LIP6

M2 memoir

Automatic asymptotics for combinatorial series

Sébastien JULLIOT
Université Paris Diderot, Paris

Supervised by
Marc Mezzarobba
LIP6, Paris

September, 2020

Abstract

Enumerative combinatorics is interested in determining the number a_n of objects of size n in a class of combinatorial objects. Alternatively, rather than a complicated closed formula, one would like to obtain an asymptotic expansion of a_n. In analysis of algorithms, for instance, computing asymptotic expansions is used to compare performance, and is therefore a salient question.

Due to the fabulous diversity of combinatorial structures, such computations have long required intuition and specially crafted "tricks", only adapted to the problem at hand, or some close family.

Nowadays, powerful techniques have been developed, enabling one to study a vast amount of combinatorial constructions with standard procedures. In fact, most of the steps involved have now been separately implemented, for instance in the `ore_algebra` module of *SageMath*.

One frequent case of application of these techniques is the *D-Finite* case, where the generating series of (a_n) is characterized by a differential equation with polynomial coefficients, and initial values of the a_is.

This internship is devoted to understanding and effectively programming a combination of these techniques in *SageMath*, in the D-Finite case. In the end, one should be able to type in a differential equation associated to a_n, and our code shall return an asymptotic expansion of a_n, up to any desired order.

The novelty resides in the merging of several existing functions used in different parts of the analysis, associated to the computation of explicit constants.

Contents

Chapter 1

Introduction

Pick your favourite combinatorial construction. Let a_n the number of such structures of size n. We wish to be able to compute an asymptotic expansion of a_n automatically.
Let $f = \sum a_n z^n$ be the formal series associated to (a_n).
We shall see that if f, as a complex series, has a positive convergence radius, then one has asymptotically $a_n = A^n \theta(n)$ where θ has sub-exponential growth. Two principles, stated in [FS09], shall guide our search :

- *First Principle of Coefficient Asymptotics* : The location of a function's singularities dictates the exponential growth (A^n) of its coefficients.
- *Second Principle of Coefficient Asymptotics* : The nature of a function's singularities determines the associate subexponential factor $\theta(n)$.

From these principles, one can now infer how the method works : first compute the possible locations for singularities, then determine the behaviour of f around each singularity, and finally "translate" these results to an asymptotic expansion of a_n.

Holonomic functions In many combinatorial situations, one does not have a closed formula for f. In such cases, it is however often possible to determine a differential equation of which f is a solution. When the coefficients of that differential equation are polynomials, one says that f is *holonomic*, or *D-finite*.
Many usual functions are D-finite: polynomials, rational fractions, the exponential, logarithm, and trigonometric functions, etc. Moreover, the class of D-finite functions is closed under sum, product, derivation (cf. [Mel20]).
If one could come up with an algorithm to compute asymptotic expansions for all holonomic functions, one would therefore be able to give precise estimates for various combinatorial sequences. Such an algorithm is not yet reachable, though. It is indeed currently unknown how to handle the case of f having an essential singularity.
Nonetheless, assuming f behaves nicely, the way is clear now.

Certified digits The current process of determining an asymptotic expansion of (f_n) involves decomposing f in different bases. Obtaining a closed formula for the associated

connection coefficients is a hard problem (cf, for instance, [Mel20] for a small discussion on this topic). One can grasp this complexity through the fact that these connection coefficients can, in general, be transcendental numbers.
The situation is not hopeless, yet. It is possible to compute these coefficients up to arbitrary precision, and it so happens that Mezzarobba wrote a function in `ore_algebra` to this end ([Mez16]).

1.1 Mathematical sketch

From a combinatorial problem to a differential equation Powerful techniques exist to translate a combinatorial construction into a *D-finite* relation. We will not cover those techniques here. If interested, one is referred to [FS09].
From now on, we will assume that a non trivial D-finite relation satisfied by f is given, which is a relation of the form

$$y^{(r)} + \frac{p_{r-1}}{p_r} y^{(r-1)} + \cdots + \frac{p_0}{p_r} y = 0 \tag{1.1}$$

where $p_0, \ldots, p_r \in \mathbb{C}[X]$.

D-finite (holonomic) function

Definition 1. *A function satisfying a D-finite relation will be said D-finite itself, or* holonomic*.*

Singularities location We will first see that f may only have singularities at roots of p_r. Thereafter, we define $\Xi := \{\text{roots of } p_r\}$. If f has at least one non-zero singularity, minimal ones (by modulus) are called *dominant singularities*.

Local basis structure theorems Following the definition of *regular singular points*, where some technical condition is satisfied, it can be proved that, in a *slit* neighbourhood of any such point ζ, equation (1.1) admits a local basis of solutions of the form

$$(z-\zeta)^{\theta_j} \log^m(z-\zeta) H_j(z-\zeta)$$

with H_j analytic at 0. The first terms of this basis can be explicitly computed, and the associated coefficients for f can be numerically approximated, up to any desired precision.

Singularity Analysis Each term $(z-\zeta)^{\alpha} \log^m$ in the previous basis expansion contributes to the asymptotics of f_n. We provide explicit formulas for this "translation".

Transfer theorems We finally investigate *transfer theorems*, allowing one to account for the non-translated terms in the local-basis.

1.2 Implementation overview

The implementation is in SageMath, and vastly relies on the `ore_algebra` and `AsymptoticRing` modules.

`extract_asymptotics` A function `extract_asymptotics` is implemented, with the following definition:

```
def extract_asymptotics(op,
                        first_coefficients,
                        order=DEFAULT_ORDER,
                        precision=DEFAULT_PRECISION,
                        verbose=False,
                        result_var='n') -> expr
```

For a holonomic function f, `extract_asymptotics` takes

- A differential operator `op`, such that $op \cdot f = 0$
- A list `first_coefficients` of the first Taylor coefficients of f
- The desired `order` of the asymptotic expansion
- The desired certified `precision` for the constants
- A boolean `verbose`. When `True`, major step computations will be printed to the user.
- A string `result_var` to control the variable name in the result expression.

It returns a list of asymptotic expansions of the coefficients of f, up to the desired `order` and with constants certified at least with the given `precision`, ordered by root modulus. Here is a sample execution for the Catalan numbers:

```
sage: op = extract_asymptotics((4*z^2 - z)*Dz^2 + (10*z - 2)*Dz + 2, [1, 1, 2, 5, 14], order=5, precision=1e-10)
[(([1.0000000000 +/- 7.28e-12])/sqrt(pi))*4^n*n^(-3/2)
+ (([-1.1250000000 +/- 3.05e-11])/sqrt(pi))*4^n*n^(-5/2)
+ (([1.1328125000 +/- 6.92e-11])/sqrt(pi))*4^n*n^(-7/2)
+ (([-1.127929688 +/- 8.58e-10])/sqrt(pi))*4^n*n^(-9/2)
+ (([1.12728882 +/- 3.67e-9])/sqrt(pi))*4^n*n^(-11/2)
+ O(4^n*n^(-6))]
```

Global structure We first locate the roots of p_r, and group them by increasing modulus.

