



# **FUNCIONES HOLÓNOMAS**

**Isidro Matos Bellido**





# **FUNCIONES HOLÓNOMAS**

Isidro Matos Bellido

Memoria presentada como parte de los requisitos  
para la obtención del título de Grado en Matemá-  
ticas por la Universidad de Sevilla.

Tutelada por

Prof. Tutor Alberto Castaño Domínguez

Prof. Tutor Antonio Rojas León



# Índice general

<b>English Abstract</b>	<b>1</b>
<b>Agradecimientos</b>	<b>3</b>
<b>1. Introducción</b>	<b>5</b>
<b>2. El álgebra de Weyl</b>	<b>7</b>
2.1. Definición, forma normal y generadores . . . . .	7
2.2. Grado y orden . . . . .	13
2.3. Ideales . . . . .	16
<b>3. Módulos sobre el álgebra de Weyl</b>	<b>19</b>
3.1. El caso de los anillos de polinomios y de funciones holomorfas . . . . .	19
3.1.1. Anillos de polinomios . . . . .	19
3.1.5. Funciones holomorfas . . . . .	21
3.2. Soluciones de una ecuación diferencial . . . . .	24
3.3. Filtraciones y graduaciones . . . . .	26
3.4. Filtraciones en D-módulos . . . . .	29
<b>4. Funciones holónomas</b>	<b>35</b>
4.1. Ideales y módulos holónomos . . . . .	35
4.2. Definición y caracterización . . . . .	37
4.3. Conservación de la holonomía . . . . .	40
<b>5. Cálculos</b>	<b>49</b>
5.1. Área de la pantalla de una televisión . . . . .	49
5.2. Volúmenes . . . . .	55
5.2.1. Volumen de un toro . . . . .	56

## II FUNCIONES HOLÓNOMAS

5.2.2.	Volumen de un astroide . . . . .	58
5.2.3.	Volumen de un pretzel . . . . .	60
5.3.	Código . . . . .	61
5.3.1.	Área de la pantalla de una televisión . . . . .	62
5.3.2.	Volúmenes . . . . .	71

# English Abstract

The main objective of this project is being able to understand what a holonomic function is, studying its properties, as well as applying them in some examples. To achieve it, we will need to define and thoroughly study the Weyl algebra, which is going to be our working environment. The development followed through the first few chapters will lead us to the definition of a holonomic function, in addition to studying in depth its properties and proving that holonomic functions are preserved under several usual operations. Finally, we will move to the practical area of the project where we will work with SageMath to solve some specific problems involving holonomic functions.





# Agradecimientos

Me gustaría empezar el trabajo agradeciendo a aquellas personas que, desinteresadamente, han ayudado a que este trabajo sea lo que es hoy.

Primero, me gustaría agradecer a mis tutores la labor que han hecho durante el trabajo. A Antonio Rojas por facilitarnos todos los tramites para que Alberto y yo pudiéramos desarrollar este tema. Y a Alberto por todo el trabajo, por soportar mis correos electrónicos suplicando ayuda y por las... ¡32 semanas! que llevamos con el trabajo, que ha hecho que se pasen volando.

Por otra parte, también me gustaría agradecer a Peter Paul, director del *Research Institute for Symbolic Computation*, por ofrecernos acceso al paquete completo de Mathematica desarrollado por el *RISC*. También a Anna-Laura Sattelberger y Bernd Sturmfels, por proporcionarnos el programa que ellos utilizaron en los cálculos de su artículo [SS19]. Y por último, agradecer a Marc Mezzarobba por facilitarnos los ejemplos de regiones tridimensionales, así como todos los datos necesarios para el cálculo de su volumen y su programa de SageMath para poder comprobar los resultados.

A todos ellos, muchas gracias.



# 1 | Introducción

La teoría de  $D$ -módulos fue introducida en los setenta para intentar dar solución a ciertos problemas del análisis algebraico, desde entonces, se ha convertido en una herramienta muy útil en muchos ámbitos de las matemáticas, incluyendo algunos aspectos más aplicados como son el cálculo efectivo o la estadística. Esto ha sido así por el gran potencial que brindan los  $D$ -módulos. En este trabajo nos centraremos en las funciones holónomas, muy presentes en cálculos en las áreas antes mencionadas. Estas funciones pertenecen a mundos analíticos pero pueden definirse de modo algebraico y finito, lo que permite trabajar con ellas computacionalmente de modo muy eficiente. El objetivo de este trabajo es llegar a entender qué son las funciones holónomas, estudiar las propiedades de las mismas y aplicar lo anterior en algunos ejemplos. Para todo esto necesitaremos definir y estudiar a fondo al álgebra de Weyl, nuestro espacio de trabajo. Esto nos permitirá en el capítulo 4 dar la definición de función holónoma, profundizar en sus propiedades y comprobar que dichas funciones conservan la holonomía al operar con ellas. Una vez hagamos esto, podremos pasar a la parte más práctica del trabajo, donde veremos casos concretos, para los cuales usaremos SageMath, y quedará patente que el estudio de funciones holónomas facilita, de manera clara, la resolución de problemas.

En esta memoria seguimos muy de cerca la estructura y la presentación del artículo [SS19]. Para poder comprender en profundidad dicho artículo necesitamos antes varias nociones importantes de teoría de  $D$ -módulos, por lo que para complementarlo nos servimos en gran medida de [Cou95], principalmente para los capítulos 2 y 3, y de [Zei90] para el capítulo 4. Para el último capítulo tomamos como referencia tanto [SS19] como [LMD19].

Hagamos un breve resumen del contenido de este trabajo. Los primeros dos capítulos serán los que acumulen mayor peso teórico, y en ellos introduciremos y probaremos los resultados más necesarios sobre el álgebra de Weyl. En el primer capítulo se

introducirá formalmente el álgebra de Weyl, definiremos dos órdenes sobre ella y finalmente probaremos que es un álgebra simple. En el segundo estudiaremos el anillo de polinomios y las funciones holomorfas como  $D_n$ -módulos, entraremos de lleno en el estudio de ecuaciones diferenciales y veremos que el álgebra de Weyl, aunque no es un álgebra graduada, sí admite filtraciones.

En el cuarto capítulo llegamos a introducir a las funciones que dan nombre al trabajo, las funciones holónomas. Para ello necesitaremos introducir unos conceptos sobre polinomios de Hilbert, esto nos permitirá dar una definición y una caracterización que será vital en el resto del trabajo. Completaremos el capítulo viendo cómo se conserva la holonomía mediante las operaciones más usuales sobre funciones, para lo cual necesitaremos una serie de propiedades sobre la transformada de Fourier.

Todo esto nos permitirá, en el capítulo final, sacar buen provecho de las funciones holónomas y sus propiedades, para el cálculo del área o el volumen de algunas regiones, donde podremos apreciar lo realmente interesante e importante que es el estudio de las funciones holónomas. Usaremos SageMath y Mathematica tanto para el cálculo de estas áreas y volúmenes, como para su posterior verificación.

