left(\prod_{j \in \mathcal{S} \cup \mathcal{C}} \frac{(2q_j)!}{2^{q_j} q_j!} \right) G_{1,2\mathbf{q}}(\mathbf{u}).$$

It is obtained by the following transformation of $F_1(\mathbf{t})$. Replace \mathbf{t} with $\mathcal{H}^{-1/2}\mathbf{s} + i\mathcal{H}^{-1}(\mathbf{u}_{\mathcal{C}}, \mathbf{0}_{\mathcal{S}})$, then for each j and k , replace s_j^k with 0 if k is odd, and with $\frac{(2q)!}{2^q q!}$ if $k = 2q$. Clearly, any t_j at the power 1 will be transformed into $i\mathcal{H}^{-1}(\mathbf{u}_{\mathcal{C}}, \mathbf{0}_{\mathcal{S}})$. By construction, $\phi_3(\mathbf{t})$ is a polynomial where the sum of exponents in each monomial is 3. Consider three integers j, k, ℓ in $\mathcal{S} \cup \mathcal{C}$. The monomial $t_j t_k t_\ell$ will be transformed into

$$\begin{aligned} &\left(e_j^T \mathcal{H}^{-1/2} \mathbf{s} + i e_j^T \mathcal{H}^{-1}(\mathbf{u}_{\mathcal{C}}, \mathbf{0}_{\mathcal{S}}) \right) \left(e_k^T \mathcal{H}^{-1/2} \mathbf{s} + i e_k^T \mathcal{H}^{-1}(\mathbf{u}_{\mathcal{C}}, \mathbf{0}_{\mathcal{S}}) \right) \\ &\quad \times \left(e_\ell^T \mathcal{H}^{-1/2} \mathbf{s} + i e_\ell^T \mathcal{H}^{-1}(\mathbf{u}_{\mathcal{C}}, \mathbf{0}_{\mathcal{S}}) \right) \end{aligned}$$

where each s_h at an odd power is replaced by 0 and each s_h^2 is replaced by 1, for all $h \in \mathcal{S} \cup \mathcal{C}$. The expression becomes

$$\begin{aligned}
& -ie_j^T \mathcal{H}^{-1}(\mathbf{u}_C, \mathbf{0}_S) e_k^T \mathcal{H}^{-1}(\mathbf{u}_C, \mathbf{0}_S) e_\ell^T \mathcal{H}^{-1}(\mathbf{u}_C, \mathbf{0}_S) \\
& + ie_j^T \mathcal{H}^{-1}(\mathbf{u}_C, \mathbf{0}_S) \sum_{h=1}^d e_k^T \mathcal{H}^{-1/2} e_h e_\ell^T \mathcal{H}^{-1/2} e_h \\
& + ie_k^T \mathcal{H}^{-1}(\mathbf{u}_C, \mathbf{0}_S) \sum_{h=1}^d e_j^T \mathcal{H}^{-1/2} e_h e_\ell^T \mathcal{H}^{-1/2} e_h \\
& + ie_\ell^T \mathcal{H}^{-1}(\mathbf{u}_C, \mathbf{0}_S) \sum_{h=1}^d e_j^T \mathcal{H}^{-1/2} e_h e_k^T \mathcal{H}^{-1/2} e_h
\end{aligned}$$

which, given the symmetry of $\mathcal{H}^{-1/2}$, is equal to (5.13). With $\phi_3^*(\mathbf{u})$ defined as in the lemma, we deduce that after evaluating $F_1(\mathbf{t})$ at $\mathbf{t} = \mathcal{H}^{-1/2} \mathbf{s} + i\mathcal{H}^{-1}(\mathbf{u}_C, \mathbf{0}_S)$, then replacing for each j any odd power of s_j with 0 and s_j^2 with 1, we obtain

$$\begin{aligned}
\sum_{\mathbf{q} \in S_1^*} \left(\prod_{j \in S \cup C} \frac{(2q_j)!}{2^{q_j} q_j!} \right) G_{1,2\mathbf{q}}(\mathbf{u}) = \\
\frac{A(\zeta)}{\prod_{j \in S} (1 - \zeta_j)} \left(\phi_3^*(\mathbf{u}) - \left[\frac{\nabla A(\zeta)}{A(\zeta)} + \mathbf{b} - (\boldsymbol{\kappa}_n - \boldsymbol{\lambda}_n) \right]^T \mathcal{H}^{-1}(\mathbf{u}_C, \mathbf{0}_S) \right).
\end{aligned}$$

Injecting this in Equation (5.12) for $k = 1$ yields the expression of $a_{1,n}$ from the current lemma. \square

5.6 Conclusion

Summary of the Contributions. In this chapter, we developed multivariate saddle point theorems. To this end, we first established sufficient conditions on the generating function to guarantee the existence and uniqueness of the saddle point, thereby extending classical univariate results to the multivariate setting. Then, building on this foundation, we formulated a multivariate large powers theorem and formulated its adaption to cases where the saddle point coincides with a singularity of the generating function. Finally, we derived asymptotic expansions of arbitrary order, enabling finer precision in our results.

Future Work. In this chapter, we focused on the specific setting relevant to the previous chapter, namely coefficient extraction of the form

$$[z^{\boldsymbol{\lambda}_n}] A(z) B(z)^n.$$

A natural extension would be to explore more general frameworks, for instance when the polynomial $B(z)$ is not simply raised to the power n but appears through a more intricate functional form, as in Chapter 2, where the classical large powers setting did not apply directly.

Another direction is to compute more terms in the asymptotic expansions, either manually or with computer assistance. This could eventually lead to symbolic algorithms that perform these computations automatically.

Anytime Sorting Algorithms

6.1 Introduction

Now that we have examined the probability of encountering the Condorcet paradox, we shift our attention towards the construction of the rankings themselves. Indeed, throughout the thesis, we supposed that voters were able to provide complete rankings over their preferences, but in practice, is it that simple for voters to elicit their own preferences?

In Australia, elections to the House of Representatives are conducted using the *Instant-Runoff Voting (IRV)* system, which is based on preference-ranking ballots.

The Instant-Runoff Voting (IRV) system, also known as *Ranked-Choice Voting (RCV)*, *Single Transferable Voting (STV)*, *Preferential Voting*, or the *Alternative Vote*, proceeds as follows:

1. Each voter submits a ranking of all candidates from most to least preferred.
2. The first-place votes are counted.
3. If one candidate obtains an absolute majority (more than half of the votes), that candidate is declared the winner.
4. Otherwise, the candidate with the fewest first-place votes is eliminated. Ballots ranking this candidate first are then transferred to the next preferred candidate listed on each ballot.
5. Steps 2 to 4 are repeated until one candidate achieves an absolute majority and is elected.

When ties occur, two main tie-breaking approaches can be distinguished. The first, *immediate tie-breaking*, resolves ties as they arise using a deterministic or random rule (for instance, a predefined candidate order or a random draw). The second, the *parallel universes* approach, simultaneously considers all possible sequences of tie-breaking decisions. Each possible choice defines a separate "parallel universe" in which IRV is executed, and the final set of winners corresponds to all candidates who win in at least one of these universes. Note that this method does not necessarily yield a single winner.

Instant-Runoff Voting is widely implemented in Australia, not only in political elections but also in various non-governmental decision-making contexts, such as job recruitment processes or committee selections. Beyond Australia, IRV is employed in several national elections across the world. Notably, it is used to elect the heads of state in India, Ireland, and Sri Lanka. A variant in which ballots are truncated to the top three candidates is also used to elect members of the National Parliament of Papua New Guinea.

Figure 6.1 illustrates the execution of the Instant-Runoff Voting process on a profile with $m = 3$ candidates and $n = 11$ voters. The profile of votes is shown in the first table. Each column corresponds to a ranking of the candidates (displayed in black), while the number of voters who submitted that ranking is indicated above in blue. In this profile, candidate 1 receives 3 first-place votes, while candidates 2 and 3 each receive 4 first-place votes. Therefore, candidate 1 has the fewest first-place votes and is eliminated according to the IRV procedure. The ballots are then updated by removing candidate 1 from all rankings. This updated set of ballots is displayed in the second table, and the remaining votes are transferred to each voter's next preferred candidate. The third table shows these transferred votes to illustrate the resulting profile after the elimination of candidate 1. By examining this resulting profile, we see that candidate 3 now holds the majority of first-place votes among the remaining candidates, so they win the election in the second round.

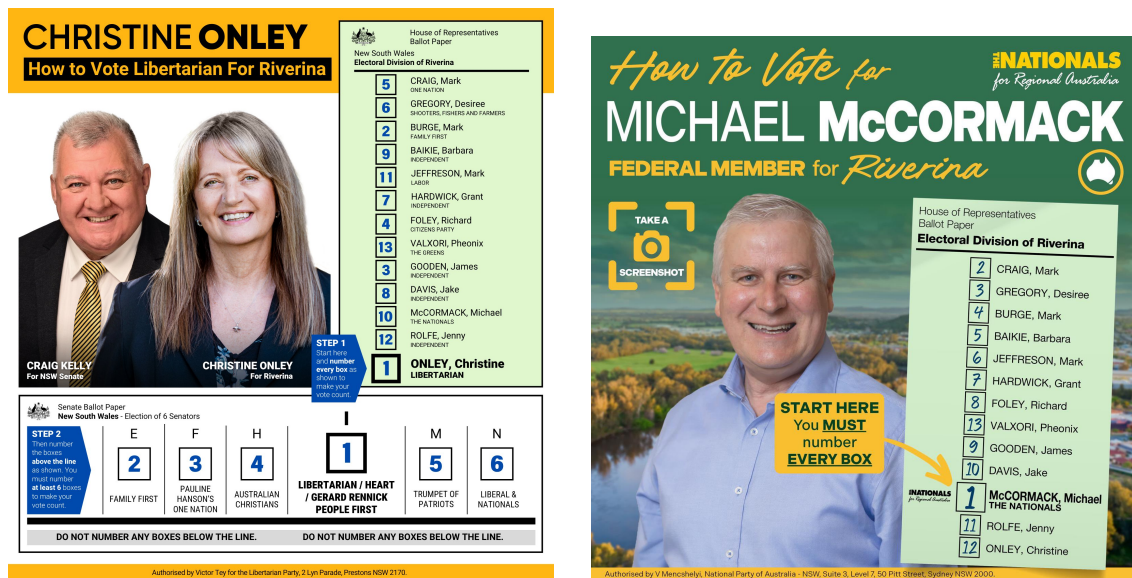
1	2	4	0	2	2
1	1	2	2	3	3
2	3	1	3	1	2
3	2	3	1	2	1

1	2	4	0	2	2
2	3	2	2	3	3
3	2	3	3	2	2

5	6
2	3
3	2

Fig. 6.1: Execution of the IRV process on a profile with $m = 3$ candidates and $n = 11$ voters. The first table shows the initial profile of votes. Each column represents a ranking (in black), with the number of voters choosing that ranking indicated above in blue. Candidate 1 receives the fewest first-place votes and is therefore eliminated. The second table displays the ballots after candidate 1 is removed, and the third table shows the aggregated profile. In the second round, candidate 3 secures the majority and thus wins the election.

In the federal elections of Australia, a ballot is considered valid only if all candidates are ranked. Consequently, in each of the 151 electoral divisions, voters are asked to rank all candidates in their division by order of preference. The number of options per division can vary significantly. Some divisions present relatively few choices, for instance, during the 2025 federal election, the divisions of Bean and Fenner each had only four candidates. However, others offer a much larger selection. Notably, the divisions of Riverina and Calwell each featured as many as 13 candidates. However, others offer a much larger selection. Notably, the divisions of Riverina and Calwell each featured as many as 13 candidates, as consultable on the webpage [Bui].



(a) “How to vote” card shared by candidate Christine Onley on X. (b) “How to vote” card shared by candidate Michael McCormack on X.

Fig. 6.2: “How to vote” cards distributed by two candidates running for the 2025 Australian federal election in the Division of Riverina.

Since incomplete ballots are considered invalid, it is essential for voters to rank all candidates. However, accurately ordering up to 13 candidates can be a challenging task. To assist with this, political parties distribute *how-to-vote cards* at the entrance of polling stations on election day. Figure 6.2 shows two examples of such cards, shared by candidates running in the Riverina district for the 2025 Australian federal election. Both were posted on the social media platform X (formerly Twitter). The first one was shared by Christine Onley, candidate for the Libertarian Party, in two posts shown in Figure 6.3. The second one was shared by Michael McCormack, candidate for the National Party, also in two posts, shown in Figure 6.4. These cards raise several important questions.

First, do all parties have equal capacity to produce and distribute such material, or are smaller parties disproportionately affected?

Second, is the order on these cards intended to be followed literally? On Christine Onley’s card, the instruction reads: “Start here and number every box **as shown** to make your vote count.” However, in her April 21 post (Figure 6.3a), she explicitly encourages voters to “chose where [their] preferences go”. This mention disappears in her May 2 post (Figure 6.3b). It is therefore unclear whether the order on the card reflects any real preference or was chosen arbitrarily.

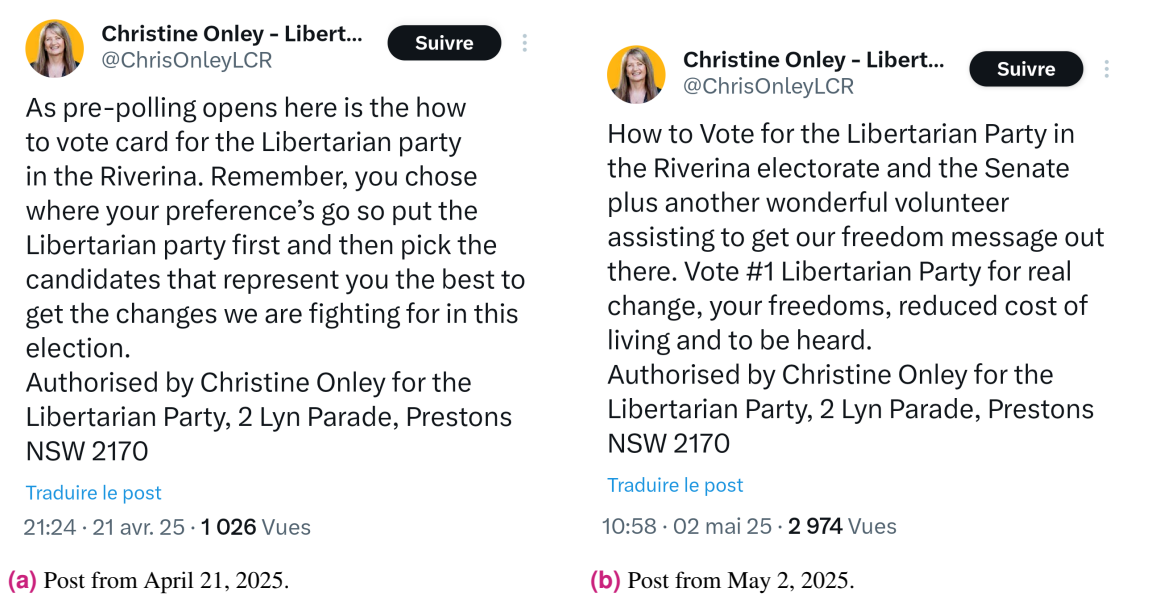
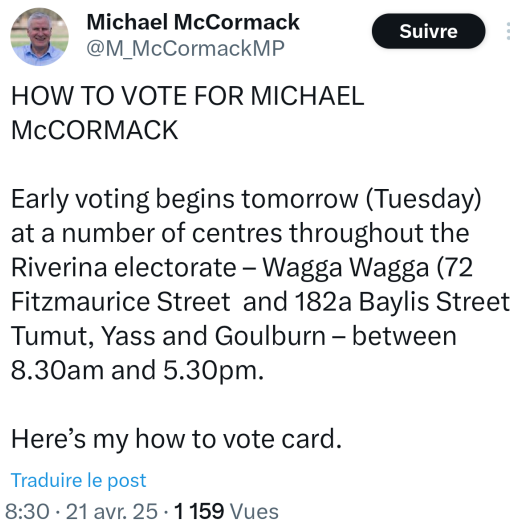
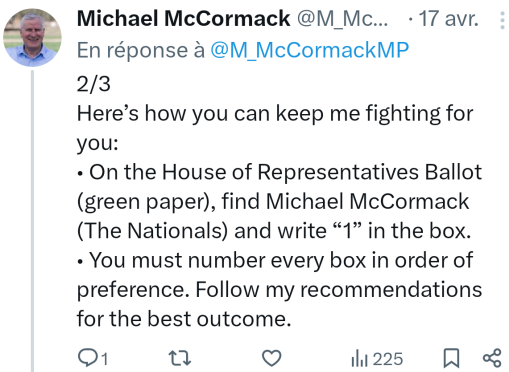


Fig. 6.3: Two campaign-related posts shared by candidate Christine Onley on X during the 2025 Australian federal election. The original posts can be found at <https://x.com/ChrisOnleyLCR/status/1914399964072902675> and <https://x.com/ChrisOnleyLCR/status/1918228690695274875>. For better readability, the images originally attached to these posts (both being the "how to vote" card from Figure 6.2a.) have been removed. In the first post, Onley emphasizes that voters only need to place a 1 next to her name and that the rest of the preferences are up to them. In the second post, this emphasis on voter freedom is absent.

In contrast, McCormack writes in his April 17 post (Figure 6.4a): “Follow my recommendations for the best outcome,” which suggests some kind of strategic voting. This raises a broader question: are these suggested rankings designed to genuinely reflect ideological proximity, or are they a strategic tool meant to weaken rival parties? Interestingly, this phrasing disappears in a later post on April 21 (Figure 6.4b), leaving the strategic intent behind McCormack’s how-to-vote card ambiguous.

Therefore, several questions remain. Are these cards merely intended to teach voters how to cast a valid ballot? If not, do they aim to help voters express their true preferences, or are they designed



(a) Post from April 17, 2025.

(b) Post from April 21, 2025.

Fig. 6.4: Two campaign-related posts shared by candidate Michael McCormack on X during the 2025 Australian federal election. The original posts can be found at https://x.com/M_McCormackMP/status/1912780006163685441 and https://x.com/M_McCormackMP/status/1914205041185137021. For readability and space considerations, the images originally attached to these posts (both being the "how to vote" card from Figure 6.2b.) have been removed, as well as the portion of the April 17 tweet referring to the Senate. In the first post, McCormack explicitly encourages voters to follow the recommended ordering to "get better outcomes".

to guide them toward a tactical vote? Overall, wouldn't it be preferable for voters to determine their own rankings independently? If so, how can they be supported in doing so?