Then, as long as no root has been *proved* to be a singularity of f, we iterate through the groups and sum their contributions. A root of p_r can indeed not always be proved to be a singularity of f only by computing the coefficients of f in the local basis: if one of these coefficients is precisely 0, successive approximations will never be able to distinguish it from 0, yet not proving either that it would be nil.

Then for each root, we make use of (a personal version of) `local_basis_expansions`. That function allows us to compute a local basis of solutions to `op`, along with their generating series expansion up to any desired order. A call to `numerical_transition_matrix` then allows us to determine the expression of f in that local basis, with certified constants. Each term should then be transferred to `SingularityAnalysis` (from the `asymptotic_ring` module). Summing the results finally yields the desired expansion.
The process can be summarized as follows

Algorithm 1: Main algorithm

```
Input: f defined by a a_n d^n/dz^n f + ... a_0 = 0, initial coefficients f_0, ..., f_n, order
       and precision
Output: Asymptotic expansion of f with at least order terms, and coefficients
        with given precision
begin
  | Compute the roots of a_n and group them by increasing modulus.
  | Initialise a list L of contributions to the asymptotic expansion.
  | while No contribution confirmed do
  |   | Load next group G of roots
  |   | foreach root ρ ∈ G do
  |   |   | Compute the contribution of ρ.
  |   | end
  |   | Sum contributions of G and add to L
  | end
  | return L
end
```

1.3 State of the art

From the theoretical point of view, the method that we implement is now quite well established and understood. It has cousins, in particular the Birkhoff-Trjitznisky method, which is intended for P-recursive sequences.

Theory

About D-finite functions Most considerations on D-finite functions, such as the singularities possible locations and the structure theorems, are known for quite a time now. They are already described in Poole's book *Introduction to the Theory of Linear Differential Equations* ([Poo36]) in 1936.
A crucial paper is *Sur les séries de Taylor n'ayant que des singularités algébrico-logarithmiques sur leur cercle de convergence* ([Jun31]), where Junger explains how to compute explicit formulas for the asymptotic expansion of $\left(\frac{1}{1-z/\zeta}\right)^{\alpha}\left(\log\frac{1}{1-z/\zeta}\right)^{\beta}$ when β is an integer (which, for us, will always be true).
Another fundamental paper for our considerations is *Singularity Analysis of Generating Functions*, where Flajolet and Odlyzko [FO90] prove the transfer theorem.

The books *Analytic Combinatorics* [FS09] by Flajolet and Sedgewick, and *An Invitation to Analytic Combinatorics* [Mel20] by Stephen Melczer proved extremely useful as sources of pedagogy and examples.
This internship also heavily relied on the computation of certified connection coefficients, described in *Rigorous Multiple-Precision Evaluation of D-Finite Functions in SageMath* [Mez16], by Marc Mezzarobba.

P-recursive sequences and the Birkhoff-Trjitznisky method A sequence (u_n) is said P-recursive if it satisfies a recurrence equation with polynomial coefficients:

$$p_0(n)u_n + p_1(n)u_{n-1} + \cdots + p_r u_{n-r} = 0$$

It is a classical fact that a sequence is P-recursive if and only if the associated formal series is D-finite (see, for instance [FS09] or [Mel20]).
The Birkhoff-Trjitznisky method allows one to compute an asymptotic expansion of u_n, but the coefficients can only be heuristically estimated (no bound of correctness).

Implementations

In Maple The `Asyrec` package, written by Doron Zeilberger (see this webpage), implements the Birkhoff-Trjitznisky method ; it therefore works with P-recursive sequences rather than D-finite functions, and the constants are only empirically estimated.
The `gfun` package by Bruno Salvy [SZ92] allows one to perform various operations with differential operators and D-finite functions. It does not, however, offer methods for singularity analysis.
The `gdev` package, also by Bruno Salvy [Sal91] performs singularity analysis on generating functions, given in closed form.

In SageMath The `ore_algebra` module (here) by Manuel Kauers, Maximilian Jaroschek, and Fredrik Johansson, performs various tasks on differential operators (among other things), such as computing a local basis of solutions, approximate the expression of a solution in such a basis, etc.
The `asymptotic_expansions` module offers a function `SingularityAnalysis` that computes an asymptotic expansion of the coefficients of a function in the specific form

$$\left(\frac{1}{1-z/\zeta}\right)^{\alpha}\left(\log\frac{1}{1-z/\zeta}\right)^{\beta}$$

Me make intensive use of both these modules.

Mathematica The package `GeneratingFunctions` by Mallinger performs automatic manipulations and transformations of holonomic functions.

Chapter 2

Mathematical background

2.1 Notations

Definition 2. *Let f be a holomorphic function. We note $f^{(k)}$ its k-th derivative, and $[z^k]f(z)$ its $(k+1)$th Taylor coefficient.*

2.2 Reminders from complex analysis

We only recall the Cauchy coefficients formula, as it is essential to our work.

Closed disk, circle

Definition 3. *Let $r > 0$ and $z \in \mathbb{C}$. We denote $\mathscr{D}(z_0, r)$ the closed disk of center z_0 and radius r, and $\mathscr{C}(z_0, r)$ its frontier.*

Cauchy's coefficient formula

Theorem 1. *Let f be analytic on some neighbourhood Ω of $z_0 \in \mathbb{C}$, and $r > 0$ such that $\mathscr{D}(z_0, r) \subset \Omega$.*
Then for all n, one has

$$f_n = \frac{1}{2i\pi} \int_{\mathscr{C}(z_0,r)} \frac{f(z)}{(z-z_0)^{n+1}} dz$$

Complex logarithm

A few identities For a reference on the following definitions and identities, the reader is referred to [BC96].
From now on, except when explicitly stated otherwise, $\log z$ and $\arg z$ will stand for the principal determination of the complex logarithm and argument.