Aunque nuestro enfoque haya sido el cálculo del área o el volumen de regiones, como ya hemos dicho, los cálculos efectivos con funciones holónomas son muy útiles en áreas a priori alejadas de la geometría algebraica. El ejemplo concreto de la estadística se estudia en [SS19, Chapter 3].

El trabajo está hecho desde la perspectiva de un alumno de cuarto de carrera, que ni siquiera había terminado de cursar todas las asignaturas del Departamento de Álgebra cuando comenzó, y hemos intentado que este lo más autocontenido posible, aunque algunos resultados han tenido que ser omitidos o bien porque serían dignos de un trabajo de fin de grado por sí solos, o bien por ser resultados auxiliares, necesarios, pero que no entraban dentro del temario del trabajo. La intención ha sido que no sean necesarios conocimientos previos sobre  $D$ -módulos, un estudiante de último curso de matemáticas debería conocer, o, al menos, haber estudiado durante la carrera, la mayoría de los resultados sobre álgebra general del trabajo, y los resultados necesarios de geometría algebraica pueden encontrarse, por ejemplo, en [Har77]. Los resultados necesarios de análisis de Fourier han sido enunciados remitiéndose, para la prueba, a las referencias [Duo01] y [SS03].

## 2 | El álgebra de Weyl

En este primer capítulo introduciremos y definiremos el álgebra de Weyl, así como daremos una noción de grado y orden que necesitaremos en el próximo capítulo para poder definir las filtraciones y graduaciones. Por último, probaremos que es simple, y como consecuencia, que todo endomorfismo sobre el álgebra de Weyl es inyectivo.

### 2.1 Definición, forma normal y generadores

En esta sección nos centraremos en definir el concepto de álgebra de Weyl, la definiremos como un subálgebra del espacio de endomorfismos de un cuerpo  $K$ , y finalizaremos definiéndola alternativamente mediante sus generadores y las relaciones que verifican entre sí. Además, estudiaremos su forma canónica y caracterizaremos su base como  $K$ -espacio vectorial.

Sea  $K$  un cuerpo de característica cero y  $K[x] = K[x_1, \dots, x_n]$  el anillo de polinomios en  $n$  variables.

$K[x]$  es un espacio vectorial de dimensión infinita. Su álgebra de operadores se denota  $\text{End}_K(K[x])$ , en este anillo las operaciones serán la suma (+) y la composición de operadores ( $\cdot$ ). El *álgebra de Weyl* se definirá como una subálgebra de  $\text{End}_K(K[x])$ .

A la acción de los operadores del álgebra de Weyl la denotaremos por  $\bullet$ , para diferenciarla de la multiplicación.

**| Definición 2.1.1 ( $n$ -ésima álgebra de Weyl).** Sean  $\hat{x}_1, \dots, \hat{x}_n$  los operadores definidos por  $\hat{x}_i \bullet f = x_i f$  para  $f \in K[x]$  y los operadores  $\partial_1, \dots, \partial_n$  definidos por  $\partial_i \bullet f = \partial f / \partial x_i$ , que son operadores lineales de  $K[x]$ .

Definimos la  $n$ -ésima álgebra de Weyl,  $D_n$ , como la  $K$ -subálgebra de  $\text{End}_K(K[x])$  generada por los operadores  $\hat{x}_1, \dots, \hat{x}_n$  y  $\partial_1, \dots, \partial_n$ . La denotaremos como  $D_n =$

$K[\hat{x}_1, \dots, \hat{x}_n] \langle \partial_1, \dots, \partial_n \rangle$ ; en el caso  $n = 0$ , tomaremos  $D_0 = K$ .

Tal y como los hemos definido, los elementos de  $D_n$  son combinaciones lineales sobre  $K$  de monomios sobre los generadores. Pero dicha álgebra no es conmutativa, por lo que hay que tener cuidado al representar sus elementos. Esto puede comprobarse al aplicar el operador  $\partial_i \hat{x}_i$  a cualquier  $f \in K[x]$ , de donde obtendríamos que

$$(\partial_i \hat{x}_i) \bullet f = \partial_i \bullet (x_i f) = \frac{\partial x_i}{\partial x_i} f + x_i \frac{\partial f}{\partial x_i} = f + x_i \frac{\partial f}{\partial x_i},$$

es decir, tendríamos la igualdad  $\partial_i \hat{x}_i = 1 + \hat{x}_i \partial_i$ , donde 1 es el operador identidad. Usando conmutadores, podemos reescribir lo anterior como  $[\partial_i, \hat{x}_i] = 1$ , y más generalmente tenemos que  $[\partial_i, \hat{x}_j] = \delta_{ij}$ , donde  $\delta$  es la función *delta de Kronecker*. Por último, tenemos que  $[\partial_i, \partial_j] = [\hat{x}_i, \hat{x}_j] = 0$ , ya que estos operadores conmutan entre sí.

A partir de ahora denotaremos al operador  $\hat{x}_i$  como  $x_i$ , como se suele hacer normalmente, de esta manera tendremos que  $x_i$  actuará como una variable y como un operador. A su vez, en  $D_1$  denotaremos los generadores como  $x$  y  $\partial$  por comodidad.

Ahora vamos a construir una base para el álgebra de Weyl como  $K$ -espacio vectorial, y a esta base la llamaremos *base canónica*. Como es habitual, diremos que un elemento de  $D_n$  está expresado en forma canónica (o normal) si está expresado como combinación lineal de elementos de la base canónica. Esto nos permitirá comparar elementos de manera más simple.

Para simplificar la notación, definiremos un par de conceptos. Un *multi-índice* es una tupla  $\alpha \in \mathbb{N}^n$ , que se puede escribir como,  $\alpha = (\alpha_1, \dots, \alpha_n)$ . Escribiremos  $x^\alpha = x_1^{\alpha_1} \cdots x_n^{\alpha_n}$ , donde  $x = (x_1, \dots, x_n)$ . El grado de dicho monomio será  $|\alpha| = \sum_{i=1}^n \alpha_i$ .

Enunciaremos ahora una proposición de carácter técnico que será necesaria para construir una base sobre  $D_n$ .

**Proposición 2.1.2.**

(1) Sean  $\alpha, \beta \in \mathbb{N}^n$  y  $x = (x_1, \dots, x_n)$ , entonces

$$\partial^\beta \bullet x^\alpha = \begin{cases} \frac{\alpha!}{(\alpha-\beta)!} x^{\alpha-\beta} & \text{si } \beta \leq \alpha \\ 0 & \text{en otro caso} \end{cases}.$$

- Decimos que  $\beta \leq \alpha$  si para todo  $1 \leq i \leq n$  se verifica que  $\beta_i \leq \alpha_i$ , y a su vez  $\beta < \alpha$  si para todo  $1 \leq i \leq n$  se verifica  $\beta_i < \alpha_i$ .
- (2) Sea  $f \in K[x]$  y  $\beta \in \mathbb{N}^n$ . Se verifica la igualdad

$$\partial^\beta f = \sum_{\sigma \leq \beta} \binom{\beta}{\sigma} \partial^\sigma(f) \partial^{\beta-\sigma}.$$

- (3) Sean  $P \in D_n$ , con  $P = a_0 + \sum_{i=1}^n a_i \partial_i$ , donde los  $a_i \in K[x]$  para todo  $i$ , y  $f \in K[x]$ , se verifica que  $[P, f] = P \bullet f$ .