An alternative approach would be to help voters construct their preferences directly. The Build a Ballot website [Bui], which lists all candidates by district, also provides direct links to party or candidate webpages (when available), giving voters convenient access to political platforms. This centralized and easily navigable resource offers a strong incentive to design methods that support voters in eliciting their own preferences.

6.1.1 Comparison-Based Elicitation

Rather than requiring voters to produce a full ranking all at once, it is more natural to allow preferences to emerge progressively through simple pairwise comparisons. This method is not only cognitively less demanding, but also reflects the way people typically evaluate alternatives when relevant information is readily accessible. Moreover, to adapt to the varying levels of time and effort that voters are willing to invest, the sorting process should be flexible—allowing voters to stop at any time while still receiving a meaningful estimate ranking.

This motivates the use of *anytime sorting algorithms*—sorting procedures that can be interrupted at any point and still return an estimate of the final sorted list. Such algorithms provide intermediate outputs after each comparison. The core challenge is to design algorithms that require as few comparisons as possible, which is particularly important when comparisons are costly, as is often the case with human judgments.

The relevance of this approach extends beyond voting. For example, in scenarios involving human input, assigning a utility score to each item is often unreliable, whereas pairwise comparisons tend to yield more accurate judgments [GSS09]. In supervised learning tasks based on ordered data—such as evaluating the quality of machine translations—providing sorted lists instead of numerical scores is advantageous, since human comparisons are considerably more time-consuming than the computational operations performed by machines. To optimize overall efficiency, it is therefore crucial to distinguish the complexity of comparisons from the total computational complexity. In this context, anytime sorting has two main applications: instead of performing exact sorting on a small part of the raw input, one may wish to sort approximately more inputs. Uncertainty on the input and processing rates make it desirable to have an anytime approach. Also, users may be worn out after a certain amount of questions and want to stop the sorting process prematurely, so it is essential for the algorithm to generate accurate intermediate estimates that closely approximate the final sorted list.

Similar problems arise in situations where the considered items involve massive datasets requiring data transfers for each comparison [MF18]. The expense associated with each comparison necessitates the exploration of efficient algorithms that minimize the number of comparisons. Furthermore, practical constraints may require the sorting process to be interrupted before the algorithm's termination, so we need a reliable and accurate estimate of the sorted order based on the progress made thus far.

6.1.2 Related Work

Limiting the number of comparisons when sorting is a well-known problem [Cor+09]. Among many famous algorithms such as quicksort, mergesort, heapsort, binary insertion sort and Shellsort, the Ford-Johnson algorithm [FJ59] stands out as very close to the theoretical limit in number of comparisons, and even optimal for certain values [Pec04]. These algorithms do not consider a possible premature interruption.

A related problem is that of sorting under partial information. This problem entails determining a total order that is partially known, in the sense that a compatible partial order is available [Car+10]. Many variants of the sorting problem can be found within the realm of partial information, including scenarios such as ranking the k best values [DM18]. While the partial information framework does not consider intermediate estimates, similarities with our contributions exist as we propose an algorithm that exploits the partial information extracted so far to determine the next comparison. Furthermore, partial information relates to classical sorts when considering empty information. For example, initializing Algorithm 1 in [Car+10] with no prior amounts to the classical binary insertion sort, and their Algorithms 2 and 3 are both equivalent to mergesort (in its bottom-up version, which we discuss further in this chapter).

Sorting when the probabilistic distribution of the inputs is known is another relevant problem to consider. Moran and Yehudayoff [MY16] propose a variant of insertion sort as a solution, which has also been employed by Peters and Procaccia [PP21] for sorting with human comparisons. However, our approach differs in two main aspects. Firstly, our algorithms focus on providing good intermediate estimates throughout the sorting process. Secondly, we assume equal probabilities for all input permutations. In that case, their algorithm is equivalent to binary insertion sort.

The potential interruption in the sorting process places us within the domain of anytime algorithms, which maintain result estimates at all times. Surprisingly, sorting has received limited attention in this literature, with studies focusing on specific algorithms like selection sort, Shellsort, and quicksort [GZ96; Hor88]. However, the metrics used to measure the estimation error are not always true distances. In our benchmark, we exclude selection sort due to its $\mathcal{O}(m^2)$ comparison complexity¹, while including Shellsort and quicksort for evaluation.

Among related problems, *progressive algorithms* can also be interrupted at any time, but the focus is on provable bounds for worst-case performance rather than on empirical average efficiency [Ale+14].

¹Since the elements to be ordered in our setting are the m candidates, all forthcoming results will be expressed in terms of m rather than the conventional notation n . This choice is made to preserve consistency with the notation used throughout the rest of the thesis.

Contract-based algorithms also make a trade-off between time and accuracy, but assume that the available time is known in advance [Zil95; Zil96].

One of the most closely related studies is the work by Giesen et al. [GSS09], which delves into the link between the number of comparisons performed and the estimation error. Using *Spearman's footrule metric* (defined as the sum of the absolute differences between the ranks of elements in the current list and in the sorted list) to evaluate the error, they show that, for a list of size m , the error after $k \leq m(\ln(m) - 6)$ comparisons is at least $m^2 e^{-(k/m)-6}$ in the worst case. Conversely, they introduce ASort, an anytime algorithm with guarantee that the error is less than $m^2 e^{-(k/6m)+1}$. While these results show that ASort is asymptotically optimal in some sense, the ratio between the two bounds above ($e^{5k/6m+7}$) is at least $e^7 > 1000$ and potentially unbounded. This means that the practical performance of ASort remains to be investigated. One of the goals of our work is to provide a practical benchmark for ASort while also introducing new algorithms with superior empirical performance.

6.1.3 Contributions

This chapter builds on several joint papers with François Durand and Fabien Mathieu: [CDM23b; CDM23a; CDM24; CDM25]. The main contributions are as follows.

- I conducted the discussion about the Australian federal elections, notably the analysis of how-to-vote cards and online voting assistants.
- In [CDM24], we consolidated and refined our previous work introduced in [CDM23b; CDM23a].
 - We generalized the work on ASort by Giesen et al. [GSS09] under the anytime framework and proposed the anytime sorting problem, where the *performance profile* of an algorithm is measured by the *Spearman's footrule metric* between its tentative estimates and the perfectly sorted list, as a function of the number of comparisons already performed.
 - We reexamined classical sorting algorithms from the perspective of anytime sorting. Notably, we proposed simple heuristics called *estimators*, which leverage the available partial information at each step to approximate the final sorted result.
 - We introduced *multizip sort*, an enhanced variant of merge sort tailored for anytime use.

- We introduced the *Corsort* family of anytime sorting algorithms, which rely on estimators to determine the next comparison to perform.
- Using extensive simulations, we compared the behavior of ASort, Multizip Sort, and Corsort, analyzing three types of complexity: the comparison complexity, the global complexity, and the interruption complexity (*i.e.*, the computational cost of producing an intermediate estimate when the sorting process is interrupted). We also evaluated the quality of their intermediate estimates.
- The estimators proposed in [CDM24] come with a high interruption cost of $\mathcal{O}(m^3)$. This issue is addressed in [CDM25], where Fabien Mathieu introduced a new estimator, denoted Y , tailored for merge sorts. This estimator preserves the qualities of the previous ones, while reducing the interruption cost to $\mathcal{O}(m \log m)$.

Table 6.1 summarises the computational complexities. The interruption cost of each algorithm depends on its estimator, indicated in its name (e.g., Multizip- ρ uses the estimator ρ). If unspecified, the algorithm uses its native estimator.

6.1.4 Limitations

The anytime sorting algorithms we propose are designed to address specific scenarios, and their suitability depends on the following assumptions:

- It is impractical or unreliable to assign numerical utilities to the items to sort.
- The sorting process may be interrupted due to tight time constraints or resource limitations.
- An approximate ranking is acceptable, although the approximation error does matter.
- Good empirical performance is required, possibly at the expense of formal guarantees.
- The cost of comparisons is orders of magnitude higher than other computing operations.
- Worse case scenarios and non-uniform dataset/inputs are not the point of the study.

If these assumptions do not hold, alternative solutions may be more appropriate. For example, algorithms like Ford-Johnson are well-suited for situations with costly comparisons and a low likelihood of interruption; contract-based algorithms offer better suitability when interruption times are known and formal guarantees are required. ASort offers similar guarantees, even though it is outperformed experimentally.

	Comparison complexity	Global complexity	Interruption complexity	Interest(s)
Ford-Johnson- ρ	$\sim m \log_2(m)$	$\mathcal{O}(m^2)$	$\mathcal{O}(m^3)$	Comparison complexity
Binary insertion- ρ	$\sim m \log_2(m)$	$\mathcal{O}(m^2)$	$\mathcal{O}(m^3)$	Comparison complexity
Quicksort- ρ	$\mathcal{O}(m \ln(m))$	$\mathcal{O}(m \ln(m))$	$\mathcal{O}(m^3)$	Fast termination in practice
ASort	$\mathcal{O}(m \ln(m))$	$\mathcal{O}(m \ln(m))$	$\mathcal{O}(m)$	Good native estimator with theoretical guarantees
Top-down merge- ρ	$\sim m \log_2(m)$	$\mathcal{O}(m \ln(m))$	$\mathcal{O}(m^3)$	Complexities
Bottom-up merge- ρ	$\sim m \log_2(m)$	$\mathcal{O}(m \ln(m))$	$\mathcal{O}(m^3)$	Good performance profile, complexities
Multizip- ρ	$\sim m \log_2(m)$	$\mathcal{O}(m \ln(m))$	$\mathcal{O}(m^3)$	Very good performance profile, complexities
Multizip- Y	$\sim m \log_2(m)$	$\mathcal{O}(m \ln(m))$	$\mathcal{O}(m \ln(m))$	Very good performance profile, complexities
Corsort - ρ	$\mathcal{O}(m^2)^*$	$\mathcal{O}(m^4)^*$	$\mathcal{O}(m \ln(m))$	Best performance profile empirically, comparison complexity*

Tab. 6.1: Cost of the algorithms considered in this chapter (we exclude heapsort and Shellsort due to their poor termination times, see Figure 6.12). For readability, we only include the most efficient merge sort variant with the Y estimator; note that applying Y to other merge-based algorithms would similarly reduce their interruption cost from $\mathcal{O}(m^3)$ to $\mathcal{O}(m \log m)$. All complexities are well-known results except for the last three rows.

* Simulations suggest $\mathcal{O}(m \ln(m))$ comparisons, yielding a global complexity of $\mathcal{O}(m^3 \ln(m))$ operations.

Furthermore, our proposal does not rely on any specific prior information about the input list. This makes our proposal very generic but also means that if such prior exists, we do not leverage it to reduce the average number of comparisons required, as it is done by Peters and Procaccia [PP21].

Lastly, we have not provided a formal proof that our Corsort algorithm has an average comparison complexity in $\mathcal{O}(m \ln(m))$. This conjecture is only supported by simulations demonstrating its superior performance compared to heapsort, Shellsort, and quicksort/ASort.

6.1.5 Roadmap

The rest of the chapter is organized as follows. Section 6.2 formally defines anytime sorting, and presents a way to transform any sorting algorithm into an anytime version. Section 6.3 presents two new algorithms designed for efficient anytime sorting: *Corsort* and *multizip sort*. Section 6.4

evaluates the proposed solutions through simulations. Section 6.5 discuss the real-life scenario of Australian federal election. Section 6.6 concludes.

6.2 Transforming Classical Sort Into an Anytime Version

Formally, we want to sort a list $X = (X[1], \dots, X[m])$ of items, where $m > 0$, by performing comparisons of the type: is $X[i]$ less than $X[j]$? An *anytime sorting algorithm* is an algorithm capable, after k comparisons have been performed, of returning an *estimate* X_k of the result, for each $k \in \mathbb{N}$. By convention, if k is greater than the number of comparisons needed for the algorithm's termination, then X_k is the sorted list.

We measure the error of an estimate X_k by the *Spearman's footrule metric* between X_k and the sorted list [DG77], defined as the sum S_k of the absolute differences between the ranks of elements in X_k and in the sorted list:

$$S_k = \sum_{i=1}^m |r(X_k[i]) - i|,$$

where $r(x)$ denotes the rank of x in the sorted list.

The function $k \rightarrow S_k$, which represents the evolution of the error made as the algorithm runs, is called its *performance profile*.

Ideally, we are looking for an anytime sorting algorithm whose performance profile is consistently lower than that of the other algorithms tested.

6.2.1 Classical Sorting

Some classical algorithms can be seen as anytime because they maintain a list that converges to the sorted list and can be used as an estimate X_k . This is the case of quicksort, mergesort, binary insertion sort, and Shellsort, all of which we shall implement in a way that is favorable to the spirit of the original algorithm. Note that an anytime algorithm is capable of returning an estimate after each comparisons, and not only after a complete step of execution. The following definitions of X_k take into account that particular setting.

- **Quicksort:** The position of the pivot is updated in the list after each comparison, so that X_k is the list consisting of the left sublist, the pivot, and the right sublist placed in their current positions, followed by the rest of the list that has not yet been processed.

- **Mergesort:** X_k is the list consisting of blocks of fixed size arranged in their current order.
- **Binary Insertion Sort:** X_k consists of a sorted prefix of fully inserted elements, possibly followed by one element currently being inserted (partially moved), and the remaining elements still in their original order.
- **Shellsort:** X_k is the input list modified by applying, in order, the first k comparisons prescribed by the gap sequence. Each comparison may result in a swap, and the list reflects only these k local updates; all other elements remain unchanged from their previous positions.

Changing the order in which comparisons are made may improve the intermediate estimates X_k . For mergesort, the standard recursive implementation, also called *top-down*, sorts the left part, then the right part of each considered sub-list (recursively), and reunites them by a merge procedure [Cor+09, Chapter 2]. However, it is also possible to go through it *bottom-up*, by sorting sub-lists of increasing size [Knu98, Chapter 5.2.4]. Figure 6.5 illustrates this difference on an example² of size 8: the top-down execution sorts completely the left side of the list before completely sorting the right side of the list, whereas the bottom-up execution starts by sorting all lists of size 2, both on left and right sides. In all classical mergesort implementations, the merge procedure is a single block. We propose a variant of the bottom-up implementation, which we call *multizip sort*, where all merge procedures of a given depth of the recursion tree are interleaved. Figure 6.5c illustrates the idea: the first part of the execution is unchanged and sorts all lists of size 2. Then, when *bottom-up* sorts the two lists of size 4 sequentially, *multizip* alternates comparisons in the two lists until they are both sorted. The rationale for proposing *multizip sort* will be discussed in Section 6.3.1. Note that all the considered variants of mergesort make the exact same comparisons, but not in the same order. Intuitively, bottom-up and multizip ensure that the information available on the elements of the list (through comparisons) is more balanced. This should be more favorable for the intermediate estimates X_k , as Section 6.4.2 will verify experimentally.

For quicksort, balancing the comparisons is equivalent to the ASort algorithm [GSS09], using quickselect [Hoa61] as a subroutine for median identification. Specifically, ASort determines the median, separates smaller from larger elements, and proceeds recursively. When utilizing quickselect to identify the median, which uses successive pivots like quicksort, ASort performs precisely the same comparisons as quicksort, albeit not in the same order. This is illustrated in Figure 6.6.

Certain other classical algorithms allow to obtain an estimate X_k by a natural transformation of their current state. For example, heapsort keeps in memory the heap associated to the partially sorted list.

²For brevity, we omit the commas to note the lists taken as examples in the tables and figures. For example, the list (a, b, c, d) is noted as $(abcd)$.

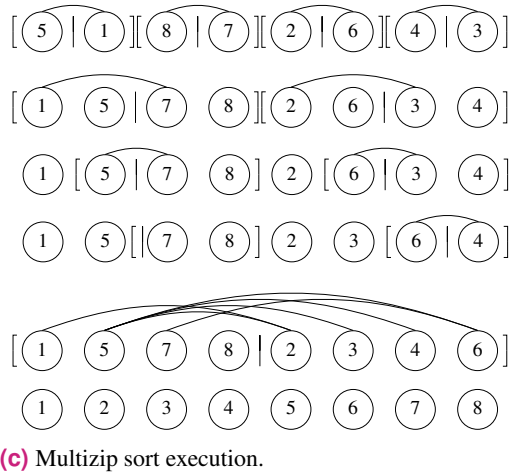
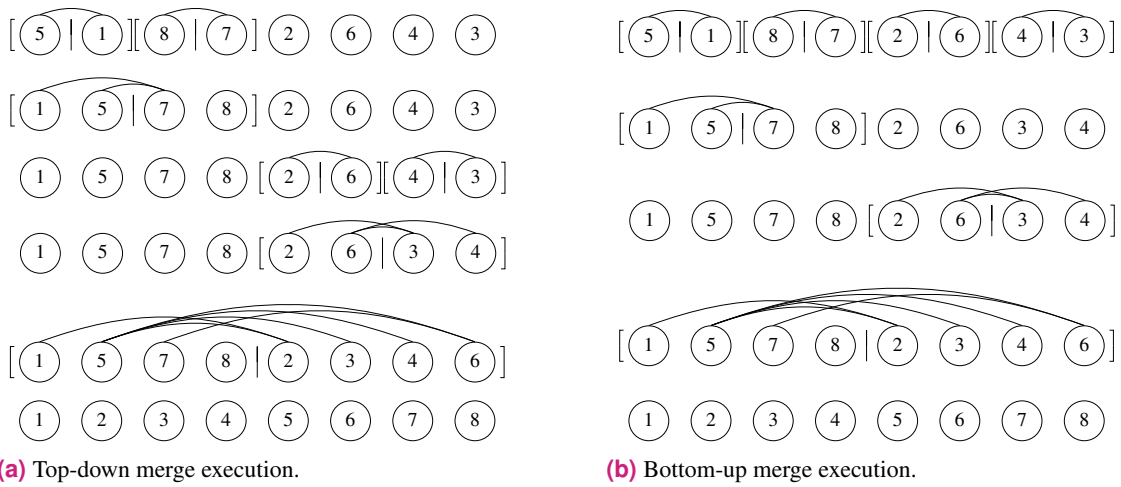


Fig. 6.5: Sorting the list $X = (51872643)$ with top-down merge, bottom-up merge and multizip sort. Each edge represents a comparison. The bracket notation $[|]$ delimits two sublists already sorted that are being merged.