N_+, N_-

Definition 4. *Let $z_1, z_2 \in \mathbb{C}^*$. Define $N_+(z_1, z_2)$ and $N_-(z_1, z_2)$ with*

$$N_\pm = \begin{cases} -1 & \text{if } \pi < \arg(z_1) \pm \arg(z_2) \\ 0 & \text{if } -\pi < \arg(z_1) \pm \arg(z_2) \leq \pi \\ 1 & \text{if } \arg(z_1) \pm \arg(z_2) \leq -\pi \end{cases}$$

Remark 1. *The previous definition is intended to have to following relations hold:*

$$\begin{cases} \arg(z_1 z_2) & = \arg(z_1) + \arg(z_2) + 2\pi N_+ \\ \arg\left(\frac{z_1}{z_2}\right) & = \arg(z_1) - \arg(z_2) + 2\pi N_- \end{cases}$$

Theorem 2. *Let $a, b, c \in \mathbb{C}^*$. Then*

$$\log(ab) = \log a + \log b + 2i\pi N_+(a, b)$$
$$\log\left(\frac{a}{b}\right) = \log a - \log b + 2i\pi N_-(a, b)$$
$$(ab)^c = a^c \times b^c \times e^{2i\pi c N_+(a,b)}$$
$$\left(\frac{a}{b}\right)^c = \frac{a^c}{b^c} e^{2i\pi c N_-(a,b)}$$

Corollary 1. *Let $x \in \mathbb{R}^{+*}$ and $z, t \in \mathbb{C}^*$. Then $\arg(x) = 0$, so all classical identities over the real numbers extend identically:*

$$\log(xz) = \log x + \log z$$
$$\log\left(\frac{x}{z}\right) = \log x - \log z$$
$$(xz)^t = x^t \times z^t$$
$$\left(\frac{x}{z}\right)^t = \frac{x^t}{z^t}$$

2.3 Asymptotic expansions and scales

Asymptotic scale (ϕ_n)

Definition 5. *Let $z_0 \in \mathbb{C}$.*
Let (ϕ_n) be a sequence of functions, continuous in z_0.
We say that (ϕ_n) is an asymptotic scale *if, for every n, we have*

$$\phi_{n+1}(z) = o(\phi_n(z))$$

as z tends to z_0

Asymptotic expansion

Definition 6. *Let (ϕ_n) be an asymptotic scale in a neighbourhood of z_0 and $N \in \mathbb{N}$. If we have*

$$f(z) = \sum_{n=0}^{N-1} a_n \phi_n(z) + O(\phi_N(z))$$

as z tends to z_0, then we say that f admits an asymptotic expansion *of order N around z_0.*
When this expression is valid for all N, we note

$$f \sim \sum a_n \phi_n$$

2.4 Some differential equations theory

In this section, we present the results enabling one to study a differential equation

$$y^{(r)} + \frac{p_{r-1}}{p_r} y^{(r-1)} + \cdots + \frac{p_0}{p_r} y = 0 \tag{2.1}$$

Scalar and system equations

Differential equations The equation

$$y^{(r)} = a_{r-1}(z) y^{(r-1)} + \cdots + a_0(r) y \tag{2.2}$$

where the a_i are holomorphic is said to be a *scalar* (differential) equation.

Differential systems The equation

$$Y' = A(z) Y \tag{2.3}$$

where $A(z)$ is an $n \times n$ matrix and $Y(z)$ is an n-dimensional vector is said to be a (differential) *system of equations.*

From scalar to system The following transformation is a classical trick to transform a scalar equation into a system:
If y is a solution to

$$y^{(r)}(z) = a_{r-1}(z) y^{(r-1)}(z) + \cdots + a_0(z) y(z)$$

then $Y : z \mapsto \begin{pmatrix} y(z) \\ y'(z) \\ \vdots \\ y^{(n-1)}(z) \end{pmatrix}$ is a solution to

$$Y' = A(z) Y'(z)$$

where

$$A(z) = \begin{pmatrix} 0 & 1 & & & \\ & 0 & 1 & & \huge{0} \\ \huge{0} & & \ddots & & 1 \\ a_0(z) & & \dots & & a_{r-1}(z) \end{pmatrix}$$

We call A the *companion matrix* of equation (2.4).

Solutions space

Basis of solutions The following classical theorem is admitted.

Cauchy's existence and uniqueness theorem

Theorem 3. *Let n an integer. Let also $A(z)$ an $n \times n$-matrix and $f(z)$ an n-dimensional vector, both holomorphic in some simply connected region $\Omega \subset \mathbb{C}$. Then, for any $z_0 \in \Omega$ and $y_0 \in \mathbb{C}^n$, the equation*

$$Y' = A(z)Y \tag{2.4}$$

has a unique solution such that

$$y(z_0) = y_0$$

That solution is holomorphic on Ω.

It follows

Basis of solutions for systems

Corollary 2. *Let $z_0 \in \mathbb{C}$. Suppose there exists a neighbourhood Ω of z_0 such that A and f are holomorphic on Ω.*
Then the set of solutions to equation (2.4) defined in z_0 forms an n-dimensional vector space.

2.5 Location of Singularities

Existence of a local basis of solutions

Theorem 4. *Let $p_0, \dots, p_r \in \mathbb{C}[X]$ and $z_0 \in \mathbb{C}$ such that $p_r(z_0) \neq 0$.*
Then, in some neighbourhood of z_0, the equation

$$y^{(r)} + \frac{p_{r-1}}{p_r} y^{(r-1)} + \cdots + \frac{p_0}{p_r} y = 0 \tag{2.5}$$

admits a basis of r analytic solutions.

Proof. The polynomial p_r has finite degree, therefore has a finite number of roots.
Since $p_r(z_0) \neq 0$, there is some neighbourhood Ω of z_0 where p_r does not vanish.
It follows that all $\frac{p_i}{p_r}$ are analytic on Ω.
Cauchy's theorem then applies and concludes the proof. ■

Possible locations of singularities

Corollary 3. *The only points where f* may *admit singularities are the zeros of p_r.*

2.6 Structure theorems

For further treatment of this section, one is referred to [Was65] (chapter II in particular).

Another transformation

Let $\zeta \in \mathbb{C}$, y such that

$$y^{(r)} + a_{r-1}(z)y^{(r-1)} + \cdots + a_0(z)y = 0$$

and define $Y : z \mapsto \begin{pmatrix} y(z) \\ \vdots \\ (z-\zeta)^{r-1} y^{(r-1)}(z) \end{pmatrix}$ (that is, $Y_i : z \mapsto (z-\zeta)^{i-1} y^{(i-1)}(z)$).
For all $i \leq r-1$, we have

$$(z-\zeta)Y_i'(z) = (i-1)Y_i(z) + Y_{i+1}(z)$$

and

$$(z-\zeta)Y_r'(z) = (r-1)Y_r(z) - (z-\zeta)a_{r-1}(z)Y_{r-1}(z) - \cdots - (z-\zeta)^r a_0(z)Y_1(z)$$

Then, equation (2.1) is equivalent to

$$(z-\zeta)Y' = A_\zeta(z)Y \tag{2.6}$$

where $A_\zeta(z)$ is an $r \times r$ matrix

$$A_\zeta(z) = \begin{pmatrix} 0 & 1 & & \\ & 1 & 1 & \text{\Large 0} \\ \text{\Large 0} & \ddots & \ddots & \\ & & r-1 & 1 \\ -a_0(z)(z(\zeta))^r & \dots & -a_{r-1}(z)(z(\zeta)) & r-1 \end{pmatrix}$$

Regular singular points and indicial polynomials

Regular singular points

Definition 7. *Consider a differential equation (E)*

$$y^{(r)} + \frac{p_{r-1}}{p_r} y^{(r-1)} + \dots + \frac{p_0}{p_r} y = 0 \tag{2.7}$$

We say that ζ is a regular singular *point of (E), if ζ is a pole of $\frac{p_i}{p_r}$ of order at most $r - i$, for all $i \in [0, r-1]$.*
Equivalently, ζ is a regular singular point if $A_\zeta(z)$ is analytic in some neighbourhood of ζ, when one writes $(z-\zeta)Y' = A_\zeta(z)Y$.