*Demostración.*

- (1) Supongamos que  $\alpha = (\alpha_1, \dots, \alpha_n)$  y  $\beta = (\beta_1, \dots, \beta_n)$ , entonces

$$\begin{aligned} \partial^\beta \bullet x^\alpha &= \frac{\partial^{|\beta|}}{\partial x_1^{\beta_1} \dots \partial x_n^{\beta_n}} x_1^{\alpha_1} \dots x_n^{\alpha_n} = \frac{\partial^{\beta_1}}{\partial x_1^{\beta_1}} x_1^{\alpha_1} \dots \frac{\partial^{\beta_n}}{\partial x_n^{\beta_n}} x_n^{\alpha_n} = \\ &= \frac{\alpha_1!}{(\alpha_1 - \beta_1)!} x_1^{\alpha_1 - \beta_1} \dots \frac{\alpha_n!}{(\alpha_n - \beta_n)!} x_n^{\alpha_n - \beta_n} = \frac{\alpha!}{(\alpha - \beta)!} x^{\alpha - \beta}. \end{aligned}$$

Si hubiera algún  $i \in \{1, \dots, n\}$  tal que  $\beta_i > \alpha_i$ , entonces  $\frac{\partial^{\beta_i}}{\partial x_i^{\beta_i}} x_i^{\alpha_i} = 0$  por lo que

$\partial^\beta \bullet x^\alpha = 0$ . En particular, también tenemos que si  $\alpha = \beta$  entonces  $\partial^\beta \bullet x^\alpha = \alpha!$

- (2) Probémoslo por inducción sobre  $n$ . El caso  $n = 1$  lo veremos por inducción sobre  $\beta$ . Si  $n = 1$  y  $\beta = 1$ , entonces

$$\begin{aligned} \partial f &= \partial \sum_{j=0}^{\deg(f)} a_j x^j = \sum_{j=0}^{\deg(f)} a_j \partial x^j \stackrel{(*)}{=} \sum_{j=0}^{\deg(f)} a_j (x^j \partial + j x^{j-1}) = \\ &= \left( \sum_{j=0}^{\deg(f)} a_j x^j \right) \partial + \sum_{j=0}^{\deg(f)} a_j j x^{j-1} = f \partial + \partial(f) = \sum_{k=0}^1 \binom{1}{k} \partial^k(f) \partial^{1-k}. \end{aligned}$$

(\*) Se obtiene al aplicar las propiedades del primer apartado inductivamente  $\partial x^j = (\partial x) x^{j-1} = (x \partial + 1) x^{j-1} = x(\partial x) x^{j-2} + x^{j-1} = \dots = x^j \partial + j x^{j-1}$ .

Supongamos que se verifica que  $\partial^\beta f = \sum_{\sigma \leq \beta} \binom{\beta}{\sigma} \partial^\sigma(f) \partial^{\beta-\sigma}$  para todo  $\beta$  menor que cierto  $k$  y sea  $\beta = k$ .

$$\begin{aligned} \partial^k f &= \partial(\partial^{k-1} f) = \partial \sum_{i \leq k-1} \binom{k-1}{i} \partial^i(f) \partial^{k-1-i} = \\ &= \sum_{i \leq k-1} \binom{k-1}{i} \partial(\partial^i(f) \partial^{k-1-i}) = \sum_{i \leq k-1} \binom{k-1}{i} (\partial^i(f) \partial^{k-i} + \partial^{i+1}(f) \partial^{k-1-i}) = \end{aligned}$$

$$= \sum_{i \leq k-1} \binom{k-1}{i} \partial^i(f) \partial^{k-i} + \sum_{i \leq k} \binom{k-1}{i-1} \partial^i(f) \partial^{k-i} = \sum_{i \leq k} \binom{k}{i} \partial^i(f) \partial^{k-i}.$$

Veamos un caso intermedio que hará mucho más clara la idea para el caso general, veamos qué ocurre cuando  $n = 2$ . Sean  $\alpha, \beta \in \mathbb{N}$  y  $\tau = (\alpha, \beta) \in \mathbb{N}^2$ , entonces

$$\begin{aligned} \partial^\tau f &= \partial_1^\alpha \partial_2^\beta f = \partial_1^\alpha \sum_{k=0}^{\beta} \binom{\beta}{k} \partial_2^k(f) \partial_2^{\beta-k} = \sum_{k=0}^{\beta} \binom{\beta}{k} \partial_1^\alpha (\partial_2^k(f) \partial_2^{\beta-k}) = \\ &= \sum_{k=0}^{\beta} \binom{\beta}{k} \sum_{l=0}^{\alpha} \binom{\alpha}{l} (\partial_1^l \partial_2^k)(f) \partial_1^{\alpha-l} \partial_2^{\beta-k} = \sum_{k=0}^{\beta} \sum_{l=0}^{\alpha} \binom{\beta}{k} \binom{\alpha}{l} (\partial_1^l \partial_2^k)(f) \partial_1^{\alpha-l} \partial_2^{\beta-k} = \\ &= \sum_{\sigma \leq \tau} \binom{\tau}{\sigma} \partial^\sigma(f) \partial^{\tau-\sigma}. \end{aligned}$$

Veamos ahora el caso para  $n$  cualquiera, sea  $\beta \in \mathbb{N}^n$ , no nulo, y sean  $\beta_1, \dots, \beta_n$  sus componentes, entonces

$$\partial^\beta f = \partial_1^{\beta_1} \dots \partial_n^{\beta_n} f,$$

y ahora podemos aplicar sucesivamente el caso anterior, de forma que obtendríamos

$$\begin{aligned} \partial_1^{\beta_1} \dots \partial_n^{\beta_n} f &= \partial_1^{\beta_1} \dots \partial_{n-1}^{\beta_{n-1}} \sum_{j \leq \beta_n} \binom{\beta_n}{j} \partial_n^j(f) \partial_n^{\beta_n-j} = \\ &= \partial_1^{\beta_1} \dots \partial_{n-2}^{\beta_{n-2}} \sum_{(i,j) \leq (\beta_{n-1}, \beta_n)} \binom{(\beta_{n-1}, \beta_n)}{(i,j)} (\partial_{n-1}^i \partial_n^j)(f) \partial_{n-1}^{\beta_{n-1}-i} \partial_n^{\beta_n-j}, \end{aligned}$$

repetiendo este proceso con todas las  $\partial_k^{\beta_k}$  concluimos el resultado.