To obtain a fair estimate from the algorithm, we can first go through the heap (backwards), then through the elements already sorted (forwards).

Finally, some algorithms like Ford-Johnson, whose internal state does not have a structure naturally close to a list, do not seem to have a natural estimator. Since it is not always trivial to translate the execution of an algorithm into intermediate estimators, we show in the next section how to produce X_k for any comparison-based sorting algorithm.

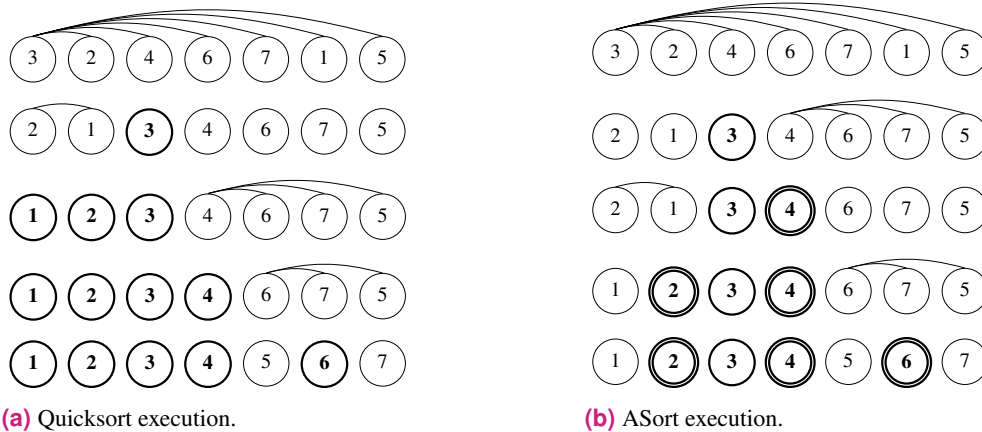


Fig. 6.6: Sorting the list $X = (3246715)$ with quicksort or ASort. A node appears in bold if it has already been used as a pivot: it partitions the list into smaller elements on the left and larger ones on the right. Intermediate steps that do not perform any comparison are omitted. For ASort (6.6b), we use quickselect [Hoa61] as median identification subroutine. Each step represents the application of a pivot, and each edge represents a comparison. A node is circled twice if it has been identified as a median. In the third step, since the median 4 has been found, we must compute the median of the left sub-list (213); but note that it is useless to make any comparison with element 3, previously used as a pivot, because it is already in its final position.

6.2.2 Universal Estimators

To make an anytime version of any sort, we propose to use an estimator that ignores the execution details and solely relies on the historical record of the comparisons made.

Let $C_k = \{X[i_1] < X[j_1], \dots, X[i_k] < X[j_k]\}$ be the result of k comparisons. C_k defines a partial order \preceq_k on the elements of the list by transitive closure³. We call *estimator* a function that associates to any partial order one of its *linear extensions*, i.e. a compatible total order.

An optimal estimator would always return a total order that minimizes the expected Spearman’s footrule distance to a uniformly random linear extension of the input partial order. Assuming the linear extensions of \preceq_k are known, the problem reduces to solving an instance of the assignment problem [Kuh55], where the cost of assigning element x to position i is the total expected error across all linear extensions. Since the Spearman distance is based on the L_1 norm, one can equivalently associate each element with its median rank across the linear extensions; this yields

³We assume for clarity that all elements are distinct. In case of redundancy, uniqueness can be enforced by appending to each element x its index i in the initial list. For example, $(17, 42, 42)$ should be seen as $((17, 1), (42, 2), (42, 3))$. When sorting the list, it is possible to be in a situation where we already know that $(17, 1) \preceq_k (42, 2)$ but we do not yet know how to compare $(17, 1)$ and $(42, 3)$.

an optimal order when all medians are distinct integers. However, enumerating the set of linear extensions is a #P-complete problem [BW91]. Consequently, the computational cost of the optimal estimator is prohibitive.

Therefore, we propose to use classical *score-and-sort* heuristics [CU20; Rob97]: in case of interruption, each element gets a score that reflects its estimated ranking in the list, and we return the list associated with sorting the scores. Recall that global complexity is distinct from comparison complexity: we assume that sorting m numerical scores is much faster than comparing two elements of the list. In case of ties, we return the elements in their original order. By a slight abuse of language, we also call *estimator* the score function itself.

To build our heuristics, in [CDM24], we define descendants and ancestors as follows: if x is an element of the list,

$$d_k(x) = |\{y \in X : y \preceq_k x\}| \quad \text{and} \quad a_k(x) = |\{y \in X : x \preceq_k y\}|$$

are respectively the number of known descendants and ancestors of x in \preceq_k (x is included in both sets by convention). Those quantities will be used in the score function, that is why we need to compute them. A simple way to compute d_k and a_k for a given k is to build the transitive closure of C_k , which can be done in $\mathcal{O}(m^3)$ using the Floyd-Warshall algorithm [Cor+09, chapter 25.2].

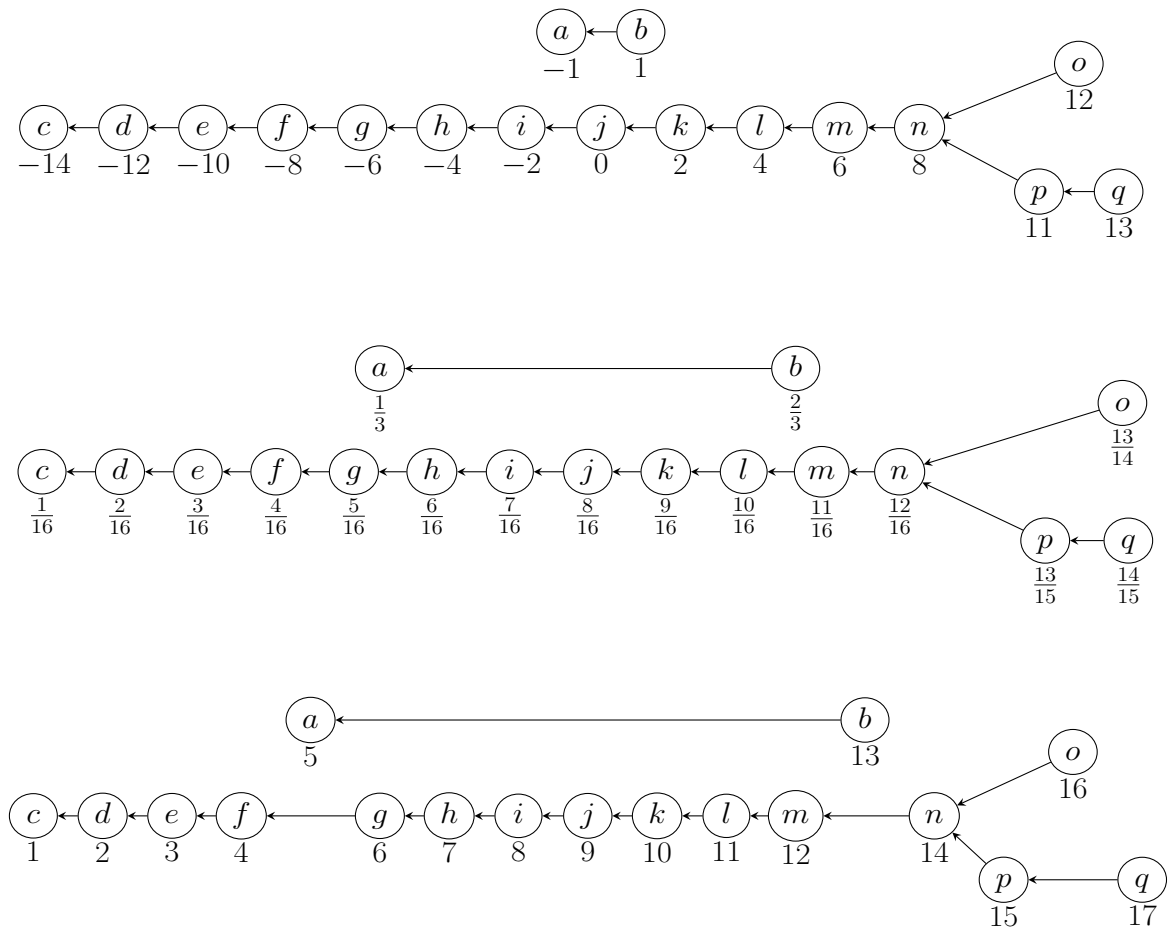
Thus, to adapt a sorting algorithm into an anytime version, it suffices to construct the partial order only in case of interruption, using the Floyd-Warshall algorithm, at a cost of $\mathcal{O}(m^3)$. Each element is then assigned a score based on its position in the resulting partial order, and the elements are sorted accordingly.

Naturally, an interruption cost of $\mathcal{O}(m^3)$ leaves room for improvement. In [CDM24], we identified this limitation and left it as an open problem. In subsequent work [CDM25], Fabien Mathieu addressed this issue for merge-based algorithms by designing a new estimator tailored to the specific structure of the partial orders they generate. For now, we focus on the contributions from [CDM24]. The improved score function introduced in [CDM25] will be presented in Section 6.2.3.

Let us now consider the two score functions defined below:

- Δ_k , defined by $\Delta_k(x) = d_k(x) - a_k(x)$, assigns to each x a score that reflects the average between its lowest and highest possible positions.
- ρ_k , defined by $\rho_k(x) = d_k(x)/(d_k(x) + a_k(x))$, positions x as if its descendants and ancestors were on average regularly spaced within the whole list.

If there is no ambiguity, we will omit the index k . The score functions Δ , ρ , but also the median rank med are valid estimators: one can check that if an element is greater than another in a partial order, then its score is higher, which ensures that the returned estimate is a linear extension. In particular, when the result of all comparisons is known, sorting the elements according to the score function returns the sorted list.



Estimator	Returned estimate	\bar{S}
Δ	$(c d e f g h i a j b k l m n p o q)$	16.2
ρ	$(c d e f g a h i j k l b m n p o q)$	14.0
med	$(c d e f a g h i j k l m b n p o q)$	13.9

Fig. 6.7: Example of score-based selection of a linear extension. The input partial order, represented by the edges of the graph, is the transitive closure of: $a < b$, $c < d < \dots < n < o$, $n < p < q$. Nodes horizontal positions are proportional to three different scores functions - from top to bottom: Δ , ρ , and med . Returned estimates and expectation \bar{S} of the error S are also provided for completeness.

Figure 6.7 shows a partial order where the three estimators produce different outputs. In this example, ρ performs better than Δ by more accurately placing the small chain component (nodes a and b) relative to the larger Y-shaped component (nodes c through q). However, it is outperformed by the median rank estimator med , which happens to be optimal in this case since all median values are distinct.

The superior performance of ρ was confirmed empirically: it usually outperformed Δ in our simulations. Nonetheless, Δ has specific advantages in the context of Corsort. It will be discussed in Section 6.3.2. Due to its empirical performance, ρ was chosen as the estimator for all simulations reported in Section 6.4, except for multizip, for which we designed a dedicated estimator, developed in Section 6.2.3.

In a nutshell, we found an estimator with strong empirical performance in terms of estimate quality, but with a high interruption cost. We now turn to the question of reducing this cost.

6.2.3 Estimator Tailored For Mergesorts

At any moment during mergesort, the elements are partitioned into disjoint substructures, either sorted chains or the current merging Y-structure, which actually corresponds to the case illustrated in Figure 6.7. Since these substructures do not overlap, the relative position of an element within the entire partial order depends solely on its position inside its local structure. This independence enables computing positional scores separately and efficiently within each component.

For a chain of size $B > 0$, the optimal solution consists in spacing all elements by $\frac{1}{B+1}$. For a Y-structure of total size S , consisting of two branches of sizes $L > 0$ and $R > 0$ connected to a trunk of size $B > 0$, the optimal solution is described in Figure 6.8.

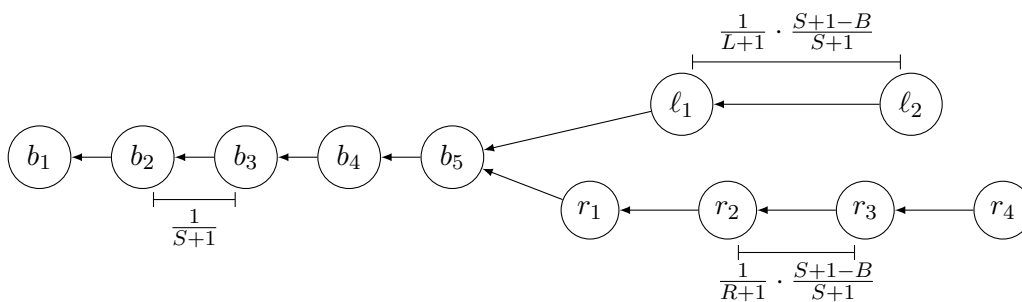


Fig. 6.8: Estimator Y assigning relative positions in a Y-shaped merge structure with trunk size B , left branch size L , and right branch size R , totaling $S = B + L + R$. Positions in the trunk are spaced by $\frac{1}{S+1}$, while branch positions are scaled by $\frac{S+1-B}{S+1}$ and evenly distributed within each branch.

Thus, the score $Y(x)$ of an element x belonging to a Y structure (with possibly empty left and right branches) is defined as

$$Y(x) = \begin{cases} \frac{\text{rank}(x)}{S+1} & \text{if } x \text{ is in the trunk (size } B), \\ \frac{B}{S+1} + \frac{S+1-B}{S+1} \cdot \frac{\text{rank}_L(x)}{L+1} & \text{if } x \text{ is in the left branch (size } L), \\ \frac{B}{S+1} + \frac{S+1-B}{S+1} \cdot \frac{\text{rank}_R(x)}{R+1} & \text{if } x \text{ is in the right branch (size } R), \end{cases}$$

where $\text{rank}_\bullet(x)$ denotes the position of x within the respective component.

Since each element's score depends only on its local merge structure and its rank within it, the computation of the estimate can be efficiently performed in case of interruption. Each merge operation maintains three progress pointers—one for each branch and one for the trunk, to track merging progress and update local ranks in constant time. As a result, all Y scores can be computed in a single linear pass, yielding a total cost of $\mathcal{O}(m)$. Finally, since the interruption procedure involves sorting the elements based on these scores, the total interruption cost is $\mathcal{O}(m \log m)$, which remains lower than that of "naive" estimators like ρ .

How does estimator Y compare to ρ in terms of performance? In the partial order shown in Figure 6.7, both estimators yield the same sorted list, as illustrated in Figure 6.9. This similarity extends beyond this specific case. As confirmed by simulations, multizip equipped with either ρ or Y shows equivalent performance (see Figure 6.13 in Section 6.4).

This similarity can be intuitively explained by the fact that both estimators rely on structural information that closely reflects progress in the merge process. While ρ captures position via the ratio of descendants to ancestors, Y estimates relative rank within local merge components. In typical mergesort executions, these two perspectives often align, resulting in nearly identical estimates. This confirms that the main motivation for using Y over ρ in merge sorts lies in its lower interruption cost.

Finally, remember that the Y score is only applicable when the underlying structure permits it—namely, chains and Y -shaped components. In other sorting procedures, such structural regularity is absent, making ρ the necessary choice.

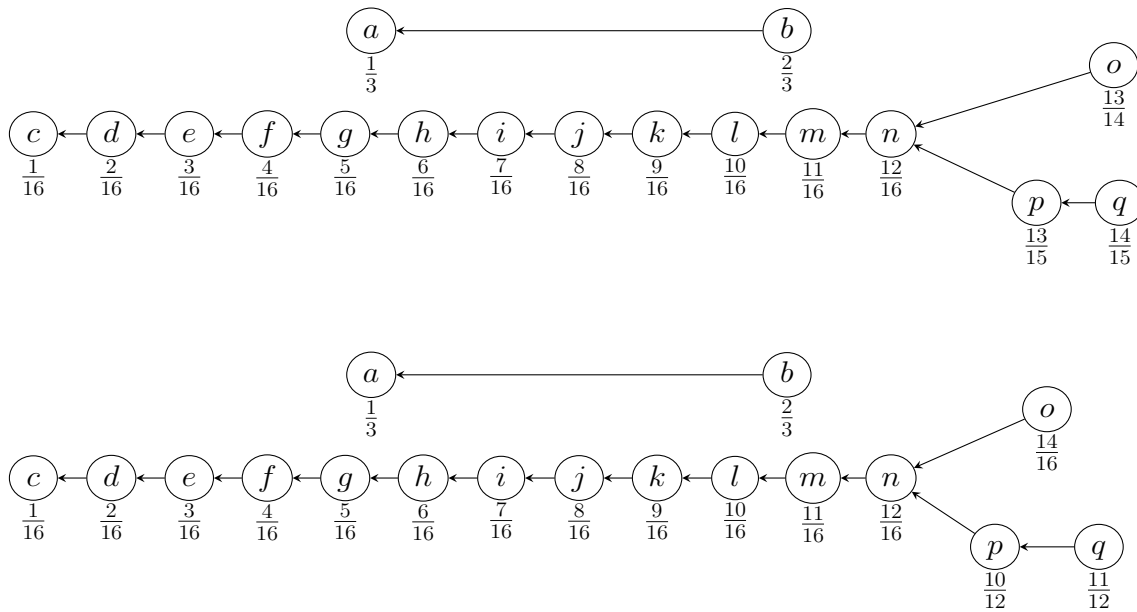


Fig. 6.9: Comparison of scores assigned by estimators ρ (top) and Y (bottom) on the same partial order. In this case, ρ and Y yield the same estimate: $(cdefghiajklmnopq)$.

6.3 New Algorithms

In this section, we introduce two new algorithms tailored for efficient anytime sorting: multizip sort and Corsort. Multizip sort is a variant of mergesort well-suited for the Δ , ρ , and Y estimators. In contrast, Corsort is a family of comparison-driven sorts that actively select the next pair to compare based on the current partial order, aiming to improve the estimate as efficiently as possible. Both algorithms are designed to produce high-quality intermediate estimates while keeping the overall comparison complexity low.

6.3.1 Multizip Sort

As previously discussed, a standard mergesort execution typically produces a partial order composed of chains (sorted sublists) and a single Y -shaped structure (the ongoing merge), as shown in Figures 6.7 and 6.9. However, such structures tend to distort the estimates produced by Δ and ρ , which motivated the introduction of the multizip sort in Section 6.2.1. In multizip, the partial order consists of several Y -shaped components of similar size, better suited to these estimators. In essence, multizip is a mergesort variant tailored to the structural assumptions behind Δ and ρ .