Indicial polynomial, I_ζ

Definition 8. *The characteristic polynomial of $A_\zeta(\zeta)$ is called the* indicial polynomial *of equation* (2.1) *and* (2.6) *at ζ, denoted I_ζ.*

Results

General structure theorem

Theorem 5. *Let ζ be a regular singular point of* (2.1)*. No assumption is made on the roots of I_ζ.*
Then, in a slit neighbourhood of ζ, there exists a basis of solutions of the form

$$(z-\zeta)^{\theta_j} (\log(z-\zeta))^m H_j(z-\zeta) \tag{2.8}$$

where θ_j are the roots of the indicial polynomial, each H_j is analytic at 0, and $m \in \mathbb{N}$.

Special cases

G-functions

Definition 9. *A formal series $f = \sum f_n z^n \in \mathbb{Q}[[z]]$ is called a* G-function *if it is D-finite and there exists $C > 0$ such that for all n, we have*

$$\begin{cases} |f_n| < C^n \\ lcd(f_1, \ldots, f_n) < C^n \end{cases}$$

where $lcd(f_1, \ldots, f_n)$ is the least common denominator of $f_1, \ldots, f_n$.

André-Chudnovsky-Katz Theorem

Theorem 6. *Let f be a G-function. Then a minimal order annihilating D-finite equation for f has only ordinary or regular singular points, and its indicial polynomial has only rational roots.*

Chapter 3

Singularity analysis and transfer theorems

In the previous sections, we have seen that a holonomic function f admits, in the neighbourhood of any singularity, an asymptotic expansion of the form (2.8).
In this section, we show how to compute an asymptotic expansion of f_n. If $f(z) = g_1(z) + \cdots + g_p(z) + 0(g_{p+1}(z))$ in the neighbourhood of ζ, then singularity analysis determines the contribution of g_i to the asymptotic expansion of f_n, and the transfer theorem allows to deduce a valid asymptotic expansion.

3.1 Singularity analysis

Polynomial case

Theorem 7. *If $\alpha \in \mathbb{N}$, then $[z^n]\left(1 - \frac{z}{\zeta}\right)^\alpha = 0$ for $n > k$ because $\left(1 - \frac{z}{\zeta}\right)^\alpha$ is a polynomial. So that case can be completely ruled out in estimating asymptotic expansions.*

Basic scale transfer

We quote from [FS09]

Basic scale transfer

Theorem 8. *Let α be an arbitrary complex number in $\mathbb{C} \setminus \mathbb{Z}_{\leq 0}$. The coefficient of z^n in*

$$f(z) = (1-z)^{-\alpha}$$

admits for large n a complete asymptotic expansion in descending powers of n,

$$[z^n]f(z) \sim \frac{n^{\alpha-1}}{\Gamma(\alpha)} \left(1 + \sum_{k=1}^{\infty} \frac{e_k^{(\alpha)}}{n^k}\right)$$

where $e_k^{(\alpha)}$ is a polynomial in α of degree $2k$. More precisely,

$$e_k^{(\alpha)} = \sum_{i=k}^{2k} (-1)^i \lambda_{k,i} (\alpha+1)(\alpha+2)\dots(\alpha+i)$$

with $\sum_{k,i\geq 0} \lambda_{k,i} v^k t^i = e^t (1+vt)^{-1-1/v}$.

Complete scale

In this section, we closely follow [Jun31].

Lemma 1. *Let $k \in \mathbb{N}^*$ and $f = \sum f_n z^n$ with*

$$f(z) = (1-z)^{-k}$$

Then for all n, we have

$$f_n = \frac{n^{k-1}}{\Gamma(k)} \left[1 + \frac{k(k-1)}{2n} + \dots + \frac{\Gamma(k)}{n^{k-1}}\right]$$

Proof. Start with

$$(1-z)^{-1} = \sum z^n$$

Now differentiating that relation $k-1$ times, we get

$$(k-1)!(1-z)^{-k} = \sum \frac{(n+k-1)!}{n!} z^n$$

Therefore $f_n = \frac{1}{(k-1)!}(n+1)(n+2)\dots(n+k-1)$, and the result follows by developing the product and grouping by powers of n. ■

The following three results are quoted from [Jun31] without proof. For a reference, one may consult [KY28]. They shall be useful in the proof of theorem 9 to assert the domains of validity of the computed expansions.

Lemma 2. *Let $\phi(z)$ admit an asymptotic expansion*

$$\varphi(z) \sim c_0 + \frac{c_1}{z} + \frac{c_2}{z^2} + \dots \tag{3.1}$$

as z goes to infinity following a half line d.
Then for every constant z_0, we also have, asymptotically along d,

$$\varphi(z_0 + z) \sim c_0 + \frac{c_1}{z} + \frac{-c_1 z_0 + c_2}{z^2} + \dots$$

Lemma 3. *Let $\varphi(z)$ be analytic in the form (3.1) on a half band*

$$\begin{cases} \Re(z) > a \\ \Im(z) \in]-b, b[\end{cases} \tag{3.2}$$

for a and b arbitrary positive real numbers.
Then $e^{\varphi(z)}$ can also be expanded, over the same band:

$$e^{\varphi(z)} \sim e^{c_0}\left(1 + \frac{c_1}{z} + \dots\right)$$

Lemma 4. *Let $\varphi(z)$ be analytic and expandable in the form (3.1) over a band (3.2), we also have*

$$\varphi'(z) \sim -\frac{c_1}{z^2} - \frac{2c_2}{z^2} - \dots$$

over any tighter band

$$\begin{cases} \Re(z) > a \\ \Im(z) \in]-b+\varepsilon, b+\varepsilon[\end{cases} \tag{3.3}$$

Lemma 5. *Let i an integer, n a natural number and $s \in \mathbb{C} \setminus \{-1, -2, \dots\}$.*
Then there exists functions $\psi_{i,j}$ that can be expanded asymptotically, such that

$$\frac{\Gamma^{(i)}(n+s)}{\Gamma(n+1)} = n^{s-1}[(\log n)^i \psi_{i,0}(n) + \dots + \psi_{i,i}(n)] \tag{3.4}$$