- (3) Sea  $f \in K[x]$ , y expresémoslo como  $f = \sum_{\alpha=0}^{\deg(f)} a_\alpha x^\alpha$ , con  $a_\alpha \in K$ . Sea  $P = x^\beta \partial_i$  con  $\beta \in \mathbb{N}^n$  y  $1 \leq i \leq n$ .

$$[x^\beta \partial_i, f] = x^\beta \partial_i f - f x^\beta \partial_i = x^\beta \partial_i f - x^\beta f \partial_i = x^\beta [\partial_i, f],$$

por lo que solo tenemos que probar que el enunciado es cierto para  $P' = \partial_i$ :

$$[\partial_i, f] = \partial_i \sum_{\alpha=0}^{\deg(f)} a_\alpha x^\alpha - \left( \sum_{\alpha=0}^{\deg(f)} a_\alpha x^\alpha \right) \partial_i = \sum_{\alpha=0}^{\deg(f)} a_\alpha (\partial_i x^\alpha - x^\alpha \partial_i).$$

Ahora, aprovechando que los  $\partial_i$  y  $x_j$  conmutan siempre que  $i \neq j$ , y que  $x_i \partial_i = \partial_i x_i - 1$ , llegamos a que

$$\sum_{\alpha} a_\alpha (\partial_i x^\alpha - x^\alpha \partial_i) = \sum_{\alpha} a_\alpha (x_1^{\alpha_1} \dots \partial_i x_i^{\alpha_i} \dots x_n^{\alpha_n} - x_1^{\alpha_1} \dots x_i^{\alpha_i} \partial_i \dots x_n^{\alpha_n}) =$$



$$\begin{aligned}
\sum_{\alpha} a_{\alpha} x_1^{\alpha_1} \cdots (\partial_i x_i^{\alpha_i} - x_i^{\alpha_i} \partial_i) \cdots x_n^{\alpha_n} &= \sum_{\alpha} a_{\alpha} x_1^{\alpha_1} \cdots (x_i^{\alpha_i} \partial_i + \alpha_i x_i^{\alpha_i-1} - x_i^{\alpha_i} \partial_i) \cdots x_n^{\alpha_n} \\
&= \sum_{\alpha} a_{\alpha} x_1^{\alpha_1} \cdots (\alpha_i x_i^{\alpha_i-1}) \cdots x_n^{\alpha_n} = \partial f / \partial x_i,
\end{aligned}$$

con lo que por linealidad quedaría probada la proposición.

La siguiente proposición será suficiente para poder caracterizar una base de  $D_n$  como  $K$ -espacio vectorial.

**Proposición 2.1.3.** El conjunto  $B = \{x^{\alpha} \partial^{\beta} : \alpha, \beta \in \mathbb{N}^n\}$  es una base de  $D_n$  como  $K$ -espacio vectorial.

**Demostración.** Veamos que todo elemento de  $D_n$  se puede expresar como combinación lineal de elementos de  $B$ . Consideremos un elemento expresado en los generadores de  $D_n$ , como los  $\partial_i$  y los  $x_j$  conmutan siempre que  $i \neq j$  y tenemos que  $[\partial_i, x_i] = 1$ , aplicando esta última propiedad las veces necesarias, podemos llevar todas las potencias  $x_i$  a la izquierda de las  $\partial_i$ , con lo que será una combinación lineal de elementos de  $B$ .

Veamos ahora la unicidad. Consideremos una combinación lineal de elementos de  $B$ ,  $P = \sum_{\alpha, \beta \in \mathbb{N}^n} c_{\alpha\beta} x^{\alpha} \partial^{\beta}$ , veamos que si algún  $c_{\alpha\beta} \neq 0$  entonces  $P \neq 0$ . Dado que  $P$  es un operador lineal, esto es equivalente a decir que existe un  $f$  tal que  $P \cdot f \neq 0$ . Sea  $\sigma \in \mathbb{N}^n$  un multi-índice tal que  $c_{\alpha\sigma} \neq 0$  para algún  $\alpha$ , pero tal que  $c_{\alpha\beta} = 0$  para todo  $\beta < \sigma$ . Entonces, aplicando la Proposición 2.1.2, tenemos que

$$P \cdot x^{\sigma} = \left( \sum_{\alpha, \beta \in \mathbb{N}^n} c_{\alpha\beta} x^{\alpha} \partial^{\beta} \right) \cdot x^{\sigma} = \left( \sum_{\alpha, \beta \in \mathbb{N}^n, \beta \leq \sigma} c_{\alpha\beta} x^{\alpha} \partial^{\beta} \right) \cdot x^{\sigma} = \sigma! \sum_{\alpha \in \mathbb{N}^n} c_{\alpha\sigma} x^{\alpha},$$

ahora bien, por construcción hemos tomado  $\sigma$  tal que para algún  $\alpha$  tenemos que  $c_{\alpha\sigma} \neq 0$ , entonces  $P \neq 0$ .

Sean ahora  $P, P'$  dos operadores lineales distintos, expresados como combinación lineal de los elementos de  $B$ , que verifiquen que para todo  $f$ ,  $P \cdot f = P' \cdot f$ . Esta condición se puede expresar como,  $P \cdot f - P' \cdot f = 0$ , que es equivalente a decir que  $(P - P') \cdot f = 0$  para todo  $f$ , ya que son operadores lineales, lo cual implica que  $P - P' = 0$ . Si ahora tomamos su expresión como combinación lineal de elementos de  $B$ , llegamos a que  $\sum_{\alpha, \beta \in \mathbb{N}^n} (c_{\alpha\beta} - d_{\alpha\beta}) x^{\alpha} \partial^{\beta} = 0$ , para ciertos  $c_{\alpha\beta}, d_{\alpha\beta} \in K$ , que es

verdad si y solo si  $c_{\alpha\beta} = d_{\alpha\beta}$  para todo  $\alpha, \beta$ , por lo que las dos representaciones son iguales, y la combinación lineal es única. |

Cuando los elementos están expresados como combinación lineal de los elementos de esta base  $B$ , se dirán que están en forma canónica o normal. Es un buen ejercicio para ver cómo trabajar con elementos que no estén en base canónica lo siguiente:

**Observación 2.1.4.** Consideremos encontrar la forma canónica de los siguientes elementos de  $D_3$ .

- (1)  $\partial_1^2 x_1^2$
- (2)  $\partial_2^3 x_1 \partial_3 x_3 + x_3 \partial_1 x_1$
- (3)  $\partial_1 x_1 \partial_2 \partial_3 x_3 x_2$
- (4)  $\partial_1^3 x_1^2 + \partial_2^2 x_2^3$

Pueden comprobarse los resultados usando el programa Macaulay2. Primero se deberá definir el álgebra de Weyl  $D_3$  con el comando:

```
D = CC[x1,x2,x3,d1,d2,d3, WeylAlgebra => {x1=>d1,x2=>d2,x3=>d3}].
```

A partir de ahí es simple, Macaulay2 por defecto devuelve la forma canónica de los elementos. Por ejemplo, si introducimos el cuarto ejemplo, es decir,  $d_1^3 x_1^2 + d_2^2 x_2^3$ , obtenemos  $x_1^2 d_1^3 + x_2^3 d_2^2 + 6 x_1 d_1^2 d_2 + 6 x_2 d_1 d_2^2 + 6 x_1 x_2 d_1 d_2$ .