Subsequently, we introduced the estimator Y , which achieves similar performance to ρ but offers lower interruption cost. For this reason, simulations primarily use multizip with Y , although multizip remains compatible with Δ , ρ , or any valid estimator. This choice was guided solely by efficiency and computational cost.

Multizip execution. In Figure 6.5c, we presented the execution of multizip sort on the list (51872643), which we recall in Figure 6.10 for ease of reading. Here we detail this example.

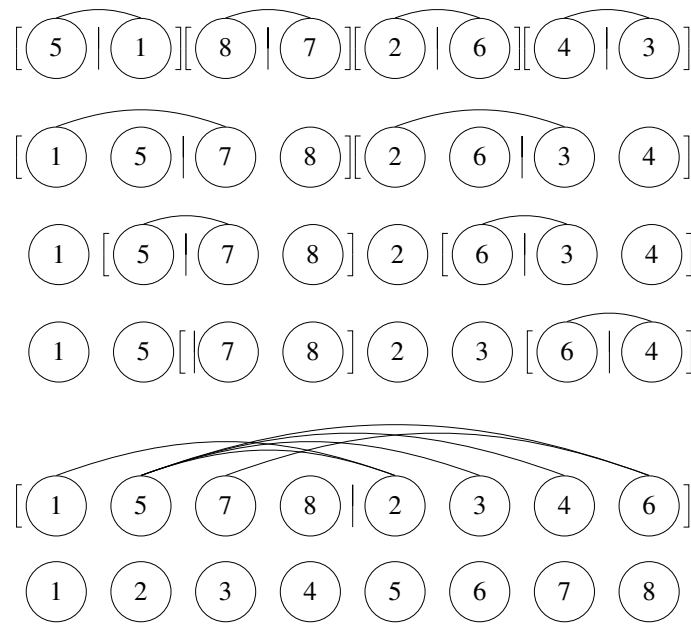


Fig. 6.10: Sorting the list $X = (51872643)$ with multizip sort. Each edge represents a comparison. The bracket notation $[]$ delimits two sublists already sorted that are being merged.

After enough splitting operations, the initial list is decomposed into sublists of size 0 or 1. As, for simplicity, we chose an example whose size is a power of two, the decomposition contains only sublists of size 1 and no empty list. Then the merging process begins (the line numbers below refer to Figure 6.5c or 6.10).

- First multizip merge:
 - Line 1: $Y_1 = [5]$, $Y_2 = [1]$, $Y_3 = [8]$, $Y_4 = [7]$, $Y_5 = [2]$, $Y_6 = [6]$, $Y_7 = [4]$, $Y_8 = [3]$.
Compare $(5, 1)$, $(8, 7)$, $(2, 6)$, and $(4, 3)$, like in bottom-up merge sort.
- Second multizip merge:

- Line 2: $Y_1 = [1, 5]$, $Y_2 = [7, 8]$, $Y_3 = [2, 6]$, $Y_4 = [3, 4]$. Compare (1, 7) and (2, 3).
- Line 3: $Z_1 = [1]$, $Y_1 = [5]$, $Y_2 = [7, 8]$, $Z_2 = [2]$, $Y_3 = [6]$, $Y_4 = [3, 4]$. Compare (5, 7) and (6, 3).
- Line 4: $Z_1 = [1, 5]$, $Y_1 = []$, $Y_2 = [7, 8]$, $Z_2 = [2, 3]$, $Y_3 = [6]$, $Y_4 = [4]$. For Z_1 , no more comparison is needed because Y_1 is empty. Compare (6, 4) to finish merging Y_3 and Y_4 into Z_2 .
- Third multizip merge:
 - Line 5: $Y_1 = [1, 5, 7, 8]$, $Y_2 = [2, 3, 4, 6]$. Compare (1, 2), (5, 2), (5, 3), (5, 4), (5, 6), and (7, 6), like in top-down or bottom-up merge sort.

Note that in multizip sort, the first multizip merge is always identical to the first round of merging operations in bottom-up merge sort. Indeed, each sublist is of size 0 or 1, hence each merging operation involves at most one comparison, which implies that there is no difference between treating each merging operation as a block and performing the necessary comparisons in a round-robin fashion. On the other hand, the last merging operation (between the two sorted half-lists) is always the same for the three variants: top-down merge sort, bottom-up merge sort, and multizip sort. Hence to show the difference between bottom-up merge sort and multizip sort, it is necessary to have an example with three levels of merging, such as the one we present.

Multizip implementation. For the sake of clarity and conciseness, we present the pseudocode of multizip sort (Algorithm 1) in a version that does not sort the list in place.

Algorithm 1 Multizip sort

Input: List X .

Output: List X sorted in ascending order.

```

1:  $Y_1 := X$ .
2:  $\ell := 1$ .
3: while  $\exists i \in \{1, \dots, \ell\}, |Y_i| > 1$  do
4:    $Y_1, \dots, Y_{2\ell} := \text{Split}(Y_1, \dots, Y_\ell)$ .
5:    $\ell := 2\ell$ .
6: while  $\ell > 1$  do
7:    $Y_1, \dots, Y_{\ell/2} := \text{Multizip merge}(Y_1, \dots, Y_\ell)$ .
8:    $\ell := \ell/2$ .
return  $Y_1$ .

```

Multizip sort uses two subalgorithms:

- Split (Algorithm 2) takes several lists and cuts each of them into two lists of equal sizes (up to one element). In multizip sort, it is called enough times so that all lists end up being of size 0 or 1; in particular, all of them are trivially sorted.
- Multizip merge (Algorithm 3) takes an even number of sorted lists $Y_1, \dots, Y_{2\ell}$ and merges them two by two, as in the classical bottom-up merge sort. The particularity here is that instead of completely merging Y_1 and Y_2 (“closing one zipper”), then Y_3 and Y_4 (“closing another zipper”), etc, we perform only the first comparison for each merge operation, then only the second comparison for each merge operation, etc. (metaphorically closing all zippers in parallel, little by little, in a round-robin fashion).

Algorithm 2 Split

Input: Lists Y_1, \dots, Y_ℓ .

Output: Lists $Z_1, \dots, Z_{2\ell}$, where for each $i \in \{1, \dots, \ell\}$, $Y_i = \text{concatenate}(Z_{2i-1}, Z_{2i})$, $|Z_{2i-1}| = \lceil |Y_i|/2 \rceil$, and $|Z_{2i}| = \lfloor |Y_i|/2 \rfloor$.

```

1: for  $i = 1$  to  $\ell$  do
2:    $y_i := |Y_i|$ .
3:    $m_i := \lceil y_i/2 \rceil$ .
4:    $Z_{2i-1} := [Y_i[1], \dots, Y_i[m_i]]$ .
5:    $Z_{2i} := [Y_i[m_i + 1], \dots, Y_i[y_i]]$ .
return  $Z_1, \dots, Z_{2\ell}$ .
```

Algorithm 3 Multizip merge

Input: An even number of lists $Y_1, \dots, Y_{2\ell}$, where each list Y_i is sorted in ascending order.

Output: Lists Z_1, \dots, Z_ℓ , where each list Z_i consists of the elements of Y_{2i-1} and Y_{2i} , sorted in ascending order.

```

1:  $Z_i := [], \forall i \in \{1, \dots, \ell\}$ .
2: while  $\exists i, Y_i \neq []$  do
3:   for  $i = 1$  to  $\ell$  do
4:     if  $Y_{2i-1} = []$  or  $Y_{2i} = []$  then
5:        $Z_i = \text{concatenate}(Z_i, Y_{2i-1}, Y_{2i})$ .
6:       Empty  $Y_{2i-1}$  and  $Y_{2i}$ .
7:     else
8:       if  $Y_{2i-1}[1] \leq Y_{2i}[1]$  then
9:          $Z_i.append(Y_{2i-1}.pop(1))$ .
10:      else
11:         $Z_i.append(Y_{2i}.pop(1))$ .
return  $Z_1, \dots, Z_\ell$ .
```

Multizip complexities. As multizip schedules the same comparisons as mergesort, its comparison complexity is equivalent to $m \log_2(m)$ even in worst case, which is asymptotically optimal [Cor+09]. Each step requires $\mathcal{O}(1)$ computing operations so the global complexity is $\mathcal{O}(m \ln(m))$. If an interruption occurs, we need to build the transitive closure of the comparisons made, which is done in $\mathcal{O}(m^3)$ for the initial estimator ρ and done in $\mathcal{O}(m \ln(m))$ for the Y -estimator.

To summarize, multizip sort provides a structured and efficient comparison schedule inspired by mergesort, well-suited to estimators such as ρ and Δ . When paired with the estimator Y , it offers a promising compromise between performance and computational cost. However, its fixed structure is leaving room for more adaptive strategies. The *Corsort* family of algorithms aims to guide the comparison process dynamically, selecting at each step the most informative comparison with respect to a chosen heuristic.

6.3.2 Comparison-Oriented Sort

We call *Corsort* (*Comparison-ORiented Sort*) a sorting algorithm that, at each step of its execution, selects the next comparison based on the current partial order \preceq_k . We assume that only pairs that are not comparable yet (according to \preceq_k) are chosen until convergence is achieved.

The heart of a Corsort, i.e., the choice of the next comparison, aims at two objectives. Firstly, it must ensure fast termination, i.e., minimize the total number of comparisons. This is a problem of *sorting under partial information*, which amounts to choosing a comparison whose result is as uncertain as possible [Car+10]. To this end, we propose to rely on closeness for a score function. The rationale is that two elements having close scores should indicate a high uncertainty regarding the outcome of their comparison. Secondly, to improve the quality of intermediate estimates X_k , we need to acquire information on elements for which we have little, as these create uncertainty that will be reflected in the error S_k . We introduce $I_k(x) = d_k(x) + a_k(x)$ to represent the amount of “information” acquired on an element x , and we wish to compare elements for which I_k is low. Experimentally, we compared several variants of Corsort and we selected the following one, called Corsort- Δ :

- The next comparison is made by seeking a pair of incomparable items whose Δ scores are as close as possible.
- We use I_k for tie-breaking. Specifically, among the pairs that minimize Δ -closeness, we pick up one that minimizes $\max(I_k(x), I_k(y))$.

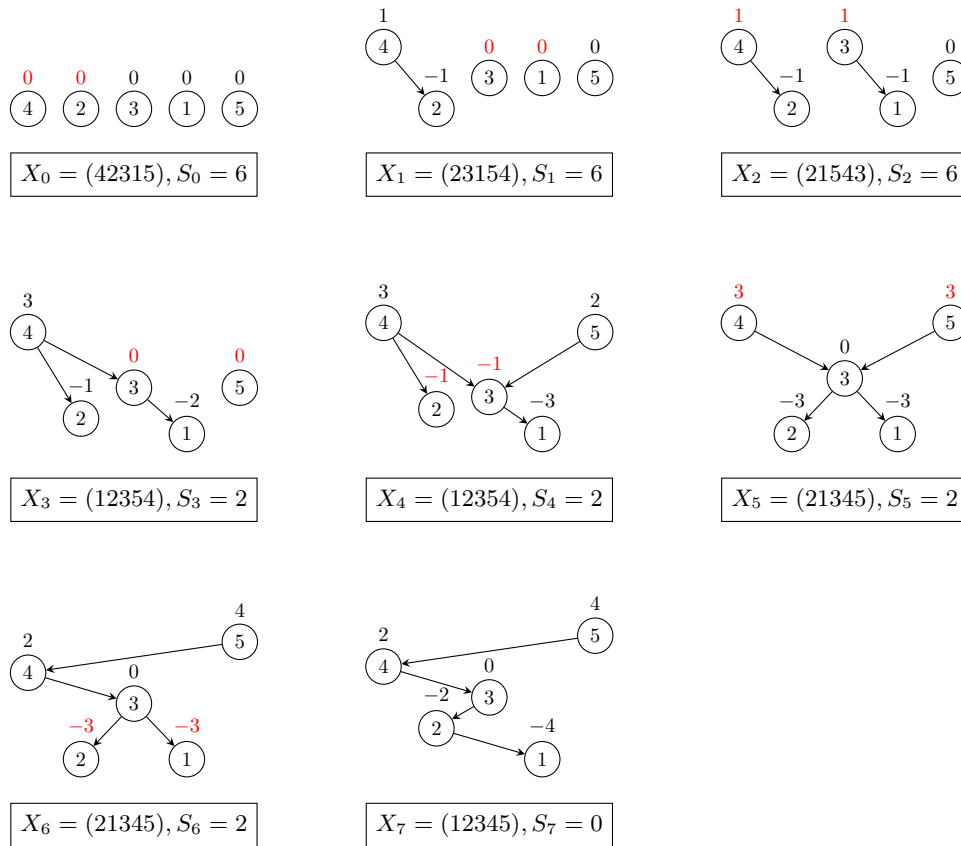


Fig. 6.11: Execution of Corsort- Δ on the list $X = (42315)$. Each step k depicts the partial order after k comparisons. The next comparison to perform is visually highlighted by the two vertices whose values of Δ_k (displayed above each element) are in red. Each subfigure represents ρ_k as each element's height and gives the corresponding estimate X_k based on ρ_k and the associated error S_k .

As for multizip sort, the intermediate estimates X_k can be given by any estimator, but we chose ρ for efficiency reasons.

Corsort execution. As an example, Figure 6.11 shows the execution of the selected Corsort on the list $X = (42315)$. As we argued earlier that ρ is better than Δ , one may wonder why Δ is preferred for the choice of the comparison. An intuition that explains the advantage of Δ is the need of balance between termination and intermediate estimates: ρ can take $\mathcal{O}(m^2)$ distinct values, against $\mathcal{O}(m)$ for Δ . As a result, Δ generates more ties than ρ (a typical case is X_4 on Figure 6.11: $\rho_4(2) < \rho_4(3)$ but $\Delta_4(2) = \Delta_4(3)$), so the tie-breaking rule, which targets accuracy

of the estimate, is used more often. Experimentally, this gives a better performance profile than a ρ -based Corsort, at the price of a (slightly) longer termination.

Corsort implementation. Algorithm 4 describes Corsort- Δ ρ , which uses Δ to choose the next comparison to perform and ρ to estimate the sorted list. Some variants can be obtained by replacing Δ and ρ with other estimators.

Algorithm 4 Corsort- Δ ρ

Input: List X of size m .

Output: List X sorted in ascending order if the algorithm ends with no interruption. An estimate of the sorted list otherwise.

- 1: $M :=$ the identity matrix of size m , representing the empty partial order on X .
 - 2: **while** there is no interruption and $\exists(i, j), M_{i, j} = 0$ **do**
 - 3: Update Δ and I according to M .
 - 4: $\mathcal{S} := \{(i, j), M_{i, j} = 0\}$.
 - 5: $(i, j) := \arg \min_{\mathcal{S}} (|\Delta(i) - \Delta(j)|, \max(I(i), I(j)))$.
 - 6: **if** $X[i] > X[j]$ **then**
 - 7: $(i, j) := (j, i)$.
 - 8: $M :=$ Partial order update($M, (i, j)$).
 - 9: Update ρ according to M .
 - 10: **return** X sorted by ascending value of ρ .
-

Contrary to multizip sort and other classical sorts, Corsort leverages the partial order at each step of its execution. Algorithm 5 updates the partial order M after each comparison, ensuring that the transitive closure is maintained. To do so, it updates the descendants of the ancestors of the greater element with the descendants of the smaller element, and conversely. The cost of this incremental approach is $\mathcal{O}(m^2)$ for each new comparison performed.

Algorithm 5 Partial order update

Input: A matrix M such that $M_{k, \ell} = +1$ if $k \preceq \ell$, $M_{k, \ell} = -1$ if $k \succ \ell$, and $M_{k, \ell} = 0$ if we have not compared k and ℓ yet. A pair (i, j) .

Output: The transitive closure of M with the additional comparison $i \preceq j$.

- 1: descendants $_i := \{k, M_{k, i} = +1\}$.
 - 2: ascendants $_j := \{\ell, M_{j, \ell} = +1\}$.
 - 3: **for** $k \in$ descendants $_i$ **do**
 - 4: **for** $\ell \in$ ascendants $_j$ **do**
 - 5: $M_{k, \ell} := +1$.
 - 6: $M_{\ell, k} := -1$.
 - return** M .
-

Let M be defined with the convention of Algorithm 5 and i be the index of an element. To compute the number of ascendants $a(i)$ or descendants $d(i)$, it suffices to read the row or column i of matrix M :

$$a(i) = \sum_{j=1}^m \mathbb{1}[M_{i,j} = +1],$$

$$d(i) = \sum_{j=1}^m \mathbb{1}[M_{j,i} = +1].$$

The computation of $I(i)$, $\Delta(i)$, and $\rho(i)$ is straightforward:

$$I(i) = d(i) + a(i),$$

$$\Delta(i) = d(i) - a(i),$$

$$\rho(i) = \frac{d(i)}{d(i) + a(i)}.$$

Corsort complexities. Corsort never compares a given pair twice, so it makes at most $m(m-1)/2$ comparisons. At each step, one needs to compute the next pair, compare, then update a and d , for a cost of $\mathcal{O}(m^2)$ operations, which sums up to a global complexity in $\mathcal{O}(m^4)$. The cost of interruption, a score-and-sort in $\mathcal{O}(m \ln(m))$, is comparatively negligible.

Therefore, Corsort incurs a high comparison and computational cost, but allows for efficient interruption, as it incrementally maintains the partial order throughout the process. While the theoretical comparison complexity of Corsort can reach $\mathcal{O}(m^2)$ in the worst case, we shall see in the next section that its empirical performance is significantly better, typically close to $\mathcal{O}(m \log m)$ in practice.

We now turn to a comparative evaluation of these algorithms, examining both their termination time and their anytime performance, and contrasting them with classical sorting methods.

6.4 Evaluation and Comparative Results

To compare the performance of ASort, multizip sort and Corsort with the classical sorting algorithms discussed in Section 6.2.1, we have developed an open-source Python package specifically designed for creating anytime sorting algorithms and evaluating their effectiveness, we discuss of the implementation details in Section 6.4.3.