Proof. Start from Stirling's series for $\log \Gamma(z)$:

$$\begin{aligned} \log \Gamma(z) &\sim \sum_{n=1}^{\infty} \frac{B_{2n}}{2n(2n-1)z^{2n-1}} \\ &\sim \frac{1}{2}\log(2\pi) + \left(z + \frac{1}{2}\right)\log(z) - z + \frac{1}{12z} - \frac{1}{360z^2} + \dots \end{aligned}$$

where B_n is the n-th Bernoulli number.
Then by lemma 3

$$\Gamma(z) = \left(\frac{z}{e}\right)^z \cdot z^{-1/2} \cdot \varphi(z)$$

where $\varphi(z)$ can be expanded in asymptotic series over any half-band of type (3.2). Now, differentiating i times, we get

$$\Gamma^{(i)}(z) = \left(\frac{z}{e}\right)^z z^{-1/2}\left[(\log z)^i \varphi_{i,0}(z) + \cdots + \varphi_{i,i}(z)\right]$$

where the functions $\varphi_{i,j}$ can be expanded into asymptotic series, by lemmas 2 and 4. Therefore,

$$\frac{\Gamma^{(i)}(n+s)}{\Gamma(n+1)} = \left(\frac{n}{e}\right)^{s-1} \frac{\left(1+\frac{s}{n}\right)^{n+s-1/2}}{\left(1+\frac{1}{n}\right)^{n+1/2}} \cdot \frac{(\log(n+s))^i \varphi_{i,0}(n+s) + \cdots + \varphi_{i,i}(n+s)}{\varphi(n+1)}$$

and we may finally define the functions $\psi_{i,j}$ such that

$$\frac{\Gamma^{(i)}(n+s)}{\Gamma(n+1)} = n^{s-1}[(\log n)^i \psi_{i,0}(n) + \cdots + \psi_{i,i}(n)]$$

■

Expansion theorem in the log case

Theorem 9. *Let $a \in \mathbb{C}$ and $k \in \mathbb{N}^*$.*
Let

$$f : z \mapsto (1-z)^a \left(\log \frac{1}{1-z}\right)^k$$

then for large n one has

$$f_n = \begin{cases} \frac{n^{-a-1}}{\Gamma(-a)} \sum_{i=0}^{k} (\log n)^i \phi_i(n) & \text{if } a \notin \mathbb{N} \\ (-1)^a k \Gamma(1+a) n^{-a-1} \sum_{i=0}^{k} (\log n)^i \phi_i(n) & \text{if } a \in \mathbb{N} \end{cases}$$

where the functions ϕ_i admit asymptotic expansions of the form

$$\phi_i \sim c_{i,0} + \frac{c_{i,1}}{n} + \frac{c_{i,2}}{n^2} + \dots$$

and the constants $c_{i,j}$ can be explicitly computed.

Proof. Assume $a \notin \mathbb{N}$. We can define

$$\phi_0 : z \mapsto \Gamma(-a)(1-z)^{-a} = \sum_{n=0}^{\infty} \frac{\Gamma(n-a)}{n!} z^n$$

By differentiating i times with respect to a, we get $\frac{d^i}{da}\Gamma(-a) = (-1)^i \Gamma^{(i)}(-a)$, and $\frac{d^i}{da}(1-z)^{-a} = \frac{d^i}{da} e^{-a\log(1-z)} = \frac{d^i}{da} e^{a\left(\log \frac{1}{1-z}\right)} = \left(\log \frac{1}{1-z}\right)^i (1-z)^{-a}$ so by Leibniz' rule:

$$\phi_i := \frac{d^i}{da^i}\phi_0 = (1-z)^{-a} \sum_{j=0}^{i} \binom{i}{j} (-1)^j \Gamma^{(j)}(-a) \left(\log \frac{1}{1-z}\right)^{i-j}$$

Now, when i takes successively the values $0, \dots, k$, we get a triangular system of linear equations of unknowns the functions $(1-z)^a\left(\log \frac{1}{1-z}\right)^i$, the solution of which has the form

$$(1-z)^a\left(\log \frac{1}{1-z}\right)^k = \frac{1}{\Gamma(-a)}\left[\phi_k(z) + d_{k,k-1}\phi_{k-1}(z) + \cdots + d_{k,0}\phi_0(z)\right] \tag{3.5}$$

where the coefficients $d_{i,j}$ are explicitly computable and only depend on i and j.
Now, by definition of the ϕ_is, we have for all i

$$\phi_i(z) = (-1)^i \sum_{n=0}^{\infty} \frac{\Gamma^{(i)}(n-a)}{n!} z^n$$

By expanding equation (3.5) into Taylor series, this leads to the following equality

$$f_n = \frac{1}{n!\Gamma(-a)}\left[\Gamma^{(k)}(n-a) + d_{k,k-1}\Gamma^{(k-1)}(n-a) + \cdots + d_{k,0}\Gamma(n-a)\right]$$

We now use lemma 5 to conclude (recall that $n! = \Gamma(n+1)$).

***

To deal with the case $a \in \mathbb{N}$, use the relation

$$\begin{aligned}(1-z)^a\left(\log \frac{1}{1-z}\right)^k = &- a\int (1-z)^{a-1}\left(\log \frac{1}{1-z}\right)^k \\ &+ k\int (1-z)^{a-1}\left(\log \frac{1}{1-z}\right)^{k-1}\end{aligned}$$

$a+1$ times to reduce to the first case. ■

3.2 Transfer theorems

The content of this section is from [FS09], section VI.

$\Delta(\phi, R)$

Definition 10. *For any positive numbers ϕ and R such that $R > 1$ and $\phi \in \left]0, \frac{\pi}{2}\right[$, we define*

$$\Delta(\phi, R) := \{z \mid |z| < R, z \neq 1 \textit{ and } |\arg(z-1) > \phi\}$$

*A domain D such that there exists ϕ and R as above and $D = \Delta(\phi, R)$ will be called a Δ-*domain.
A function that is analytic on some Δ-domain will be called Δ-analytic.