A continuación definiremos el álgebra de Weyl mediante sus generadores y las relaciones que estos verifican entre sí. Más concretamente, el álgebra de Weyl es un cociente del álgebra libre de  $2n$  generadores. El ideal por el que hacemos cociente es el que viene generado por las relaciones naturales de  $D_n$ .

Antes de continuar, vamos a recordar la definición de  $K$ -álgebra libre.

**| Definición 2.1.5 ( $K$ -álgebra libre).** Una  $K$ -álgebra libre generada por  $\{x_1, \dots, x_n\}$  es el conjunto de expresiones finitas  $\sum_{\alpha \in \mathbb{N}^n} a_\alpha x^\alpha$ , donde  $x^\alpha = x_1^{\alpha_1} \cdots x_n^{\alpha_n}$  y  $a_\alpha = (a_{\alpha_1}, \dots, a_{\alpha_n}) \in K^n$ , definiendo el producto de dos monomios como la yuxtaposición de estos. Denotaremos las  $K$ -álgebras libres como  $K\{x_1, \dots, x_n\}$ .

Ahora podemos definir el homomorfismo sobreyectivo  $\phi : K\{z_1, \dots, z_{2n}\} \rightarrow D_n$  dado por  $\phi(z_i) = x_i$  y  $\phi(z_{i+n}) = \partial_i$  para  $i = 1, \dots, n$ . Sea ahora  $J$  el ideal bilátero de  $K\{z_1, \dots, z_{2n}\}$  generado por  $[z_{i+n}, z_i] = 1$  para  $i = 1, \dots, n$  y  $[z_i, z_j] = 0$  para

$j \neq i + n$  y  $1 \leq i, j \leq 2n$ . Es directo de las relaciones que ya hemos visto que se verifican en  $D_n$  que  $J \subseteq \ker(\phi)$ , por lo que  $\phi$  induce un homomorfismo de  $K$ -álgebras  $\hat{\phi} : K\{z_1, \dots, z_{2n}\}/J \rightarrow D_n$ . Probemos que, en realidad, este homomorfismo es un isomorfismo.

**Proposición 2.1.6.** El homomorfismo  $\hat{\phi}$  que acabamos de definir es un isomorfismo.

**Demostración.** Sabemos que los monomios en  $z_i + J$  forman un sistema generador de  $K\{z_1, \dots, z_{2n}\}/J$  como  $K$ -espacio vectorial, y queremos ver que forman una base. Para ver esto, tengamos en cuenta que las imágenes de dichos monomios forman una base de  $D_n$  como  $K$ -álgebra y  $\hat{\phi}$  es un homomorfismo de  $K$ -álgebras. Las imágenes de los monomios en  $z_i$  en  $D_n$  son linealmente independientes, entonces los  $z_i$  deberán ser linealmente independientes en  $K\{z_1, \dots, z_{2n}\}/J$ , por lo que el homomorfismo  $\hat{\phi}$  será un isomorfismo de  $K$ -espacios vectoriales, y por consiguiente, será un isomorfismo de  $K$ -álgebras. |

## 2.2 Grado y orden

En esta sección daremos las definiciones de grado y orden de un operador de  $D_n$ , que nos servirán para ver en el próximo capítulo que el álgebra de Weyl admite una filtración por ambas relaciones de orden. Además, estas nociones nos permitirán probar que  $D_n$  es un dominio de integridad.

Comencemos tomando un  $P \in D_n$ . El grado de  $P$  es la mayor longitud del multi-índice  $(\alpha, \beta) \in \mathbb{N}^n \times \mathbb{N}^n$  tal que  $x^\alpha \partial^\beta$  aparece (con coeficiente no nulo) en la forma normal de  $P$ . Se denotará por  $\deg(P)$ , como es habitual. Tomaremos, por convenio, que  $\deg(0) = -\infty$ .

Algunas propiedades que nos resultarán de especial interés en el trabajo son las siguientes.

| **Teorema 2.2.1.** El grado satisface las siguientes propiedades para  $P, P' \in D_n$ :

- (1)  $\deg(P + P') \leq \max\{\deg(P), \deg(P')\}$ .
- (2)  $\deg(P P') = \deg(P) + \deg(P')$ .
- (3)  $\deg([P, P']) \leq \deg(P) + \deg(P') - 2$ .

**Demostración.** (1) Dado que si  $P$  y  $P'$  están en forma canónica entonces  $P + P'$  también, si sus grados son distintos el término de mayor grado no se verá afectado al

sumar, por lo que  $\deg(P + P') = \max\{\deg(P), \deg(P')\}$ .

Ahora bien, si  $\deg(P) = \deg(P')$  entonces podrían ocurrir dos cosas, o bien los términos de mayor grado se anulan entre sí, por lo que  $\deg(P + P') \leq \deg(P) = \max\{\deg(P), \deg(P')\}$  o bien los términos de mayor grado no se anulan, de forma que  $\deg(P + P') = \deg(P) = \max\{\deg(P), \deg(P')\}$ .

Veremos (2) y (3) simultáneamente, por inducción sobre  $\deg(P) + \deg(P')$ . Si cualquiera de los dos fuera 0 ( $P$  o  $P'$  serían constantes) el resultado es obvio, ya que  $\deg(P P') = \deg(P) = \deg(P) + \deg(P')$ , suponiendo que  $P'$  es la constante. Además,  $\deg([P, P']) = \deg(0) = -\infty$ , ya que el producto por constantes es conmutativo.

Supongamos que  $\deg(P), \deg(P') \geq 1$  y supongamos que el resultado es cierto siempre que  $\deg(P) + \deg(P') < k$  para cierto  $k \in \mathbb{N}$ . Sean entonces  $P, P' \in D_n$  tales que  $\deg(P) + \deg(P') = k$ . Por (1), que ya está probado, sabemos que el grado de  $P P'$  vendrá dado por el producto de los monomios de mayor grado de  $P$  y  $P'$ , por lo que para probar (2) y (3) será suficiente probar que se verifica para monomios.