6.4.1 Uninterrupted Behavior: Termination Time

Let us first analyze the comparison complexity of each selected sorting algorithm. Figure 6.12 shows the termination time (measured in number of comparisons) for the different algorithms. The curves for top-down merge, bottom-up merge, and both multizip sorts coincide as they represent different scheduling variations of mergesort. Similarly, the curves for ASort and quicksort are identical as they involve the same comparisons. We consider values of the list size m ranging from 8 to 2048. For each point, we generate 10,000 random lists and we record the number of comparisons required for a complete sort. The y-axis shows the relative deviation from the theoretical lower bound of $\log_2(m!) = m \log_2(m) - m/\ln(2) + \log_2(2\pi m)/2 + o(1)$ [Cor+09]: a curve closer to 0 indicates closer proximity to the optimal performance. To give a comprehensive view of the results distribution, each algorithm's median is depicted by a dark curve, while the 2.5% to 97.5% quantiles form a 95% confidence interval represented by a light area.

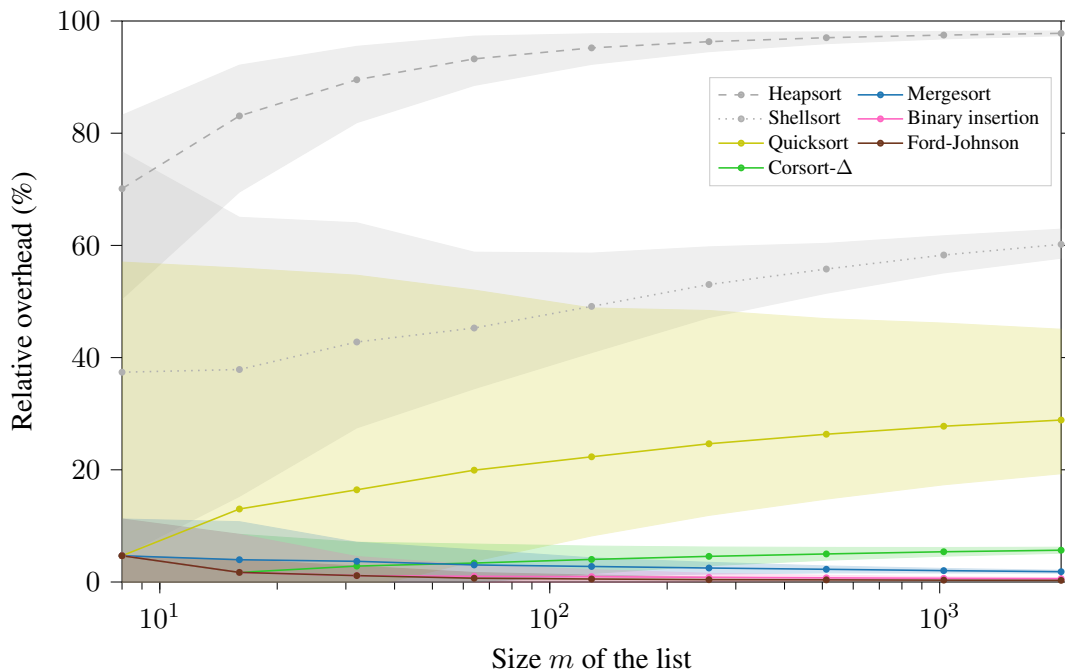


Fig. 6.12: Number of comparisons required for algorithm termination, as a relative overhead compared to the information-theoretic lower bound. Each point is obtained by sorting 10,000 random lists.

We observe that Heapsort requires almost twice as many comparisons as necessary, and Shellsort (here implemented with Ciura's gap sequence [Ciu01]) approximately 60% more. Quicksort shows better performance with less than 30% overhead for the studied values of m , but relying on a pivot selection introduces high variance. The remaining four sorting algorithms demonstrate even lower

overhead: 6% for Corsort, 2% for mergesort, 0.6% for binary insertion, and 0.3% for Ford-Johnson. These algorithms also exhibit negligible variance. Based on these findings, we conclude that Corsort emerges as a promising candidate, surpassing all other algorithms except those with comparison complexities asymptotically equivalent to $m \log_2(m)$ [FJ59]. Moreover, our simulations confirm the asymptotical optimality of mergesort, further reinforcing the position of its multizip variant as a strong candidate for anytime sorting.

6.4.2 Performance Profiles

We now analyze the performance, *i.e.* the quality of the intermediate estimates produced by each algorithm. First, we compare Multizip equipped with ρ and with Y . Figure 6.13 shows their performance profiles on a semi-logarithmic scale to better observe the termination behavior. For each algorithm, we sort 10,000 random lists of size $m = 1000$. After every specified number of comparisons k , the algorithm is interrupted and the estimation error S_k is computed. As in Figure 6.12, we represent the median and a 95% confidence interval. The error is normalized by $m(m-1)/2$. We observe that Multizip with Y performs only slightly better than with ρ , confirming that its main advantage lies in the lower interruption cost. Hence we will equip multizip with Y in the rest of the simulations.

Figure 6.14 displays the performance profiles of the sorting algorithms. Heapsort and Shellsort were excluded due to their poor termination times (cf. Section 6.4.1). The Ford-Johnson algorithm employs the estimator ρ . Quicksort is examined in three variants: its natural implementation; ASort; ASort version equipped with ρ . Three variants of mergesort are also considered: the natural top-down implementation; bottom-up merge equipped with ρ ; and multizip sort equipped with Y . Binary insertion sort is explored with both its natural estimator and the estimator ρ . Finally, as discussed in Section 6.3.2, Corsort utilizes Δ for selecting the next comparison (with additional criteria in case of ties) and ρ for yielding estimates.

For each algorithm, we sort 10,000 lists of size $m = 1000$. After each specified number of comparisons k , we interrupt the algorithm and compute the estimation error S_k . As for Figure 6.12, we represent the median and a 95% confidence interval. The error is normalized by the maximal distance $\lfloor m^2/2 \rfloor$. The bounds from [GSS09] are not displayed because they are irrelevant for the considered parameters: the lower bound starts from 0.005 and rapidly vanishes, while the ASort guaranteed upper bound is always above 1.

First and foremost, Corsort stands out with its remarkable performance profile. It consistently exhibits a decreasing trend with minimal variance, and outperforms other algorithms except for

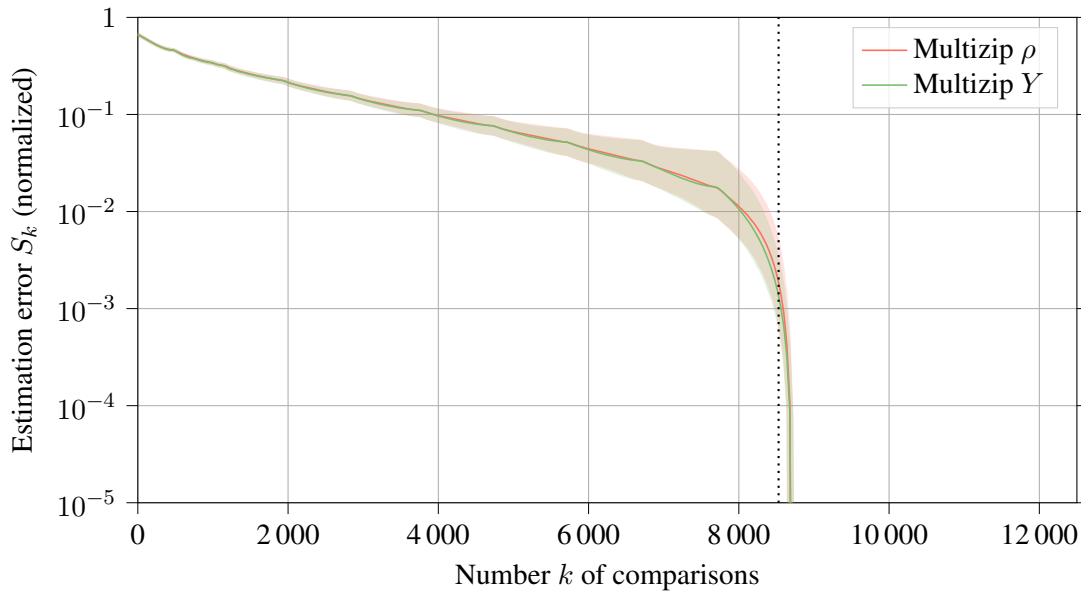


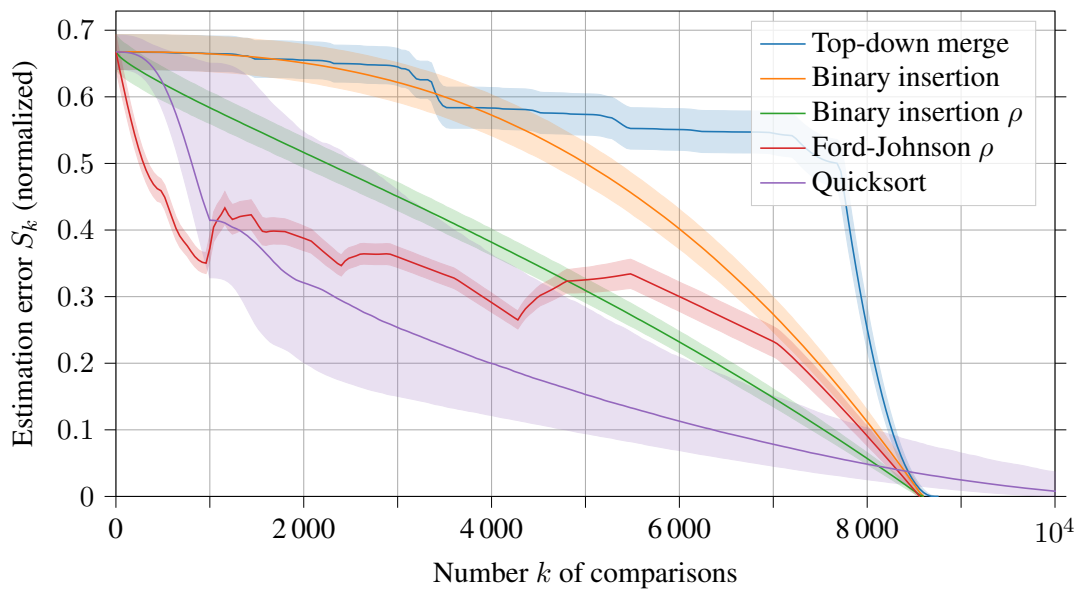
Fig. 6.13: Performance profiles of multizip- ρ and multizip- Y for $m = 1000$, each curve is obtained by sorting 10,000 random lists. Spearman distance is normalized by $m(m - 1)/2$. The dashed line shows the theoretical lower bound $\log_2(m!)$ on the number of comparisons. A semi-logarithmic scale is used to better visualize the termination behavior.

termination. Corsort emerges as a highly commendable anytime sorting algorithm, providing accurate estimates within a reasonable timeframe.

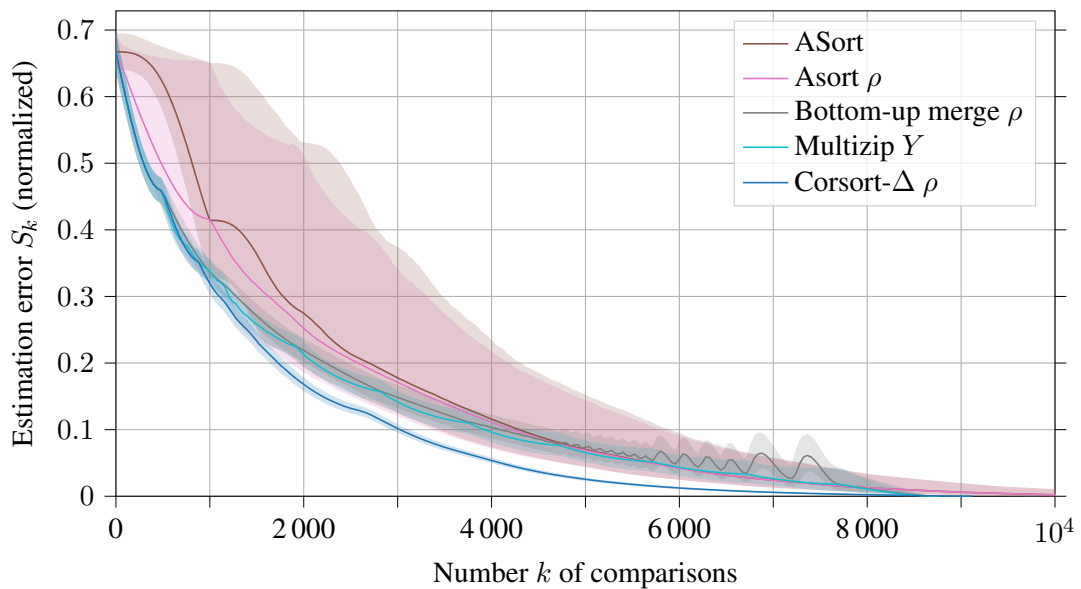
Multizip- Y sort demonstrates notable performance, ranking second almost constantly in terms of the performance profile. It offers a compelling option in scenarios where global speed is of utmost importance, with a termination cost of $\mathcal{O}(m \ln(m))$ total operations, including the interruption cost.

ASort also exhibits a good profile, especially if one focuses on the median values, for which it is slightly better than multizip for $k \in [5700, 7700]$. Its main drawback is a significant variance, shared with Quicksort, primarily explained by their pivot selection process. In contrast, the other sorting algorithms showcase greater consistency in their performance: binary insertion sorts, mergesorts, and Ford-Johnson exhibit relatively low variances, while Corsort demonstrates virtually negligible variance.

Corsort, multizip, and ASort are anytime by design and outperform algorithms such as binary insertion sort and Ford-Johnson. Yet, the differences between them are also significant, as illustrated by Table 6.2, which links the number of comparisons and the estimation error on a few values.



(a) Sorting algorithms not designed for anytime interruption.



(b) Anytime sorting algorithms.

Fig. 6.14: Performance profiles for $m = 1000$. Each curve is obtained by sorting 10,000 random lists. The error S_k is normalized by the maximal distance $\lfloor m^2/2 \rfloor$. Results are dispatched on two subfigures for better visibility. For the computation times, cf Section 6.4.1, and for the running time, cf Section 6.4.3.

# of comparisons	Corsort	Multizip	ASort
4000	5.4 (5.7)	9.7 (11.4)	11.7 (24.4)
6000	1.2 (1.3)	4.4 (6.2)	4.3 (9.0)
8000	0.2 (0.2)	1.1 (2.0)	1.3 (3.4)

Tab. 6.2: Normalized error (in percent) depending on the number of comparisons for $m = 1000$. Each entry shows the median value, then the 97.5% quantile in parenthesis. Numbers are rounded to the first decimal place.

We also observe that the use of the estimators ρ and Y contributes to an improved performance profile. Figure 6.14 only shows the case of binary insertion and ASort to limit the number of curves, but we actually observed the same phenomenon for all algorithms from Table 6.1 that have a native estimator. However, the impact of ρ is limited for ASort, where the gain is relatively small and limited to the left part of the profile.

For bottom-up merge, ρ induces a non-monotonic behavior towards the later stages of the sorting process. Indeed, ρ is optimal when the partial order is a set of independent chains, but ρ is inaccurate when there are also (non degenerated) Y-shapes, like in Figure 6.7. Near completion of the bottom-merge algorithm, the partial orders fluctuate between these two configurations, which leads to the observed phenomenon. For example, the final two “bumps” in Figure 6.14 correspond to the merge of quarters of the list. In contrast, by performing a round-robin on merge operations, our improved algorithm multizip equipped with ρ stays monotonic, as illustrated by Figure 6.13, and remains monotonic when equipped with Y . Consequently, whether it is equipped with ρ or Y , multizip remains consistently below bottom-up merge equipped with ρ .

6.4.3 Implementation

The Corsort Package. The Python code of this article has been published on GitHub (<https://github.com/emczg/corsort>) and PyPI (<https://pypi.org/project/corsort/>) under GNU General Public license v3. The documentation (<https://emczg.github.io/corsort/>) provides detailed instructions for package installation and for the replication of our results.

In particular, the *Notebooks* section of the documentation gives step-by-step instructions to reproduce all the experiments we describe.

- The *Termination times*⁴ notebook explains the computation of Figure 6.12.

⁴<https://emczg.github.io/corsort/notebooks/termination.html>

- The *Comparison of Corsort variants*⁵ notebook explains our selection of the Corsort- Δ ρ variant.
- The *Impact of estimators*⁶ notebook compares natural estimators (if they exist) and the external estimators Δ and ρ .
- The *Performance profiles*⁷ notebook explains the computation of Figure 6.14.
- The *Examples with a chain and a Y-shape*⁸ notebook gives toy examples that illustrate the behavior of estimators. In particular, it explains how the example of Figure 6.7 was built.

Running Time. For our benchmark of Figure 6.14, we sort 10,000 random lists of size 1,000. We choose this size of list for two reasons. First, our framework addresses applications where the comparisons are expensive, for example because they are manually performed by human experts, so the typical use-case does not concern lists whose size is orders of magnitude higher. Secondly, since this benchmark consists in interrupting the algorithms at each step to check the current Spearman distance to the ground truth, it slows down the process; without this artificial additional delay, it is realistic to use the algorithms with larger lists.

The table below reports the running time of each algorithm, based on experiments conducted on a MinisForum UM870 Slim equipped with an AMD Ryzen 7 8745H processor (16 threads).

Algorithm	Time
Top-down merge	7 min 48 s
Binary insertion	6 min 30 s
Binary insertion ρ	27 min 02 s
Ford-Johnson ρ	32 min 25 s
Quicksort	7 min 45 s
ASort	8 min 07 s
ASort ρ	58 min 38 s
Bottom-up merge ρ	16 min 11 s
Multizip Y	16 min 03 s
Corsort- Δ ρ	90 min 29 s

⁵<https://emczg.github.io/corsort/notebooks/corsorts.html>

⁶https://emczg.github.io/corsort/notebooks/rho_delta.html

⁷<https://emczg.github.io/corsort/notebooks/profiles.html>

⁸https://emczg.github.io/corsort/notebooks/examples_with_a_chain_and_a_y_shape.html

Note that the reported times reflect a setting where the algorithm is explicitly paused at each step to record and analyze intermediate estimates. As such, these measurements provide a theoretical upper bound on computational cost and are primarily intended to illustrate the scale of our simulations. They do not correspond to realistic deployment scenarios, which were not the primary focus of our papers [CDM24; CDM25].

However, how can such tools be developed and deployed in real-life scenarios? The following section explores this question in the context of the 2025 Australian federal election.