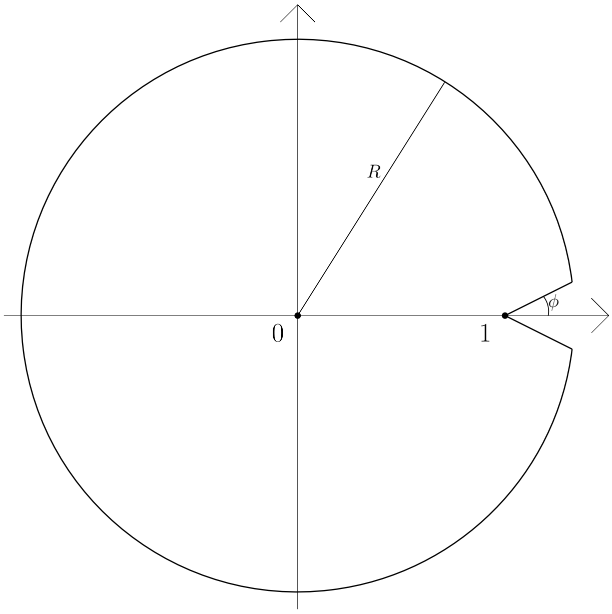

Figure 3.1: A Δ-domain

Transfer theorem for O and o

Theorem 10. *Let $\alpha, \beta \in \mathbb{R}$ and f be a Δ-analytic function.*

- *If f is such that, in the some neighbourhood of 1 and its Δ-domain, one has*

$$f(z) = O\left((1-z)^{-\alpha}\left(\log\frac{1}{1-z}\right)^{\beta}\right)$$

Then

$$[z^n]f(z) = O\left(n^{\alpha-1}(\log n)^{\beta}\right)$$

- *The same result holds, with O replaced by o.*

Chapter 4

Implementation

First, section 4.1 briefly discusses known limitations of the current program. Then section 4.2 presents most parts of the algorithm, section 4.3 briefly presents the tests. Finally, section 4.4 discusses performance aspects.

4.1 Limitations

The current version can not handle entire functions, nor functions with only a singularity in 0.
Due to the way `SageMath` implements numbers, it can not handle the case of irrational solutions to the polynomial equations. This is the only known limitation that restricts us further apart the general results of the maths chapter.

4.2 Problems and solutions

Computing the roots of p_r When `op` represents the differential operator associated to $p_r f^(r) + \cdots + p_0$, we call `op.leading_coefficient().roots(QQbar)` to compute the roots of p_r. This function returns a list of tuples `(root, multiplicity)`. Internally, a root is an algebraic number. `Sage` deals with algebraic numbers by remembering the polynomial they come from, and a sufficiently precise approximation.

Asserting a point is singular regular A root ρ of p_r has to be regular or singular regular for our analysis to be valid. We therefore implement a simple function to assert each encountered root is a regular singular point.

```
def assert_point_is_regular_singular(point, leading_mult, op):
    z, r = op.base_ring().gen(), op.order()
    for i, poly in enumerate(op):
        if r - i < leading_mult \
                and not ((z - point)^(leading_mult - (r - i))).divides(poly):
            raise ValueError(f'Got an irregular singular point ({point}).')
```

Computing the decomposition of f in 0 To compute the decomposition of f in 0, we make use of the `local_basis_structure` function, which computes the "distinguished monomial" associated to each solution in the local basis. Each of these monomials, in the form $z^n \log(z)^k$, characterizes the solution it belongs to, so we only have to fetch its coefficient in the Taylor coefficients of f.

```
def compute_initial_decomp(op, first_coefficients):
    distinguished_monomials = Point(0, op).local_basis_structure()
    proj = [0] * op.order()
    for i, sol in enumerate(distinguished_monomials):
        n, k = simplify_exponent(sol.valuation), sol.log_power
        if n in NN and n >= 0 and k == 0:
            try:
                proj[i] = first_coefficients[n]
            except IndexError:
                raise Exception('Not enough first coefficients supplied')
    return proj
```

Computing a local basis of solutions (`my_expansions`) The computation of a local basis of solutions is achieved by calling `my_expansions`. This function is a re-implementation of `local_basis_expansions`, modified to make its return easily usable (as opposed to the symbolic expression normally returned by `local_basis_expansions`, nice when printed but rather intricate).
Both make use of `LocalBasisMapper`, also implemented in `ore_algebra`, which is mostly an generator for the local basis terms.

```
def my_expansions(op, point, order):
    mypoint = Point(point, op)
    ldop = op.shift(mypoint)
    class Mapper(LocalBasisMapper):
        def fun(self, ini):
            return log_series(ini, self.shifted_bwrec, order)
    sols = Mapper(ldop).run()
    return [[(c / ZZ(k).factorial(),
                point,
                sol.leftmost + n,
                k)
            for n, vec in enumerate(sol.value)
            for k, c in reversed(list(enumerate(vec)))
            if c != 0]
        for sol in sols]
```

Computing the decomposition of f in $\rho \neq 0$ Assumming ρ is a regular or regular singular point, the function `numerical_transition_matrix` computes the transition

matrix $\mathscr{B}_{0\to\rho}$ between 0 and ρ with a prescribed precision.
The computation of the decomposition v_ρ of f in ρ is now achieved by multiplying $\mathscr{B}_{0\to\rho}$ and v_0 (vector of coordinates of f in 0).

```
trans_matrix = op.numerical_transition_matrix([0, root],
                                              eps=precision,
                                              assume_analytic=True)
coeffs_in_local_basis = trans_matrix * vector(ini)  # `ini` is v_0
```

Computing the contribution of a term The contribution of a term

$$(z-\zeta)^a(\log(z-\zeta))^b$$

is computed using `SingularityAnalysis`. But this function considers functions in a slightly different form:

$$\left(\frac{1}{1-z/\zeta}\right)^\alpha\left(\log\frac{1}{1-z/\zeta}\right)^\beta$$

It is therefore necessary to perform a little transformation:

$$\begin{aligned}(z-\zeta)^a(\log(z-\zeta))^b &= ((-\zeta)(1-z/\zeta))^a(\log((-\zeta)(1-z/\zeta)))^b\\ &= (-\zeta)^\alpha(1-z/\zeta)^a(\log(-\zeta)+\log(1-z/\zeta))^b\\ &= (-\zeta)^\alpha\left(\frac{1}{1-z/\zeta}\right)^{-a}\sum_{i=0}^{b}\binom{b}{i}(\log(-\zeta))^{b-i}\left(-\log\frac{1}{1-z/\zeta}\right)^i\end{aligned}$$

(We are justified to perform transformations like $(xy)^t = x^ty^t$ since this expression only has to be valid in a Δ-domain)
Note that $(-\zeta)^\alpha\left(\frac{1}{1-z/\zeta}\right)^{-a}(\log(-\zeta))^b$ has a non-zero contribution if and only if $a \notin \mathbb{N}$.

```
def handle_monomial(zeta, alpha, k, order, verbose, result_var, precision):
    if k == 0:
        return (-zeta)^alpha * SA(result_var, zeta=zeta, alpha=-alpha,
                                  precision=order, normalized=False)

    res = (-1)^k * sum(binomial(k, i) * \
                       (log(-1/zeta))^(k-i) * SA(result_var,
                                                 zeta=zeta,
                                                 alpha=-alpha,
                                                 beta=i,
                                                 precision=order,
                                                 normalized=False)
                      for i in range(1, k+1))
    if alpha not in NN:
        res += (log(-1/zeta))^k * handle_monomial(zeta, alpha, 0,
                                                 order, verbose,
```

```
                                           result_var, precision)
    return res * (-zeta)^alpha
```

Remark 2. *`SingularityAnalysis` returns an expression, possibly with a O term. This expression behaves as one would expect; in particular, when summing two of these expressions, the big Os "absorb" the asymptotically inferior terms.*

Compute a O for the contribution of the remaining terms If the expression of f in the local basis has only a small number of terms (say, less than `order`), one only needs to perform singularity analysis on each term.