Supongamos primero que  $P = \partial^\beta$  y  $P' = x^\alpha$  con  $|\alpha| + |\beta| = k$ . Como  $\deg(P) \geq 1$  existirá  $i \in \{1, \dots, n\}$  tal que  $\beta_i \neq 0$ , lo que nos permite escribir

$$\begin{aligned} [\partial^\beta, x^\alpha] &= \partial^\beta x^\alpha - x^\alpha \partial^\beta = \partial_i(\partial^{\beta-e_i} x^\alpha) - (x^\alpha \partial_i) \partial^{\beta-e_i} = \\ &= \partial_i(\partial^{\beta-e_i} x^\alpha) - \partial_i x^\alpha \partial^{\beta-e_i} + \partial_i x^\alpha \partial^{\beta-e_i} - (x^\alpha \partial_i) \partial^{\beta-e_i} = \\ &= \partial_i(\partial^{\beta-e_i} x^\alpha - x^\alpha \partial^{\beta-e_i}) + (\partial_i x^\alpha - x^\alpha \partial_i) \partial^{\beta-e_i} = \partial_i[\partial^{\beta-e_i}, x^\alpha] + [\partial_i, x^\alpha] \partial^{\beta-e_i}. \end{aligned}$$

Como  $|\beta - e_i| + |\alpha| < k$ , entonces, por la hipótesis de inducción,  $\deg([\partial^{\beta-e_i}, x^\alpha]) \leq \deg(\partial^{\beta-e_i}) + \deg(x^\alpha) - 2 = |\beta| + |\alpha| - 3$  y además  $\deg([\partial_i, x^\alpha]) \leq \deg(\partial_i) + \deg(x^\alpha) - 2 = |\alpha| - 1$ . Usando ahora la hipótesis de inducción otra vez,  $\deg(\partial_i[\partial^{\beta-e_i}, x^\alpha]) = \deg(\partial_i) + \deg([\partial^{\beta-e_i}, x^\alpha]) \leq |\beta| + |\alpha| - 2$  y  $\deg([\partial_i, x^\alpha] \partial^{\beta-e_i}) = \deg([\partial_i, x^\alpha]) + \deg(\partial^{\beta-e_i}) \leq |\beta| + |\alpha| - 2$ . Por lo que aplicando (1) tenemos que  $\deg([\partial^\beta, x^\alpha]) \leq |\beta| + |\alpha| - 2$ , como queríamos probar.

Por último, dado que  $[\partial^\beta, x^\alpha] = \partial^\beta x^\alpha - x^\alpha \partial^\beta$ , entonces  $\partial^\beta x^\alpha = [\partial^\beta, x^\alpha] + x^\alpha \partial^\beta$ , por lo que por (1) tenemos que  $\deg(\partial^\beta x^\alpha) \leq \max\{\deg([\partial^\beta, x^\alpha]), \deg(x^\alpha \partial^\beta)\}$ , pero dado que  $\deg([\partial^\beta, x^\alpha]) \leq |\beta| + |\alpha| - 2 \leq |\beta| + |\alpha| = \deg(x^\alpha \partial^\beta)$ , podemos concluir que  $\deg(\partial^\beta x^\alpha) = |\beta| + |\alpha|$ .

Sean ahora  $P = x^\sigma \partial^\beta$  y  $P' = x^\alpha \partial^\mu$ . Supongamos que  $|\alpha|$  o  $|\beta|$  sean no nulos, cambiando los nombres de los operadores si es necesario. Ya hemos visto que  $\partial^\beta x^\alpha = x^\alpha \partial^\beta + Q$ , donde  $Q = [\partial^\beta, x^\alpha]$  con  $\deg(Q) \leq |\alpha| + |\beta| - 2$ , por lo que  $P P' = (x^\sigma \partial^\beta)(x^\alpha \partial^\mu) =$

$x^\sigma \partial^\beta x^\alpha \partial^\mu = x^\sigma (\partial^\beta x^\alpha) \partial^\mu = x^\sigma (x^\alpha \partial^\beta + Q) \partial^\mu = x^{\sigma+\alpha} \partial^{\beta+\mu} + x^\sigma Q \partial^\mu$ . Por hipótesis tenemos que  $\deg(x^\sigma Q \partial^\mu) = \deg(x^\sigma) + \deg(Q) + \deg(\partial^\mu) \leq |\sigma| + |\beta| + |\alpha| - 2 + |\mu| \leq |\sigma + \alpha| + |\beta + \mu| = \deg(x^{\sigma+\alpha} \partial^{\beta+\mu})$ , así  $\deg(PP') = \deg(x^{\sigma+\alpha} \partial^{\beta+\mu}) = |\sigma + \alpha| + |\beta + \mu| = |\sigma| + |\beta| + |\alpha| + |\mu| = |\sigma + \beta| + |\alpha + \mu| = \deg(P) + \deg(P')$  como queríamos probar.

Además hemos probado que  $PP' = x^{\sigma+\alpha} \partial^{\beta+\mu} + Q_1$  con  $\deg(Q_1) \leq \deg(P) + \deg(P') - 2$  y a su vez  $P'P = x^{\sigma+\alpha} \partial^{\beta+\mu} + Q_2$  con  $\deg(Q_2) \leq \deg(P) + \deg(P') - 2$ , por lo que  $[P, P'] = PP' - P'P = Q_1 - Q_2$ , de donde concluimos que  $\deg([P, P']) = \deg(Q_1 - Q_2) \leq |\sigma| + |\beta| + |\alpha| + |\mu| - 2 = \deg(P) + \deg(P') - 2$ . |

Como corolario a este teorema es inmediato obtener que  $D_n$  es un dominio de integridad.

**Corolario 2.2.2.** El álgebra de Weyl  $D_n$  es un dominio de integridad.

**Demostración.** Que sea un dominio significa que no existan divisores de cero. Supongamos que no lo sea, entonces  $\exists P, P' \in D_n$  tales que  $-\infty < \deg(P), \deg(P')$ , que verifican que  $\deg(PP') = -\infty$ . Pero por el Teorema 2.2.1 tenemos que  $\deg(PP') = \deg(P) + \deg(P') > -\infty$ , con lo que llegaríamos a una contradicción. |

Además del orden dado por el grado de un operador, podemos definir un orden que nos resultará bastante interesante, en el cual solo tendremos en cuenta el grado sobre las  $\partial_i$ . Más formalmente, tenemos la siguiente definición.

**Definición 2.2.3 (Orden de un operador sobre  $D_n$ ).** Sea  $P = \sum_{\alpha, \beta \in \mathbb{N}^n} f_{\alpha\beta} x^\alpha \partial^\beta = \sum_{\beta \in \mathbb{N}^n} g_\beta(x) \partial^\beta$  un operador lineal de  $D_n$ , el orden de  $P$  es el máximo  $|\beta|$  tal que  $g_\beta$  no es nulo, es decir

$$\text{ord}(P) = \max_{\beta \in \mathbb{N}^n} \{|\beta| : g_\beta \neq 0\}.$$

Por convenio tomaremos que  $\text{ord}(0) = -\infty$ .

Pensando ya en las filtraciones que estudiaremos en el capítulo 3, podemos definir los conjuntos de los operadores de  $D_n$  con orden menor o igual que  $r$ , que denotaremos por  $F_r$ .

Al igual que vimos en el caso del grado, el orden también verifica algunas propiedades interesantes.

**Teorema 2.2.4.** Sean  $P, P' \in D_n$ , el orden verifica que:

- (1)  $\text{ord}(P + P') \leq \max\{\text{ord}(P), \text{ord}(P')\}$ .
- (2)  $\text{ord}(PP') = \text{ord}(P) + \text{ord}(P')$ .
- (3)  $\text{ord}([P, P']) \leq \text{ord}(P) + \text{ord}(P') - 1$ .