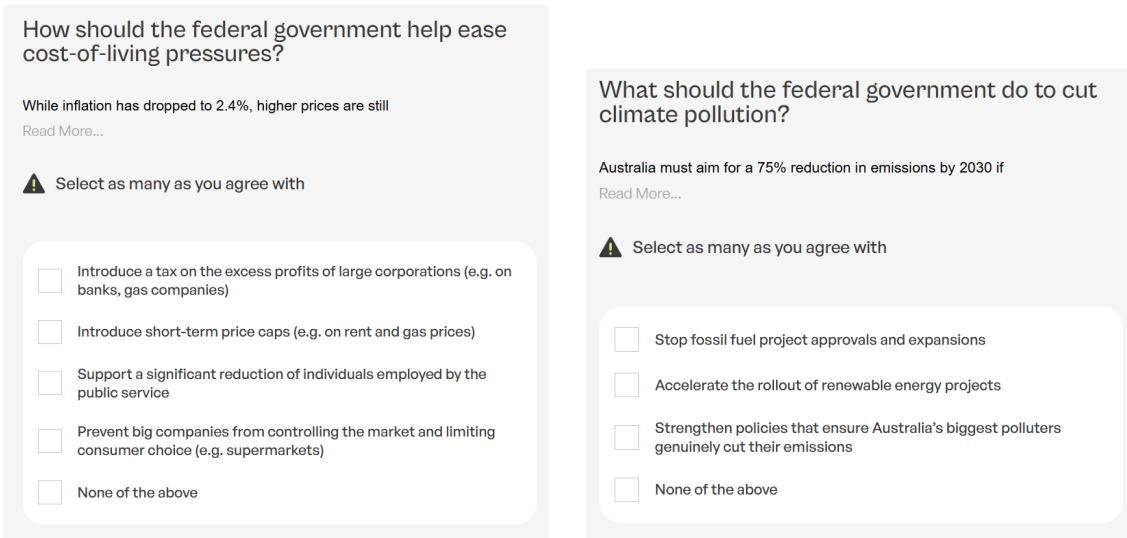
6.5 Australian Federal Election

As mentioned in the introduction, the Australian federal elections were one of our initial motivations for studying anytime sorting algorithms: voters are required to rank all candidates, which can be cognitively demanding. In such a context, an anytime algorithm appears particularly relevant. Now that we have developed algorithms capable of producing meaningful intermediate rankings at any point during execution, while maintaining low comparison and interruption costs, a natural question arises: how might such tools be applied in practice, specifically in an electoral setting? Before exploring this possibility, we first review the two main digital voting assistance tools that were available to Australian voters during the 2025 federal election.

6.5.1 Existing Tools

As noted in the introduction, *Build a Ballot* was one of the main digital platforms available to Australian voters ahead of the 2025 federal election. It not only offered a centralized database of candidate information, but also included an interactive tool designed to help users elicit their own political preferences. In this section, we take a closer look at this platform, along with another widely used system, *Vote Compass* [Vox]. Both are among the top search results for Australian voting assistance tools.

Build a Ballot presents themed lists of policy proposals (e.g., on energy, housing, or the environment) and allows users to indicate agreement with as many proposals as they wish within each theme (see Figure 6.15). At the end, the system produces a ranked list of candidates based on a computed percentage of compatibility, derived from overlap between the user's approved proposals and each candidate's stated positions. The current implementation of **Build a Ballot** introduces significant distortions in how voter preferences are inferred. For example, selecting all proposals

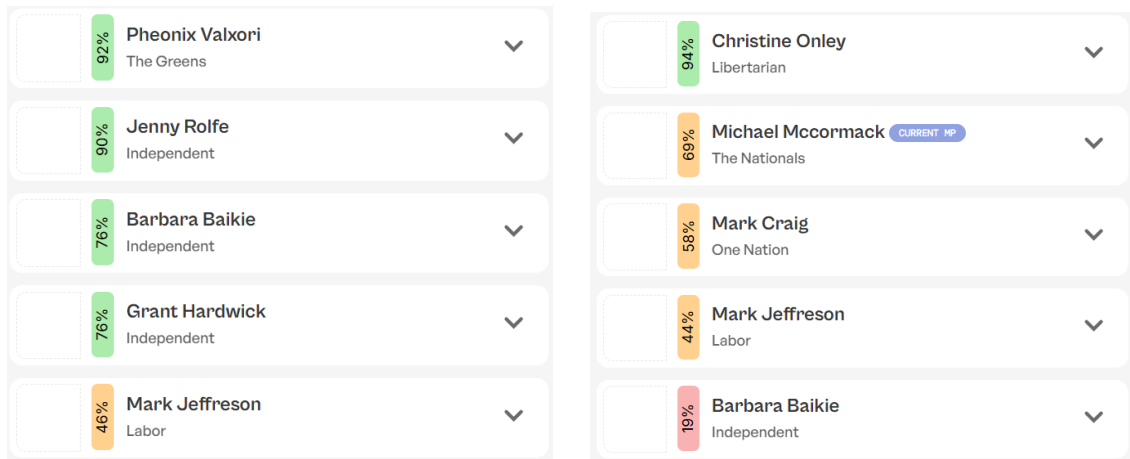


(a) Example interface showing a question and multiple proposals. (b) Same interface showing a different question.

Fig. 6.15: User interface of the Build a Ballot platform. The tool presents users with a series of policy questions, each offering a list of proposals associated with different candidates or parties. Users select the proposals they agree with, which the system then uses to compute compatibility scores.

typically results in a very high compatibility score with left-wing candidates, in particular, *The Greens*, as illustrated in Figure 6.16a. One explanation could be that the platform was developed by *Project Planet Inc.*, a climate-focused charity with strong environmental values, thus showing bias towards climate-change oriented propositions. On the contrary, approving no propositions results in a high compatibility score with the right-wing parties, as shown in Figure 6.16b. Thus, failing to approve even a single one proposition often penalizes left-wing candidates and advantages right-wing ones. This design choice not only skews compatibility scores when users do select most proposals, but also introduces a slightly structural advantage for *The Greens*: since users almost never leave all boxes unchecked, the party that authored or aligns with the largest share of proposals is unlikely to ever appear at the bottom of the rankings.

Vote Compass, a tool developed by researchers at *Vox Pop Labs*, by contrast, shows one proposal at a time and asks users to rate their agreement on a Likert scale ranging from “strongly disagree” to “strongly agree” with an additional “I don’t know” option (see Figure 6.17). At the end of the process, the algorithm places the user on a two-dimensional ideological map with some political parties, thus showing their proximity spatially. The main problem is it excludes most of political parties, which can leave users uncertain about how to translate the outcome into an actual voting decision.



(a) Top 5 Riverina candidates returned by Build A Ballot when all proposals are selected for each question. (b) Top 5 Riverina candidates returned by Build A Ballot when "None of the above" is selected for each question.

Fig. 6.16: Illustration of selection bias in Build a Ballot, based on test cases in the Riverina district. Selecting all proposals leads to strong alignment with left-leaning parties (notably *The Greens*), whereas consistently selecting "None of the above" favors right-leaning candidates.

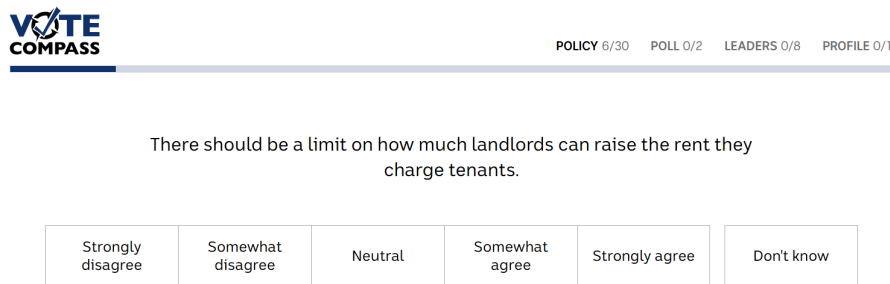


Fig. 6.17: Example interface from the Vote Compass platform. Users respond to a series of policy statements on a Likert scale, indicating their level of agreement or disagreement. The platform then computes the closest match with the positions of various political parties based on their responses.

Additionally, Vote Compass exhibits a similar bias as Build a Ballot, that is a directional bias in how user preferences are mapped: strongly agreeing with all proposals tends to place the user closer to left-leaning parties, while strongly disagreeing with all proposals favors alignment with right-leaning parties; choosing neutral answers across the board typically positions the user near the political center, as broadcast in Figure 6.18. Thus, it is legitimate to question whether some statements could be reworded to achieve greater ideological balance. For instance, the statement "There should be a limit on how much landlords can raise the rent they charge tenants" could have been reframed as "Landlords should be free to set rental prices without restriction", offering a reversed yet semantically parallel version. In contrast, for quantitatively structured prompts

such as "How much should the Australian government spend to subsidise the cost of electricity for consumers?", finding an equally neutral reformulation is more complex without altering the question's format. In this light, while Vote Compass does exhibit some degree of directional bias, it appears to stem more from practical design constraints than deliberate political intent. Moreover, the fact that neutral answers place users near the center suggests that the system maintains at least a baseline of interpretive balance.

The fact that both tools exhibit bias could be explained by a structural limitation rooted in political discourse rather than platform design. The perceived left-leaning bias may be attributed to an asymmetry in the articulation of party platforms: left-wing parties are generally more inclined to publish detailed, programmatic proposals, whereas right-wing parties tend to communicate broader ideological positions with fewer concrete policy commitments. For example, in the domain of education, the left-wing party *The Greens* outline 106 distinct and documented policy points⁹, while the right-wing party *The Nationals* list merely two objectives¹⁰, and the far-right party *One Nation* provides only a four-line statement with no actionable measures¹¹, accompanied with a 4min22s video¹² that is not directly related to the topic but that mentions the word "education" once. As a consequence, platforms that rely on matching users to policy content may unintentionally favor parties with more extensive and explicit agendas. This is not an intrinsic flaw of the platforms themselves, but rather a reflection of the uneven communicative strategies employed by political actors.

These limitations point to a deeper issue: as long as some candidates provide little or no detail about their policy positions, no system based solely on proposal matching can treat them equally. In practice, attempts to ensure fairness, such as balancing the number of proposals across the political spectrum, may end up concealing substantial asymmetries in how parties engage with voters. Equating a party that publishes a comprehensive program with one that offers only vague statements creates the illusion of parity where there is none.

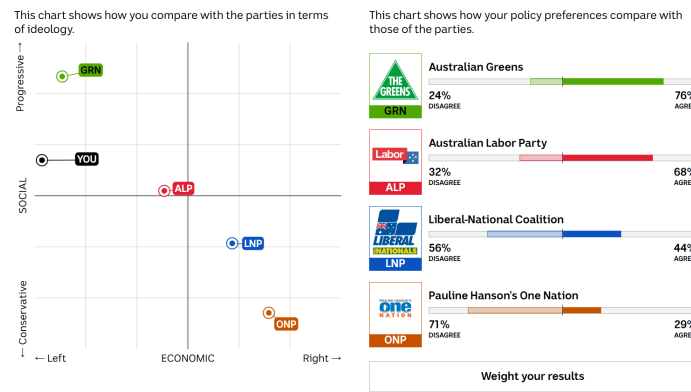
More fundamentally, both Build a Ballot and Vote Compass operate under the assumption that a small, fixed set of questions is sufficient to capture voter preferences. While this design is accessible, particularly for users who are unfamiliar with politics, it is better suited for offering a rough orientation on the ideological spectrum than for supporting the construction of a complete ranking of candidates. For voters who have already formed preliminary opinions, or who wish

⁹<https://web.archive.org/web/20250728142213/https://greens.org.au/nsw/policies/education>

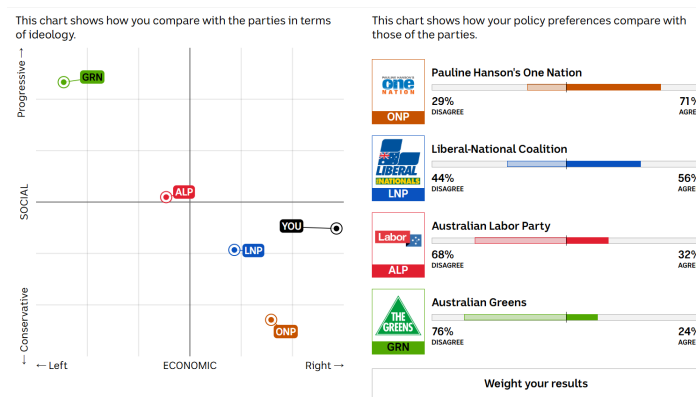
¹⁰<https://web.archive.org/web/20250728142013/https://www.nationalsplan.org.au/>

¹¹<https://web.archive.org/web/20250728141552/https://www.onenation.org.au/education>

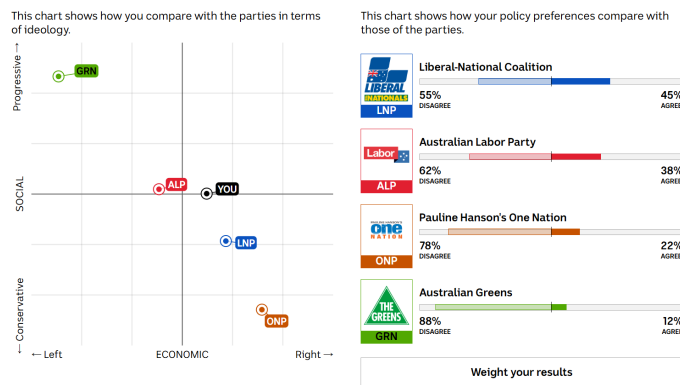
¹²<https://web.archive.org/web/20250728142546/https://www.youtube.com/watch?v=m4-Q2yDQHPA>, <https://www.youtube.com/watch?v=m4-Q2yDQHPA>



(a) Strongly agree with all proposals.



(b) Strongly disagree with all proposals.



(c) Neutral on all proposals.

Fig. 6.18: Positional bias in *Vote Compass* based on response patterns. Selecting strong agreement with all policy statements tends to align users with left-leaning parties, strong disagreement aligns them with the political right, and neutral responses place users near the center.

to engage more deeply, such formats may prove overly reductive. This limitation motivates the exploration of alternative approaches based on anytime sorting algorithms.

6.5.2 Anytime Sorting Tool

Building on our work in anytime sorting, we can envision a digital ballot assistant that allows voters to construct their preferences incrementally through pairwise comparisons. As the voter interacts with the interface, an anytime algorithm such as Corsort would maintain a consistent and evolving estimate of their full ranking. At any point, the voter could stop, review the current estimate, and optionally choose it as their final preference list. While this approach requires voters to invest more effort in seeking information either before or during the sorting process, it enables more informed decision-making, and still allows for a full candidate ranking to be produced at any point, should the voter decide to stop.

However, as highlighted in the previous section, implementing such a system in practice faces significant challenges, foremost among them ensuring voters have sufficient information to make meaningful comparisons. A centralized candidate-oriented platform could address this by:

- Including direct links to candidates' official websites when available. For candidates without an online presence, the platform might highlight this fact. At the same time, it is important to recognize that smaller parties may lack the resources to provide a well-documented website, highlighting a potential role for public authorities to ensure the access to basic web infrastructure and hosting services.
- Providing candidate biographies authored by the candidates themselves, including key policy positions, directly in the platform, for voters who wish to invest only a limited amount of time in political investigation. In cases where candidates decline to supply this information, the platform should clearly indicate this decision.
- Displaying both candidate biographies and website links alongside each pairwise comparison, so that voters can access relevant information as needed. This reflects the time cost associated with making informed comparisons, and reinforces the distinction between comparison complexity and overall computational complexity.

It is worth noting that the first point is already implemented in the Build a Ballot platform. This makes it feasible to envision a preliminary version of an anytime sorting assistant that leverages this existing infrastructure. However, the second point is not currently implemented and would require substantial effort to develop.

Now, to foster voters trust, three key conditions must be met.

- The underlying algorithmic logic should be well-documented and accessible.
- The interface should allow voters to inspect previous comparisons made and all estimates produced so far, so that they can verify the algorithm's behavior and the consistency of the estimates.
- The order in which comparisons are proposed must not introduce bias. To this end, early comparisons, particularly the first $m/2$, should be sampled randomly to avoid favoring specific candidates. This concern is not merely theoretical: before 1984, ballots listed candidates alphabetically. Therefore, names starting with *A* were found to receive undue advantage. A notable example was the 1937 Senate election, in which the Labor candidate group in New South Wales consisted of Amour, Ashley, Armstrong and Arthur, all of whom were elected. This phenomenon is known as "Donkey voting".

Beyond these implementation concerns, future work could explore extensions that accommodate cases in which voters prefer not to compare certain candidates. This may occur when voters are unfamiliar with one or both candidates in a given pair, or when they wish to express indifference. For instance, a user might choose to engage only with comparisons involving well-known candidates and stop once the remaining options fall outside their knowledge or interest.

6.6 Conclusion

Anytime Sorting. We have formalized the problem of anytime sorting and introduced novel approaches to address it. Our proposed estimators allow to transform any comparison-based sorting algorithm into an anytime algorithm. We have introduced the innovative algorithms of multizip sort and Corsort. Through extensive simulations, their strong performance has been demonstrated, showcasing their efficiency in terms of termination time and the quality of intermediate estimates. Our work advances the understanding and effectiveness of anytime sorting algorithms. In particular, despite its lack of proven upper bound and global complexity, we believe that Corsort *de facto* replaces ASort as a benchmark for empirical evaluation of future anytime sorting algorithms.

The relative novelty of our approach opens up several avenues for future research in the domain of anytime sorting. Exploring different scoring functions could potentially enhance Corsort even further, and also improve the performance profiles of classical sorting algorithms like Ford-Johnson. Formally proving that Corsort has an $\mathcal{O}(m \ln(m))$ comparison complexity poses another challenge.

Additionally, narrowing the gap between existing theoretical bounds and empirical observations is a promising direction for future research.

Australian Elections. The integration of anytime sorting algorithms into electoral technology, if done with precautions and transparency, could offer a promising path toward more flexible and voter-centered decision-making tools, capable of accommodating varying levels of voter engagement and information access. These tools open up several research directions. Empirical studies are needed to understand how voters engage with such systems: how many comparisons are they willing to perform? How reliable or stable do their preferences appear across sessions? From an algorithmic standpoint, extending anytime sorting methods to accommodate abstentions, indifference, or uncertain responses would enhance their realism and inclusivity in practical voting scenarios.