Assume the expression of f in the local basis was "truncated" when performing singularity analysis:

$$f(z) = \sum_{i=0}^{p} (z-\zeta)^{a_i} \log(z-\zeta)^{b_i} + O\left((z-\zeta)^{\alpha} \log(z-\zeta)^{\beta}\right)$$

By theorem 10 it suffices to compute a singularity analysis to order 0 on $((z-\zeta)^{\alpha}\log(z-\zeta)^{\beta}$ and add it to the previously computed contributions.

```
if order < len(expansion):
    _, _, alpha, m = expansion[order]
    if alpha not in NN or m > 0:
        last_term_expansion = handle_monomial(root, alpha, m, 0, verbose,
                                              result_var, precision)
        res.append(last_term_expansion)
```

Computing the contribution of a root To compute the contribution of a specific root ρ, we use the following algorithm (the corresponding `handle_root` function is not reproduced here for obvious length reasons)

Algorithm 2: Computing the contribution of a root

Input: All entries of the main algorithm, and a specific root ρ
Output: Contribution of ρ to the asymptotic expansion of f
begin
  Decompose f in $\mathscr{B}_\rho$.
  Initialize a sum S of terms contributions.
  foreach *term T in the local expansions* **do**
    Compute an asymptotic expansion of the Taylor coefficients of T.
    Add to S.
  end
  if *local expansion was truncated* **then**
    Compute a O for the contribution of the remaining terms.
    Add to S
  end
end
return S

4.3 Tests

A list of tests lies in `tests.sage`. Tests are separated into different testing strategies : first, a somewhat long list of simple and classical sequences are tested. Then, some possible cases of failure (no singularity, apparent singularity, irregular singular points) are tested through specially chosen examples.
The only non-passing test is a test where the program encounters an irrational number as root of the indicial polynomial.

Classical sequences

We give here a list of functions, along with an associated differential operator and an asymptotic expansion. We provide a name when the Taylor series is of combinatorial interest, and a reference for the not obvious/classical cases.

f	Differential operator	Asymptotic expansion	Sequence name
$\log(1-z)$	$(z-1)Dz^2 + Dz$	$\frac{-1}{n}$	
$\log(1+z)$	$(z+1)Dz^2 + Dz$	$\frac{(-1)^n}{n}$	
$\frac{1}{1-z}$	$(z-1)Dz + 1$	1	
$\frac{1}{1+z}$	$(z-1)Dz + 1$	$(-1)^n$	
$\frac{z}{1-2z}$	$z(1-2z)Dz - 1$	2^{n-1}	
$\frac{1}{1-z^2}$	$(1-z^2)Dz - 2z$	$\begin{cases} 1 \text{ if } n \text{ is even} \\ 0 \text{ otherwise} \end{cases}$	
$(1-z)^{3/2}$	$2(z-1)Dz - 3$	$\frac{1}{\sqrt{\pi n^5}}\left(\frac{3}{4} + \frac{45}{32n} + \frac{1155}{512n^2} + \dots\right)$	[FS09]
$\arctan(z)$	$(1+z^2)Dz^2 + 2zDz$	$\begin{cases} \frac{(-1)^n}{2n+1} \text{ if } n \text{ is odd} \\ 0 \text{ otherwise} \end{cases}$	
$\frac{1}{1-z}\log\left(\frac{1}{1-z}\right)$	$(1-z)^2Dz^2 - 3(1-z)Dz + 1$	$\log(n) + \gamma + \frac{1}{2n} + o\left(\frac{1}{n}\right)$	Harmonic numbers
$\frac{z}{1-z-z^2}$	$(1-z-z^2)Dz^2 - (2+4z)Dz - 2$	$\frac{\varphi^n - (-\varphi)^{-n}}{\sqrt{5}}$	Fibonacci numbers
$\frac{1-\sqrt{1-4z}}{2z}$	$(4z^2-z)Dz^2 + (10z-2)Dz + 2$	$\frac{4^n}{\sqrt{\pi n}}\left(1 - \frac{1}{8n} + \frac{1}{128n^2} + \dots\right)$	Catalan numbers
$\frac{1-z-\sqrt{1-2z-3z^2}}{2z}$	$(3z^4+2z^3-z^2)Dz^2+(6z^3+3z^2-z)Dz+1$	$\frac{\sqrt{3}}{2\sqrt{\pi}n^{3/2}}3^n\left(1-\frac{15}{16n}+\frac{505}{512n^2}+\dots\right)$	Motzkin numbers [FS09]
	$(4z^2-z)Dz^2 + (14z-2)Dz + 6$	$\frac{4^n}{\pi n}\left(4 - \frac{6}{n} + \frac{19-2(-1)^n}{2n^2} + \dots\right)$	Walks in $\mathbb{N}^2$ [Mel20]