**Demostración.** La propiedad (1) se prueba de manera análoga a como se hizo en el Teorema 2.2.1. Por tanto, solo deberemos modificar la demostración de las otras dos propiedades, se prueban por inducción sobre  $\text{ord}(P) + \text{ord}(P')$  siguiendo los mismos pasos que en la prueba del Teorema 2.2.1 (2) y (3), teniendo en cuenta el cambio de hipótesis. |

## 2.3 Ideales

En esta última sección aprovecharemos los contenidos desarrollados en las dos primeras para probar que el álgebra de Weyl es simple y que todo endomorfismo sobre  $D_n$  es inyectivo.

Recordemos que un anillo que solo posee al ideal nulo como ideal bilátero propio se dirá *simple*. Un anillo conmutativo simple es un cuerpo, pero este no es el caso del álgebra de Weyl, ya que esta no es conmutativa.

| **Teorema 2.3.1.** *El álgebra de Weyl es simple.*

**Demostración.** Supongamos, por reducción al absurdo, que  $I$  es un ideal no nulo bilátero de  $D_n$ . Sea  $I \ni P \neq 0$  de grado minimal. Si  $P$  es de grado 0, entonces  $I = D_n$ . Veamos el caso en el que  $\deg(P) = k > 0$  y lleguemos a una contradicción.

Sea  $(\alpha, \beta)$  un multi-índice de longitud  $k$ . Si  $x^\alpha \partial^\beta$  es un término de  $P$  y existe un  $i \leq n$  tal que  $\beta_i \neq 0$ , entonces  $\deg([x_i, P]) \leq \deg(x_i) + \deg(P) - 2 = \deg(P) - 1 = k - 1$ , y es no nulo, ya que como los  $x_i$  y  $\partial_i$  no conmutan, y  $\beta_i$  no es nulo, no se anulará el conmutador. Pero dado que  $I$  es ideal a ambos lados, entonces  $x_i P, P x_i \in I$  por lo que  $[x_i, P] \in I$ , pero  $\deg([x_i, P]) \leq k - 1 < k = \deg(P)$ , que contradice el hecho de que  $P$  sea minimal. Por tanto  $\beta = 0$ , pero como  $k > 0$  entonces existe  $j \leq n$  tal que  $\alpha_j \neq 0$ , entonces tomando ahora  $[\partial_j, P]$  y argumentando de manera análoga llegamos a que  $\deg([\partial_j, P]) \leq k - 1 < k = \deg(P)$ , con  $[\partial_j, P] \in I$  no nulo, que vuelve a contradecir la minimalidad de  $P$ . |

Como consecuencia directa de que  $D_n$  no posea ideales propios no nulos biláteros, obtenemos el siguiente corolario.

**Corolario 2.3.2.** Todo endomorfismo de  $D_n$  es inyectivo.

**Demostración.** El resultado es consecuencia directa del hecho de que el núcleo de un endomorfismo de  $D_n$  es un ideal bilátero. Esto es fácil de ver: sea  $\varphi \in \text{End}(D_n)$ , consideremos el  $\ker(\varphi)$  y sean  $f \in \ker(\varphi)$  y  $g \in D_n$ , tenemos:

1.  $\varphi(fg) = \varphi(f)\varphi(g) = 0 \cdot \varphi(g) = 0$ , por lo que  $fg \in \ker(\varphi)$ , pero
2.  $\varphi(gf) = \varphi(g)\varphi(f) = \varphi(g) \cdot 0 = 0$ , por lo que  $gf \in \ker(\varphi)$ .

Teniendo esto en cuenta, el resultado es inmediato, ya que como el núcleo de un endomorfismo es bilátero, y el único ideal bilátero de  $D_n$  es el nulo, entonces  $\ker(\varphi) = 0$  para todo  $\varphi \in \text{End}(D_n)$ . Con lo que podemos concluir que todo endomorfismo es inyectivo. |

Aunque  $D_n$  no tenga ideales biláteros no nulos, no es un anillo de división, es más, los únicos elementos que poseen inverso son las constantes. Esto es fácil de comprobar gracias al Teorema 2.2.1: sean  $P, P' \in D_n$  tales que  $PP' = 1$ , entonces  $\deg(PP') = \deg(1) = 0 = \deg(P) + \deg(P')$ , lo cual solo es verdad si  $P$  y  $P'$  fueran de grado 0, es decir, constantes. Por lo que todo operador no constante genera un ideal a la izquierda no trivial de  $D_n$ .

El álgebra de Weyl tampoco es un anillo de ideales principales a la izquierda. Como contraejemplo, se puede probar que:

**Observación 2.3.3.** Sea  $n \geq 2$ , el ideal a izquierda  $L$  de  $D_n$  generado por  $\partial_1, \dots, \partial_n$  no es principal.

**Demostración.** Sea  $n \geq 2$  y  $L = \langle \partial_1, \dots, \partial_n \rangle$ , entonces queremos ver que  $L = \langle P \rangle$ , para cierto  $P \in L \subset D_n$ . Supongamos que dicho  $P$  existe, dado que las  $\partial_i \in L$  para todo  $i$ , entonces existen  $F_i \in D_n$ , no nulos, tales que  $F_i P = \partial_i$  para  $1 \leq i \leq n$ , pero por el Teorema 2.2.1, como  $\deg(\partial_i) = 1 = \deg(F_i P) = \deg(F_i) + \deg(P)$ , pueden ocurrir dos cosas:

1.  $\deg(P) = 0$ , entonces  $P$  es una constante, por lo que  $L = D_n$ . Pero esto no es verdad, ya que  $x_i \notin L$  para  $1 \leq i \leq n$ . Si  $x_i \in L$ , existirían  $a_1, \dots, a_n \in K$  tales que  $x_i = \sum_{k=1}^n a_k \partial_k$ , pero esto implicaría que  $x_i \bullet x_i = \left(\sum_{k=1}^n a_k \partial_k\right) \bullet x_i = a_i \in K$ , lo cual es absurdo.
2.  $\deg(P) = 1$ , entonces  $\deg(F_i) = 0$ , es decir,  $F_i$  son constantes no nulas.

Entonces ya tenemos que el candidato a “generador” de  $L$  es de grado 1. Pero

$\partial_j = F_j P = F_j F_i^{-1} F_i P = F_j F_i^{-1} \partial_i$ , y esto es una contradicción porque las  $\partial_i$  son linealmente independientes sobre  $K$ . |

Por último, cabe mencionar que todo ideal a izquierda de  $D_n$  puede generarse por dos elementos. La prueba de esto excede la dificultad y el propósito del trabajo, pero una prueba puede encontrarse en [Sta78].