Conclusion

7

Summary of the Contributions

In this thesis, I have presented two methods for calculating the probability of the Condorcet paradox under the assumption of independent voters, both based on analytic combinatorics.

Since this thesis is also intended for readers who may be unfamiliar with analytic combinatorics, in Chapter 2 I introduced the main ideas through a guiding example based on fragmented permutations. I aimed to make this chapter accessible to readers not necessarily interested in the rest of the thesis, providing detailed explanations of both the symbolic method and the analytic extraction process. I hope that, taken on its own, this chapter has pedagogical value beyond the scope of the thesis.

The first method for studying the Condorcet paradox, developed in Chapter 3, focused on the Impartial Culture model and relied on a graph-theoretic encoding of voting profiles. By representing a voting profile as a bipartite graph with degree constraints, we were able to transform the original voting problem into a question of graph enumeration.

While the results were intricate from a voting-theoretic perspective, this approach yielded interesting combinatorial insights. In particular, we computed the generating function of 0-1 balanced matrices using the bipartite framework. We then extended this enumeration to general (not necessarily bipartite) graphs, which notably led to an asymptotic formula for κ -regular graphs (graphs in which all vertices have degree κ).

Due to the complexity of the results and the rigidity of the graph-based approach—it could only be applied to Impartial Culture and could not be generalized to the GIC model—we developed a second method for computing the probability of the Condorcet paradox. This approach, presented in Chapters 4 and 5, directly encoded pairwise comparisons, yielding a generating function that is suitable for asymptotic analysis. Using the saddle point method, we derived asymptotic results under the GIC model. This method not only recovered known limiting probabilities but also provided new results on the rates of convergence. In particular, we applied the theoretical findings to two cultures: Impartial Culture and Mallows.

For Impartial Culture, the probability that there exists a Condorcet winner is of the form

$$\mathbb{P}(m \text{ is CW}) = a_0 + a_{1,n}n^{-1/2} + \mathcal{O}(n^{-1}),$$

where a_0 and $a_{1,n}$ are given by

$$a_0 = \frac{1}{\sqrt{(2\pi)^{m-1} \det(\mathcal{H}_K(\mathbf{0}))}} \int_{(0,+\infty)^{m-1}} e^{-\mathbf{u}\mathcal{H}_K(\mathbf{0})^{-1}\mathbf{u}/2} d\mathbf{u},$$

$$a_{1,n} = \begin{cases} 0 & \text{if } n \text{ is odd,} \\ -\frac{6}{m+1} \frac{1}{\sqrt{(2\pi)^{m-1} \det(\mathcal{H}_K(\mathbf{0}))}} \int_{(0,+\infty)^{m-1}} \mathbf{1}^T \mathbf{u} e^{-\mathbf{u}^T \mathcal{H}_K(\mathbf{0})^{-1} \mathbf{u}/2} d\mathbf{u}. & \text{if } n \text{ is even,} \end{cases}$$

and $\mathcal{H}_K(\mathbf{0})$, its inverse and determinant are given in Equation (4.7).

In the case of the Mallows model, we studied two main variants: one in which the candidate of interest occupies the first position in the reference ranking, and another where it appears last. The latter variant was examined in two settings, with three and four candidates, corresponding respectively to the models $\mathcal{M}_{3 \text{ last}}$ and $\mathcal{M}_{4 \text{ last}}$. For both cases, since the reference ranking can be interpreted as representing the underlying ground truth, we showed that the probability of the *a priori* undesirable event in which the candidate ranked last nevertheless emerges as the Condorcet winner tends exponentially fast to 0. Conversely, under the model $\mathcal{M}_{3 \text{ first}}$, where candidate 3 occupies the first position in the reference ranking, we showed that the probability that candidate 3 is the Condorcet winner tends exponentially fast to 1.

Beyond these results, some of the multivariate analytic combinatorics theorems developed in Chapter 5 were of independent interest. In particular, we consider the definitions of rank and periodicity for multivariate generating functions as some of our most significant contributions, along with Lemma 15, which extends the Daffodil Lemma [FS09, Lemma IV.1] to the multivariate case.

Finally, Chapter 6 departed from the probabilistic analysis and addressed a practical question: how can individuals elicit their own preferences efficiently? To tackle this, we proposed efficient anytime sorting algorithms for constructing preferences, introducing estimators that transform any comparison-based sorting algorithm into an anytime algorithm, as well as the new algorithms Multizip Sort and Corsort. Extensive simulations demonstrated their strong performance in terms of both termination time and the quality of intermediate estimates. In particular, although Corsort lacks a formal upper bound and complexity analysis, it effectively replaces ASort as a benchmark for empirical evaluation of future anytime sorting algorithms.

This chapter also discussed potential applications to real-world voting contexts, illustrated by the 2025 federal elections in Australia. In particular, it opened a discussion around two online voting assistants designed to help voters elicit their preferences, highlighting the practical relevance of anytime sorting algorithms.

Overall, this thesis can be viewed as evidence that bridges naturally emerge between different areas of mathematics and computer science. Each chapter, taken independently, either led to contributions of interest in other domains, such as the analytic combinatorics results in Chapters 3 to 5, or opened a more exploratory discussion, as the one in Chapter 6, which raised practical questions related to a real-life voting scenario.

Future Work

Several directions for future research arise from the work presented in this thesis.

Condorcet Paradox and Saddle Point Methods. A natural direction is to investigate cases where the saddle point has null or infinite coordinates, which may occur in non-generic cultures. It would also be valuable to perform a finer analysis of the complexity of the algorithm presented in Section 4.8 for handling supercritical coordinates, particularly to identify conditions under which it runs in polynomial time. Another avenue is to examine the limiting behavior of the probability as the number of candidates m tends to infinity. Finally, extending this method to other events, such as the transitivity of the majority relation, other forms of monotonicity failures, or manipulability (i.e., susceptibility to strategic voting), could broaden its applicability.

Analytic Combinatorics and Coefficient Extraction. Future work could explore more general frameworks for coefficient extraction, beyond the specific setting

$$[z^{\lambda n}] A(z)B(z)^n,$$

for instance when the polynomial $B(z)$ appears through a more intricate functional form. Another direction is to compute additional terms in the asymptotic expansions, either manually or with computer assistance, potentially leading to symbolic algorithms capable of performing these computations automatically.

Anytime Sorting and Preference Elicitation. The novelty of our approach opens several avenues in the domain of anytime sorting. Exploring different scoring functions could further enhance Corsort and improve the performance of classical sorting algorithms, such as Ford-Johnson. Formally proving that Corsort has $\mathcal{O}(m \ln(m))$ comparison complexity is another challenge. Narrowing the gap between existing theoretical bounds and empirical observations also remains a relevant task. From an application perspective, empirical studies are needed to understand how voters engage with these systems, including the number of comparisons they are willing to perform and the reliability of their preferences across sessions. Algorithmically, extending anytime sorting methods to accommodate abstentions, indifference, or uncertain responses would increase their realism and inclusivity in practical voting scenarios.

Bibliography of Chapters 1 and 2

- [Arr50] Kenneth J Arrow. “A difficulty in the concept of social welfare”. In: *Journal of political economy* 58.4 (1950), pp. 328–346 (cit. on p. 2).
- [Bar25] Pierre Bardier. “The probability of satisfying axioms: a non-binary perspective on economic design”. In: *arXiv preprint arXiv:2502.13850* (2025) (cit. on p. 3).
- [Con85] Marie Jean Antoine Nicolas de Caritat, marquis de Condorcet. *Essai sur l’application de l’analyse à la probabilité des décisions rendues à la pluralité des voix*. Imprimerie royale, 1785 (cit. on pp. 2, 39, 99).
- [FS09] Philippe Flajolet and Robert Sedgewick. *Analytic Combinatorics*. Cambridge University Press, 2009 (cit. on pp. 4, 11, 17, 18, 22, 27, 35, 37, 42, 46, 69, 71, 82, 91, 114, 120–122, 124, 126, 128, 136, 139, 188).
- [Geh06] William Gehrlein. *Condorcet’s Paradox*. Springer, 2006 (cit. on pp. 3, 81, 83, 84).
- [Gib73] Allan Gibbard. “Manipulation of voting schemes: a general result”. In: *Econometrica: journal of the Econometric Society* (1973), pp. 587–601 (cit. on p. 2).
- [HR17] Godfrey Harold Hardy and Srinivasa Ramanujan. “Asymptotic Formulae for the Distribution of Integers of Various Types”. In: *Proceedings of the London Mathematical Society* 2.1 (1917), pp. 112–132 (cit. on p. 4).
- [Hay56] Walter K Hayman. “A generalisation of Stirling’s formula.” In: (1956) (cit. on p. 4).
- [Ken38] Maurice G Kendall. “A new measure of rank correlation”. In: *Biometrika* 30.1-2 (1938), pp. 81–93 (cit. on pp. 7, 87).
- [Knu99] Donald Knuth. “The art of computer programming”. In: *Sorting and searching* 3.426-458 (1999), p. 4 (cit. on p. 4).
- [KR05] Mukkai Krishnamoorthy and Madabhushi Raghavachari. *Condorcet Winner Probabilities - A Statistical Perspective*. 2005 (cit. on pp. 4, 81–84, 102, 104).
- [Mal57] Colin L Mallows. “Non-null ranking models. I”. In: *Biometrika* 44.1/2 (1957), pp. 114–130 (cit. on pp. 7, 84, 87).
- [NW68] Richard Niemi and Herbert Weisberg. “A mathematical solution for the probability of the paradox of voting”. In: *Behavioral Science* 13.4 (1968), pp. 317–323 (cit. on pp. 4, 6, 81–84, 102, 104).

- [Sau22] Lisa Sauermann. “On the probability of a Condorcet winner among a large number of alternatives”. In: *arXiv preprint arXiv:2203.13713* (2022) (cit. on pp. 4, 83).
- [SS04] Murat R Sertel and M Remzi Sanver. “Strong equilibrium outcomes of voting games are the generalized Condorcet winners”. In: *Social Choice and Welfare* 22.2 (2004), pp. 331–347 (cit. on pp. 8, 87).
- [Ten15] Gérald Tenenbaum. *Introduction to analytic and probabilistic number theory*. Vol. 163. American Mathematical Soc., 2015 (cit. on p. 23).
- [You74] H Peyton Young. “An axiomatization of Borda’s rule”. In: *Journal of economic theory* 9.1 (1974), pp. 43–52 (cit. on p. 3).

Bibliography of Chapter 3

- [AZ06] Moa Apagodu and Doron Zeilberger. “Multi-variable Zeilberger and Almkvist–Zeilberger algorithms and the sharpening of Wilf–Zeilberger theory”. In: *Advances in Applied Mathematics* 37.2 (2006), pp. 139–152 (cit. on p. 62).
- [Bel19] Dževad Belkić. “All the trinomial roots, their powers and logarithms from the Lambert series, Bell polynomials and Fox–Wright function: illustration for genome multiplicity in survival of irradiated cells”. In: *Journal of Mathematical Chemistry* 57.1 (2019), pp. 59–106 (cit. on p. 60).
- [BC78] Edward A. Bender and E. Rodney Canfield. “The asymptotic number of labeled graphs with given degree sequences”. In: *Journal of Combinatorial Theory, Series A* 24.3 (1978), pp. 296–307 (cit. on pp. 40, 43).
- [BLL97] François Bergeron, Gilbert Labelle, and Pierre Leroux. *Combinatorial Species and Tree-like Structures*. Cambridge University Press, 1997 (cit. on pp. 42, 48).
- [CP23] Emma Caizergues and Elie de Panafieu. “Exact enumeration of graphs and bipartite graphs with degree constraints”. In: *European Conference on Combinatorics, Graph Theory and Applications*. 12. 2023 (cit. on p. 42).
- [CPR07] Gab-Byung Chae, Edgar M Palmer, and Robert W Robinson. “Counting labeled general cubic graphs”. In: *Discrete mathematics* 307.23 (2007), pp. 2979–2992 (cit. on p. 41).
- [CL99] William YC Chen and James D Louck. “Enumeration of Cubic Graphs by Inclusion–Exclusion”. In: *Journal of Combinatorial Theory, Series A* 86.1 (1999), pp. 151–157 (cit. on p. 42).
- [CM24] Frédéric Chyzak and Marni Mishna. “Differential equations satisfied by generating functions of 5-, 6-, and 7-regular labelled graphs: a reduction-based approach”. In: *arxiv* (2024) (cit. on p. 42).
- [Con85] Marie Jean Antoine Nicolas de Caritat, marquis de Condorcet. *Essai sur l’application de l’analyse à la probabilité des décisions rendues à la pluralité des voix*. Imprimerie royale, 1785 (cit. on pp. 2, 39, 99).

- [Dou+24] Robert Dougherty-Bliss, Christoph Koutschan, Natalya Ter-Saakov, and Doron Zeilberger. “The (Symbolic and Numeric) Computational Challenges of Counting 0-1 balanced matrices”. In: *arXiv preprint arXiv:2410.07435* (2024) (cit. on p. 62).
- [FS09] Philippe Flajolet and Robert Sedgewick. *Analytic Combinatorics*. Cambridge University Press, 2009 (cit. on pp. 4, 11, 17, 18, 22, 27, 35, 37, 42, 46, 69, 71, 82, 91, 114, 120–122, 124, 126, 128, 136, 139, 188).
- [Geh83] William V Gehrlein. “Condorcet’s paradox”. In: *Theory and decision* 15.2 (1983), pp. 161–197 (cit. on pp. 39, 83, 84).
- [Ges90] Ira M. Gessel. “Symmetric functions and P-recursiveness”. In: *Journal of Combinatorial Theory, Series A* 53 (1990), pp. 257–285 (cit. on p. 42).
- [GJR83] I. P. Goulden, D. M. Jackson, and J. W. Reilly. “The Hammond series of a symmetric function and its application to P-recursiveness”. In: *SIAM Journal on Algebraic Discrete Methods* 4.2 (1983), pp. 179–193 (cit. on p. 41).
- [GJ86] Ian P Goulden and David Martin Jackson. “Labelled graphs with small vertex degrees and P-recursiveness”. In: *SIAM Journal on Algebraic Discrete Methods* 7.1 (1986), pp. 60–66 (cit. on p. 41).
- [Gro92] Harald Gropp. “Enumeration of regular graphs 100 years ago”. In: *Discrete mathematics* 101.1-3 (1992), pp. 73–85 (cit. on p. 40).
- [Joy81] André Joyal. “Une théorie combinatoire des séries formelles”. In: *Advances in mathematics* 42.1 (1981), pp. 1–82 (cit. on p. 48).
- [Kau23] Manuel Kauers. *D-finite Functions*. Springer, 2023 (cit. on p. 62).
- [Kou10] Christoph Koutschan. “A fast approach to creative telescoping”. In: *Mathematics in Computer Science* 4.2 (2010), pp. 259–266 (cit. on p. 63).
- [Kou09] Christoph Koutschan. *Advanced applications of the holonomic systems approach*. na, 2009 (cit. on p. 63).
- [Kou14] Christoph Koutschan. “Holonomic functions in Mathematica”. In: *ACM Communications in Computer Algebra* 47.3/4 (2014), pp. 179–182 (cit. on p. 63).
- [Lip89] Leonard Lipshitz. “D-finite power series”. In: *Journal of algebra* 122.2 (1989), pp. 353–373 (cit. on pp. 40, 51, 58).
- [McK83] Brendan D McKay. “Applications of a technique for labelled enumeration”. In: *Congressus Numerantium* 40 (1983), pp. 207–221 (cit. on p. 43).
- [Mis05] Marni Mishna. “Automatic enumeration of regular objects”. In: *arXiv preprint math/0507249* (2005) (cit. on p. 42).

- [Pan18] Élie de Panafieu. “Analytic combinatorics of connected graphs”. In: *Random Structures & Algorithms* (2018) (cit. on p. 67).
- [Pan24] Élie de Panafieu. “Asymptotic expansion of regular and connected regular graphs”. In: *arxiv* (2024) (cit. on pp. 43, 79).
- [Pól37] G Pólya. “Kombinatorische Anzahlbestimmungen für Gruppen, Graphen und chemische Verbindungen”. In: *Acta Mathematica* 68.1 (1937), pp. 145–254 (cit. on p. 41).
- [Rea59a] Ronald C Read. “Some enumeration problems in graph theory”. PhD thesis. University of London (University College of the West Indies), 1959 (cit. on p. 41).
- [Rea59b] Ronald C Read. “The enumeration of locally restricted graphs (I)”. In: *Journal of the London Mathematical Society* 1.4 (1959), pp. 417–436 (cit. on p. 41).
- [Rea60] Ronald C Read. “The enumeration of locally restricted graphs (II)”. In: *Journal of the London Mathematical Society* 1.3 (1960), pp. 344–351 (cit. on pp. 41, 42).
- [RW80] Ronald C Read and Nicholas C Wormald. “The number of labelled 4-regular graphs”. In: *Journal of Graph Theory* 4.2 (1980), pp. 203–212 (cit. on p. 41).
- [Wor18] Nicholas Wormald. “Asymptotic enumeration of graphs with given degree sequence”. In: *Proceedings of the International Congress of Mathematicians: Rio de Janeiro 2018*. World Scientific, 2018, pp. 3245–3264 (cit. on p. 40).
- [Zei90] Doron Zeilberger. “A holonomic systems approach to special functions identities”. In: *Journal of computational and applied mathematics* 32.3 (1990), pp. 321–368 (cit. on p. 62).