Here is the return of a typical call to `tests.sage` (slightly formatted and with few digits for convenience):

```
·> sage tests.sage
log(1+z) -> [-1.00000*n^(-1)*(e^(I*arg(-1)))^n + O(n^(-3)*(e^(I*arg(-1)))^n)]

log(1-z) -> [([-1.00000 +/- 1e-10])*n^(-1) + O(n^(-3))]

1/(1-z) -> [1.00000]

1/(1+z) -> [1.00000*(e^(I*arg(-1)))^n]

1/(1 - z^2) -> [[0.500000 +/- 4.77e-7] + [0.500000 +/- 4.77e-7]*(e^(I*arg(-1)))^n
               + O(n^(-4)) + O(n^(-4)*(e^(I*arg(-1)))^n)]

1/(1-z) * log(1/(1-z)) (H_n) -> [([1.00000 +/- 1e-10] + [+/- 5.17e-26]*I)*log(n)
                                 + ([1.00000 +/- 1e-10] + [+/- 5.17e-26]*I)*euler_gamma
                                 + ([-1.00000 +/- 1e-10] + [+/- 5.17e-26]*I)*log(-1)
                                 + [+/- 1.63e-25] + [3.14159 +/- 3.94e-6]*I
                                 + ([0.500000 +/- 1e-11]
                                 + [+/- 2.59e-26]*I)*n^(-1)
                                 + O(n^(-2))]

z/(1-2z) -> [[0.500000 +/- 4.77e-7]*2^n]

Arctan -> [([+/- 5.57e-15] + [-0.500000 +/- 4.77e-7]*I)*n^(-1)*(e^(I*arg(1*I)))^n
            + ([+/- 5.57e-15] + [0.500000 +/- 4.77e-7]*I)*n^(-1)*(e^(I*arg(-1*I)))^n
            + O(n^(-3)*(e^(I*arg(-1*I)))^n) + O(n^(-3)*(e^(I*arg(1*I)))^n)]

random walks in Z*N -> [(([2.00000 +/- 1.91e-6])/sqrt(pi))*4^n*n^(-1/2)
                        + (([-1.25000 +/- 2.51e-6])/sqrt(pi))*4^n*n^(-3/2)
                        + (([1.14062 +/- 7.34e-6])/sqrt(pi))*4^n*n^(-5/2)
                        + (([-1.1230 +/- 5.61e-5])/sqrt(pi))*4^n*n^(-7/2)
                        + O(4^n*n^(-9/2))]

random walks in N^2 -> [([1.27324 +/- 1.98e-6] + [+/- 1.57e-7]*I)*4^n*n^(-1)
                        + ([-1.90986 +/- 2.96e-6] + [+/- 2.35e-7]*I)*4^n*n^(-2)
                        + ([0.318310 +/- 7.23e-7])*4^n*n^(-3)*(e^(I*arg(-1)))^n
                        + O(4^n*n^(-3))]

Fibonacci numbers -> [[0.44721 +/- 4.20e-6]*1.618033988749895?^n + O(1.618033988749895?^n*n^(-5))]

Catalan numbers -> [(([1.00000 +/- 9.54e-7])/sqrt(pi))*4^n*n^(-3/2)
                    + (([-1.12500 +/- 4.00e-6])/sqrt(pi))*4^n*n^(-5/2)
                    + (([1.1328 +/- 2.16e-5])/sqrt(pi))*4^n*n^(-7/2)
                    + (([-1.1279 +/- 9.19e-5])/sqrt(pi))*4^n*n^(-9/2)
                    + O(4^n*n^(-5))]

Motzkin numbers -> [(([0.86602 +/- 6.62e-6])/sqrt(pi))*3^n*n^(-3/2)
                    + (([-0.81190 +/- 6.64e-6])/sqrt(pi))*3^n*n^(-5/2)
                    + (([0.8542 +/- 2.92e-5])/sqrt(pi))*3^n*n^(-7/2)
                    + (([-0.855 +/- 3.66e-4])/sqrt(pi))*3^n*n^(-9/2)
                    + O(3^n*n^(-5))]

(1-z)^(3/2) -> [0.750000/sqrt(pi)*n^(-5/2)
               + 1.40625/sqrt(pi)*n^(-7/2)
               + (([2.25586 +/- 6.25e-7])/sqrt(pi))*n^(-9/2)
               + (([3.46069 +/- 3.36e-6])/sqrt(pi))*n^(-11/2)
               + (([5.22901 +/- 1.54e-6])/sqrt(pi))*n^(-13/2)
               + O(n^(-15/2))]

z(1-z) -> No singularity was found

exp(z) -> No singularity was found

sin(z) -> No singularity was found
```

4.4 Performance

Remark 3. *The following tests were only carried out through simple tools (`time` module from python, by averaging multiple calls, on my laptop with several other programs open, etc). Although the results seem in accordance with intuition and with the time spent on less expensive tests, one should keep in mind they only give a vague idea of real performance. In particular, I did not attempt to compute the "real" complexity.*

On my laptop, it takes in average 14 seconds to compute the expansion for Motzkin numbers, with `order=20` and `precision=10^(-100)`. Computing up to `precision=10^(-10000)` takes about 16 seconds. Reducing to `order=10` and `precision=10^(-100)` takes about 13 seconds, and `order=4` and `precision=10^(-100)` "only" 11 seconds.

Chapter 5

Conclusion

The goal of this internship was to produce a piece of code, able to compute an asymptotic expansion for the Taylor coefficients of a D-Finite function. It had to take in a function $\sum f_n z^n$, in the form of a differential equation and the first values of f_n, and return an asymptotic expansion of f_n, up to any desired order and precision.
That goal is achieved, as various tests indicate. The code runs in seconds, even for 20 terms and far too many certified digits.
Although many steps involved had already been implemented in `Sagemath`, no function was available to perform the full analysis. Therefore, Marc and I intend to submit the code for acceptance to `ore_algebra`, after a few corner cases fixes and refactoring steps.
For the future, possible improvements include handling singularities in 0, entire series, and computing explicit bounds on the error made by the computed asymptotic expansion.

Acknowledgement

I wish to express here my gratitude towards Marc for his trust and availability. This internship was only carried out to a term thanks to his constant attention and availability. In a time of great uncertainty about my future, and despite the challenges of an entirely remote internship, it was always a pleasure to work under his benevolent guidance.

Bibliography

[KY28] K. Knopp and R.C. Young. *Theory and Application of Infinite Series*. Theory and Application of Infinite Series vol. 2. Blackie & Son, 1928.

[Jun31] R. Jungen. “Sur les séries de Taylor n’ayant que des singularités algébrico-logarithmiques sur leur cercle de convergence.” In: *Commentarii mathematici Helvetici* 3 (1931), pp. 266–306. URL: `http://eudml.org/doc/138566`.

[Poo36] Poole. *Introduction To The Theory Of Linear Differential Equations.pdf*. 1936.

[Was65] Wolfgang Wasow. *Asymptotic expansions for ordinary differential equations*. 1965.

[FO90] Philippe Flajolet and Andrew Odlyzko. “Singularity Analysis of Generating Functions”. In: *SIAM Journal on Discrete Mathematics* 3.2 (1990), pp. 216–240. ISSN: 0895-4801. DOI: `10.1137/0403019`.

[Sal91] B. Salvy. “Examples of automatic asymptotic expansions”. In: *SIGSAM Bull.* 25 (1991), pp. 4–17.

[SZ92] Bruno Salvy and Paul Zimmermann. *GFUN : a maple package for the manipulation of generating and holonomic functions in one variable*. Research Report RT-0143. INRIA, 1992, p. 14. URL: `https://hal.inria.fr/inria-00070025`.

[BC96] James Ward Brown and Ruel V Churchill. *Complex variables and applications*. McGrawHill, 1996.

[FS09] Philippe Flajolet and Robert Sedgewick. *Analytic combinatorics*. 2009, pp. 1–810. ISBN: 9780511801655. DOI: `10.1017/CBO9780511801655`.

[Mez16] Marc Mezzarobba. “Rigorous Multiple-Precision Evaluation of D-Finite Functions in SageMath”. In: *5th International Congress on Mathematical Software (ICMS 2016)*. Berlin, Germany, July 2016. URL: `https://hal.archives-ouvertes.fr/hal-01342769`.

[Mel20] Stephen Melczer. “An Invitation to Analytic Combinatorics”. In: Cambridge, 2020.