Bibliography of Chapter 4

- [APX12] Hossein Azari, David Parks, and Lirong Xia. “Random utility theory for social choice”. In: *Advances in Neural Information Processing Systems 25* (2012) (cit. on p. 83).
- [Bar94] Alexander I Barvinok. “A polynomial time algorithm for counting integral points in polyhedra when the dimension is fixed”. In: *Mathematics of Operations Research* 19.4 (1994), pp. 769–779 (cit. on p. 84).
- [Bel81] Colin Bell. “A random voting graph almost surely has a Hamiltonian cycle when the number of alternatives is large”. In: *Econometrica* (1981), pp. 1597–1603 (cit. on p. 84).
- [Bel78] Colin Bell. “What happens when majority rule breaks down?” In: *Public Choice* 33.2 (1978), pp. 121–126 (cit. on p. 84).
- [BL92] Sven Berg and Dominique Lepelley. “Note sur le calcul de la probabilité des paradoxes du vote”. In: *Mathématiques et Sciences humaines* 120 (1992), pp. 33–48 (cit. on p. 84).
- [Bra+16] Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel D Procaccia. *Handbook of computational social choice*. Cambridge University Press, 2016 (cit. on p. 87).
- [Cai+25] Emma Caizergues, François Durand, Marc Noy, Élie de Panafieu, and Vlady Rav-elomanana. “Probability of a Condorcet Winner for Large Electorates: An Analytic Combinatorics Approach”. In: *arXiv preprint arXiv:2505.06028* (2025) (cit. on p. 82).
- [Con85] Marie Jean Antoine Nicolas de Caritat, marquis de Condorcet. *Essai sur l’application de l’analyse à la probabilité des décisions rendues à la pluralité des voix*. Imprimerie royale, 1785 (cit. on pp. 2, 39, 99).
- [De 14] Nicolaas Govert De Bruijn. *Asymptotic methods in analysis*. Courier Corporation, 2014 (cit. on p. 121).
- [DF29] Bruno De Finetti. “Funzione caratteristica di un fenomeno aleatorio”. In: *Atti del Congresso Internazionale dei Matematici: Bologna del 3 al 10 de settembre di 1928*. 1929, pp. 179–190 (cit. on p. 85).

- [DPR04] Jean-Paul Doignon, Aleksandar Pekeč, and Michel Regenwetter. “The repeated insertion model for rankings: Missing link between two subset choice models”. In: *Psychometrika* 69.1 (2004), pp. 33–54 (cit. on p. 100).
- [Dur25] François Durand. *Actinvoting: Analytic Combinatorics Tools in Voting*. <https://github.com/francois-durand/actinvoting>. 2025 (cit. on pp. 83, 100, 106).
- [Fis73] Peter Fishburn. “A proof of May’s Theorem $P(m, 4) = 2 P(m, 3)$ ”. In: *Behavioral Science* 18.3 (1973), p. 212 (cit. on p. 83).
- [FS09] Philippe Flajolet and Robert Sedgewick. *Analytic Combinatorics*. Cambridge University Press, 2009 (cit. on pp. 4, 11, 17, 18, 22, 27, 35, 37, 42, 46, 69, 71, 82, 91, 114, 120–122, 124, 126, 128, 136, 139, 188).
- [GK68] Mark Garman and Morton Kamien. “The paradox of voting: Probability calculations”. In: *Behavioral Science* 13 (1968), pp. 306–316 (cit. on p. 83).
- [Geh06] William Gehrlein. *Condorcet’s Paradox*. Springer, 2006 (cit. on pp. 3, 81, 83, 84).
- [Geh02] William Gehrlein. “Obtaining representations for probabilities of voting outcomes with effectively unlimited precision integer arithmetic”. In: *Social Choice and Welfare* 19 (3 2002), pp. 503–512 (cit. on p. 84).
- [GF76a] William Gehrlein and Peter Fishburn. “Condorcet’s paradox and anonymous preference profiles”. In: *Public Choice* (1976), pp. 1–18 (cit. on p. 84).
- [GF78] William Gehrlein and Peter Fishburn. “Probabilities of election outcomes for large electorates”. In: *Journal of Economic Theory* 19.1 (1978), pp. 38–49 (cit. on p. 84).
- [GF76b] William Gehrlein and Peter Fishburn. “The probability of the paradox of voting: A computable solution”. In: *Journal of Economic Theory* 13.1 (1976), pp. 14–25 (cit. on p. 83).
- [Geh83] William V Gehrlein. “Condorcet’s paradox”. In: *Theory and decision* 15.2 (1983), pp. 161–197 (cit. on pp. 39, 83, 84).
- [Gil78] Raphael Gillett. “A recursion relation for the probability of the paradox of voting”. In: *Journal of Economic Theory* 18.2 (1978), pp. 318–327 (cit. on p. 83).
- [Gui52] Georges-Théodule Guilbaud. “Les théories de l’intérêt général et le problème logique de l’agrégation”. In: *Économie appliquée* 5 (1952), pp. 501–584 (cit. on p. 83).
- [HS55] Edwin Hewitt and Leonard J Savage. “Symmetric measures on Cartesian products”. In: *Transactions of the American Mathematical Society* 80.2 (1955), pp. 470–501 (cit. on p. 85).

- [HC00] HC Huang and Vincent CH Chua. “Analytical representation of probabilities under the IAC condition”. In: *Social Choice and Welfare* 17.1 (2000), pp. 143–155 (cit. on p. 84).
- [Kel74] Jerry Kelly. “Voting anomalies, the number of voters, and the number of alternatives”. In: *Econometrica* (1974), pp. 239–251 (cit. on p. 84).
- [Ken38] Maurice G Kendall. “A new measure of rank correlation”. In: *Biometrika* 30.1-2 (1938), pp. 81–93 (cit. on pp. 7, 87).
- [KR05] Mukkai Krishnamoorthy and Madabhushi Raghavachari. *Condorcet Winner Probabilities - A Statistical Perspective*. 2005 (cit. on pp. 4, 81–84, 102, 104).
- [Lep89] Dominique Lepelley. “Contribution a l’analyse des procedures de decision collective”. PhD thesis. Caen, 1989 (cit. on p. 84).
- [LLS08] Dominique Lepelley, Ahmed Louichi, and Hatem Smaoui. “On Ehrhart polynomials and probability calculations in voting theory”. In: *Social Choice and Welfare* 30.3 (2008), pp. 363–383 (cit. on p. 84).
- [LB14] Tyler Lu and Craig Boutilier. “Effective sampling and learning for mallows models with pairwise-preference data.” In: *J. Mach. Learn. Res.* 15.1 (2014), pp. 3783–3829 (cit. on p. 100).
- [Luc59] Robert Duncan Luce. *Individual choice behavior: A theoretical analysis*. Wiley, 1959 (cit. on p. 83).
- [MB02] Hans Maassen and Thom Bezembinder. “Generating random weak orders and the probability of a Condorcet winner.” In: *Social Choice and Welfare* 19.3 (2002), pp. 517–532 (cit. on p. 84).
- [Mal57] Colin L Mallows. “Non-null ranking models. I”. In: *Biometrika* 44.1/2 (1957), pp. 114–130 (cit. on pp. 7, 84, 87).
- [May71] Robert May. “Some mathematical remarks on the paradox of voting”. In: *Behavioral Science* 16.2 (1971), pp. 143–151 (cit. on pp. 83, 84, 104).
- [NW68] Richard Niemi and Herbert Weisberg. “A mathematical solution for the probability of the paradox of voting”. In: *Behavioral Science* 13.4 (1968), pp. 317–323 (cit. on pp. 4, 6, 81–84, 102, 104).
- [PW13] Robert Pemantle and Marc C. Wilson. *Analytic Combinatorics in Several Variables*. Cambridge University Press, 2013 (cit. on pp. 114, 120, 121, 136, 139, 143).
- [Pla75] Robin Plackett. “The analysis of permutations”. In: *Journal of the Royal Statistical Society Series C: Applied Statistics* 24.2 (1975), pp. 193–202 (cit. on p. 83).

- [Rub54] Harold Ruben. “On the Moments of Order Statistics in Samples from Normal Populations”. In: *Biometrika* 41.1/2 (June 1954), pp. 200–227 (cit. on p. 104).
- [Sau22] Lisa Sauermann. “On the probability of a Condorcet winner among a large number of alternatives”. In: *arXiv preprint arXiv:2203.13713* (2022) (cit. on pp. 4, 83).
- [SS04] Murat R Sertel and M Remzi Sanver. “Strong equilibrium outcomes of voting games are the generalized Condorcet winners”. In: *Social Choice and Welfare* 22.2 (2004), pp. 331–347 (cit. on pp. 8, 87).
- [Thu27] Louis Thurstone. “A law of comparative judgment”. In: *Psychological review* 34 (1927) (cit. on p. 83).
- [WS67] Oliver Williamson and Thomas Sargent. “Social choice: A probabilistic approach”. In: *The Economic Journal* 77.308 (1967), pp. 797–813 (cit. on pp. 83, 84).
- [WP07] Mark C Wilson and Geoffrey Pritchard. “Probability calculations under the IAC hypothesis”. In: *Mathematical Social Sciences* 54.3 (2007), pp. 244–256 (cit. on p. 84).

Bibliography of Chapter 5

- [Ban+01] Cyril Banderier, Philippe Flajolet, Gilles Schaeffer, and Michele Soria. “Random maps, coalescing saddles, singularity analysis, and Airy phenomena”. In: *Random Structures & Algorithms* 19.3-4 (2001), pp. 194–246 (cit. on p. 121).
- [DZ09] Amir Dembo and Ofer Zeitouni. *Large deviations techniques and applications*. Vol. 38. Springer Science & Business Media, 2009 (cit. on p. 126).
- [Duc+04] Philippe Duchon, Philippe Flajolet, Guy Louchard, and Gilles Schaeffer. “Boltzmann Samplers For The Random Generation Of Combinatorial Structures”. In: *Combinatorics, Probability and Computing* 13 (2004), p. 2004 (cit. on p. 126).
- [FSZ91] Philippe Flajolet, Bruno Salvy, and Paul Zimmermann. “Automatic average-case analysis of algorithms”. In: *Theoretical Computer Science* 79.1 (1991), pp. 37–109 (cit. on p. 127).
- [FS09] Philippe Flajolet and Robert Sedgewick. *Analytic Combinatorics*. Cambridge University Press, 2009 (cit. on pp. 4, 11, 17, 18, 22, 27, 35, 37, 42, 46, 69, 71, 82, 91, 114, 120–122, 124, 126, 128, 136, 139, 188).
- [Mel21] Stephen Melczer. *An Invitation to Analytic Combinatorics*. Springer, 2021 (cit. on pp. 124, 128, 129).
- [PW13] Robert Pemantle and Marc C. Wilson. *Analytic Combinatorics in Several Variables*. Cambridge University Press, 2013 (cit. on pp. 114, 120, 121, 136, 139, 143).
- [Ten15] Gérald Tenenbaum. *Introduction to analytic and probabilistic number theory*. Vol. 163. American Mathematical Soc., 2015 (cit. on p. 23).

Bibliography of Chapter 6

- [Ale+14] S. P. A. Alewijnse, T. M. Bagautdinov, M. de Berg, et al. “Progressive Geometric Algorithms”. In: *Proceedings of the Thirtieth Annual Symposium on Computational Geometry*. 2014, pp. 50–59 (cit. on p. 153).
- [BW91] Graham Brightwell and Peter Winkler. “Counting linear extensions is #P-complete”. In: *Proceedings of the Twenty-Third ACM Symposium on Theory of Computing - STOC '91*. 1991, pp. 175–181 (cit. on p. 161).
- [Bui] Build a Ballot. *Candidate Finder*. URL: <https://www.buildaballot.org.au/research/candidate-finder> (cit. on pp. 149, 151).
- [CDM24] Emma Caizergues, François Durand, and Fabien Mathieu. “Anytime Sorting Algorithms”. In: *Proceedings of the 33rd International Joint Conference on Artificial Intelligence, IJCAI-24*. 2024 (cit. on pp. 154, 155, 161, 179).
- [CDM23a] Emma Caizergues, François Durand, and Fabien Mathieu. “L’abus de comparaisons est mauvais pour la santé.” In: *JIAF*. 2023, pp. 159–168 (cit. on p. 154).
- [CDM25] Emma Caizergues, François Durand, and Fabien Mathieu. “Sortus Interruptus: Trier des phoques en $O(n \log(n))$ ”. In: *ALGOTEL 2025–27èmes Rencontres Francophones sur les Aspects Algorithmiques des Télécommunications*. 2025 (cit. on pp. 154, 155, 161, 179).
- [CDM23b] Emma Caizergues, François Durand, and Fabien Mathieu. “Trier des cochons sauvages”. In: *AlgoTel 2023-25èmes Rencontres Francophones sur les Aspects Algorithmiques des Télécommunications*. 2023 (cit. on p. 154).
- [CU20] Clément Calauzènes and Nicolas Usunier. “On ranking via sorting by estimated expected utility”. In: *Advances in Neural Information Processing Systems - NeurIPS '20*. Vol. 33. 2020, pp. 3699–3710 (cit. on p. 161).
- [Car+10] Jean Cardinal, Samuel Fiorini, Gwenaël Joret, Raphaël M. Jungers, and J. Ian Munro. “Sorting under partial information (without the ellipsoid algorithm)”. In: *Proceedings of the Forty-Second ACM Symposium on Theory of Computing - STOC '10*. 2010, pp. 359–368 (cit. on pp. 153, 169).

- [Ciu01] Marcin Ciura. “Best increments for the average case of Shellsort”. In: *Fundamentals of Computation Theory: Thirteenth International Symposium*. 2001, pp. 106–117 (cit. on p. 173).
- [DG77] Persi Diaconis and Ronald L. Graham. “Spearman’s footrule as a measure of disarray”. In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 39.2 (1977), pp. 262–268 (cit. on p. 157).
- [DM18] Eyal Dushkin and Tova Milo. “Top- k sorting under partial order information”. In: *Proceedings of the 2018 International Conference on Management of Data*. 2018, pp. 1007–1019 (cit. on p. 153).
- [FJ59] Lester R. Ford and Selmer M. Johnson. “A tournament problem”. In: *The American Mathematical Monthly* 66.5 (1959), pp. 387–389 (cit. on pp. 153, 174).
- [GZ96] Joshua Grass and Shlomo Zilberstein. “Anytime Algorithm Development Tools”. In: *SIGART Bulletin Special Issue on Anytime Algorithms and Deliberation Scheduling* 7.2 (1996), pp. 20–27 (cit. on p. 153).
- [Hoa61] Charles A. R. Hoare. “Algorithm 65: find”. In: *Communications of the ACM* 4.7 (1961), pp. 321–322 (cit. on pp. 158, 160).
- [Hor88] Eric J. Horvitz. “Reasoning under Varying and Uncertain Resource Constraints”. In: *Proceedings of the Seventh AAAI National Conference on Artificial Intelligence - AAAI '88*. Saint Paul, Minnesota, 1988, pp. 111–116 (cit. on p. 153).
- [Knu98] Donald E. Knuth. *The Art of Computer Programming, Volume 3 (2nd Ed.): Sorting and Searching*. Addison Wesley Longman Publishing Co., Inc., 1998 (cit. on p. 158).
- [Kuh55] Harold W. Kuhn. “The Hungarian method for the assignment problem”. In: *Naval Research Logistics Quarterly* 2.1-2 (1955), pp. 83–97. eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1002/nav.3800020109> (cit. on p. 160).
- [MF18] Amir Mesrikhani and Mohammad Farshi. “Progressive sorting in the external memory model”. In: *The CSI Journal on Computer Science and Engineering* 15.2 (2018) (cit. on p. 152).
- [MY16] Shay Moran and Amir Yehudayoff. “A Note on Average-Case Sorting.” In: *Order* 33.1 (2016), pp. 23–28 (cit. on p. 153).
- [Pec04] Marcin Peczarski. “New Results in Minimum-Comparison Sorting”. In: *Algorithmica* 40.2 (2004), pp. 133–145 (cit. on p. 153).

- [PP21] Dominik Peters and Ariel D. Procaccia. “Preference Elicitation as Average-Case Sorting”. In: *Proceedings of the AAAI Conference on Artificial Intelligence - AAAI '21*. Vol. 35(6). 2021, pp. 5647–5655 (cit. on pp. 153, 156).
- [Vox] Vox Pop Labs. *Vote Compass*. URL: <https://www.abc.net.au/news/vote-compass/> (cit. on p. 179).
- [Zil95] Shlomo Zilberstein. “Optimizing Decision Quality with Contract Algorithms”. In: *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence - IJCAI '95*. 1995, pp. 1576–1582 (cit. on p. 154).
- [Zil96] Shlomo Zilberstein. “Using Anytime Algorithms in Intelligent Systems”. In: *AI Magazine* 17.3 (1996), p. 73 (cit. on p. 154).

RÉSUMÉ

Cette thèse étudie la probabilité du paradoxe de Condorcet, une situation dans laquelle l'agrégation des préférences individuelles conduit à un cycle, même quand chaque électeurice possède des préférences transitives. Tandis que la plupart des travaux précédents utilisent des approches probabilistes, nous explorons une voie alternative fondée sur la combinatoire analytique. Nous commençons par étudier la distribution uniforme des préférences en encodant un profil de vote sous forme de graphe. Au passage, nous obtenons des résultats pertinents sur les graphes réguliers et les matrices équilibrées. Nous étendons ensuite l'analyse à des modèles de préférences plus généraux. Via des méthodes de point col, non seulement nous retrouvons des limites connues pour la probabilité d'existence du paradoxe de Condorcet, mais nous obtenons en plus des taux de convergence. Cette approche fournit également des résultats en combinatoire analytique multivariée présentant un intérêt indépendant. Enfin, puisque l'ensemble de la thèse suppose que les électeurices fournissent des classements complets, le dernier chapitre s'écarte de l'étude probabiliste pour aborder une question pratique : comment aider les électeurices à construire efficacement leur classement ? Nous proposons une procédure de tri et discutons de ses applications potentielles dans des contextes réels de vote à travers l'exemple de l'élection fédérale australienne de 2025.

MOTS CLÉS

choix social, probabilité, paradoxe de Condorcet, système de vote, combinatoire analytique, méthode du point col, méthode symbolique, tri interruptible, élicitation de préférence

ABSTRACT

This thesis studies the probability of the Condorcet paradox, a situation in which the aggregation of individual preferences yields a cyclic outcome, despite each voter having transitive preferences. While most previous work focused on probabilistic approaches, we investigate an alternative route based on analytic combinatorics. We first consider the uniform distribution of preferences using a graph-based encoding of voting profiles, leading to significant results on regular graphs and balanced matrices that stand on their own. We then extend the analysis to more general preference models via saddle point methods, not only recovering known limits for the probability of a Condorcet winner but also refining them by providing explicit convergence rates. This approach also produces results in multivariate analytic combinatorics that are of independent interest. Finally, since we assume throughout the thesis that voters provide complete rankings, the last chapter departs from the probabilistic study to address a practical question: how can we help voters to efficiently elicit their full preferences, particularly when the number of candidates is large? To tackle this problem, we propose sorting algorithms and discuss their potential applications in real-world voting contexts, illustrated by the 2025 Australian federal election.

KEYWORDS

social choice, probability, Condorcet paradox, voting rule, analytic combinatorics, saddle point method, symbolic method, anytime sorting, preference elicitation