## 3 | Módulos sobre el álgebra de Weyl

En este capítulo estudiaremos los ejemplos más notables de módulos sobre el álgebra de Weyl, los anillos de polinomios y las funciones holomorfas. Estos casos nos permitirán caracterizar el espacio de soluciones de un sistema de ecuaciones diferenciales. Y terminaremos el capítulo definiendo los conceptos de graduación y filtración, viendo que  $D_n$  admite filtraciones y estudiando las proporcionadas por el grado y el orden que vimos anteriormente.

### 3.1 El caso de los anillos de polinomios y de funciones holomorfas

Esta sección se centrará en el estudio de los anillos de polinomios como  $D_n$ -módulos a la izquierda, así como del anillo de funciones holomorfas.

#### 3.1.1 Anillos de polinomios

Empezamos viendo el ejemplo de los anillos de polinomios. Hemos introducido el álgebra de Weyl como un subanillo del anillo de endomorfismos. Tenemos que  $D_n(K)$  es un subanillo de  $\text{End}(K[x])$ . Por esto, podemos deducir que el anillo de polinomios es un  $D_n$ -módulo a la izquierda, con las acciones definidas en la sección 2.1.

Recordemos algunas definiciones importantes. Sea  $R$  un anillo, se dice que un  $R$ -módulo es *irreducible* (o *simple*) si no tiene submódulos propios. Recordemos también que el anulador de un elemento  $u \in M$  sobre  $R$  es el ideal de  $R$  cuyos elementos anulan a  $u$ . Sea ahora  $M$  un  $R$ -módulo a la izquierda, se dice que un elemento  $u \in M$  es de *torsión* si su anulador sobre  $R$  es un ideal no nulo a la izquierda. Si todo elemento

de  $M$  es de torsión se dice que  $M$  es un *módulo de torsión*.

Veamos una caracterización del concepto de anillo de división que será de suma utilidad posteriormente.

**Lema 3.1.2.** Un dominio  $R$  es un anillo de división si y solo si no tiene ideales propios no nulos a la izquierda.

**Demostración.** Veámoslo por doble implicación.

- $\Leftarrow$  Sea  $z \in R$  no nulo, donde  $R$  no tiene ideales a la izquierda no triviales. De esta forma, si definimos el ideal  $I = \{rz : r \in R\}$ , este debe ser  $R$ , por lo que habrá un  $r \in R$  tal que  $rz = 1$ . Si esto fuera verdad tendríamos que  $(rz)r = r(zr) = r$ , es decir,  $r(zr - 1) = 0$ , de donde concluimos que  $zr = 1$  y queda probado que todo elemento no nulo posee inverso.
- $\Rightarrow$  Sea  $R$  un anillo de división y sea  $I \subset R$  un ideal no nulo, para todo elemento  $r$  de  $I$  hay un  $t \in R$  tal que  $tr = 1$ , pero esto implica que  $1 \in I$ , por lo que  $R = I$ . Podemos concluir que  $R$  no posee ideales propios no nulos.

I

Queremos probar que  $K[x]$  es un  $D_n$ -módulo irreducible de torsión, por lo que primero necesitaremos una caracterización de ser módulo de torsión.

**Lema 3.1.3.** Sea  $R$  un anillo y sea  $M$  un  $R$ -módulo irreducible a la izquierda.

- (1) Si  $0 \neq u \in M$ , entonces  $M \cong R/\text{ann}_R(u)$ .
- (2) Si  $R$  no es un anillo de división, entonces  $M$  es un módulo de torsión.

**Demostración.** Consideremos el homomorfismo de  $R$ -módulos  $\phi : R \rightarrow M$  definido por  $\phi(1) = u$ . Como  $u \neq 0$  y  $M$  es irreducible, entonces  $\phi$  es sobreyectivo. Si no fuera sobreyectivo, tendríamos que  $\phi(R) \subset M$  sería un submódulo propio de  $M$ , lo que contradice que  $M$  sea irreducible. Entonces  $\ker(\phi) = \{r \in R : \phi(r) = u \cdot r = 0\} = \text{ann}_R(u)$ , y gracias al primer teorema de isomorfía tenemos que  $R/\ker(\phi) = R/\text{ann}_R(u) \cong M$ , con lo que (1) queda probado.

Supongamos ahora que  $M$  no sea de torsión, es decir, que exista un  $u \in M$ , no nulo, tal que  $\text{ann}_R(u) = 0$ . Por (1) tendríamos que  $M \cong R$ , como  $M$  es irreducible,  $R$  también lo sería. Que  $R$  sea irreducible significa que todos sus ideales a la izquierda

son triviales, que será cierto si y solo si  $R$  es un anillo de división, por el lema anterior. Con esto hemos probado el contrarrecíproco de (2), con lo que la prueba concluye. |

Ya tenemos todos los ingredientes necesarios para poder demostrar que  $K[x]$  es un  $D_n$ -módulo irreducible de torsión.

**Proposición 3.1.4.**  $K[x]$  es un  $D_n$ -módulo irreducible de torsión. Además,

$$K[x] \cong D_n / D_n \langle \partial_1, \dots, \partial_n \rangle.$$

**Demostración.** Para empezar, como ya sabemos, las constantes son generadoras de  $K[x]$ . Sea ahora  $f \neq 0$  un polinomio y consideremos el submódulo  $D_n f \subseteq K[x]$ . Sea  $x_1^{i_1} \cdots x_n^{i_n}$  un monomio de máximo grado posible entre los que aparecen en  $f$  y sea  $a$  su coeficiente. Así,  $(\partial_1^{i_1} \cdots \partial_n^{i_n}) \cdot f = i_1! \cdots i_n! a$  es una constante no nula en el submódulo generado por  $f$ . Entonces  $D_n f = K[x]$ , por lo que  $K[x]$  es irreducible. Aplicando el Lema 3.1.3, como  $D_n$  no es un anillo de división, se tiene que  $K[x]$  es un módulo de torsión.

Dado que 1 es anulado por  $\partial_1, \dots, \partial_n$ , el ideal a la izquierda  $J = \langle \partial_1, \dots, \partial_n \rangle$  está contenido en su anulador. Veamos que en realidad esto es una igualdad (de hecho  $J$  es maximal). Sea  $P \in \text{ann}_{D_n}(1)$ ,  $P$  se puede escribir como  $f + Q$  donde  $Q \in J$  y  $f \in K[x]$ , por la forma normal de un operador. Dado que  $P \in \text{ann}_{D_n}(1)$ ,  $0 = P \cdot 1 = (f + Q) \cdot 1 = f \cdot 1 + Q \cdot 1 = f \cdot 1$ , ya que  $Q \in J \subset \text{ann}_{D_n}(1)$ , llegamos a la conclusión de que  $0 = f \cdot 1$ , lo que implica que  $f$  es nulo. De forma que  $P = Q \in J$ , y  $\langle \partial_1, \dots, \partial_n \rangle = \text{ann}_{D_n}(1)$ . El isomorfismo que buscamos es consecuencia directa del Lema 3.1.3. |

### 3.1.5 Funciones holomorfas

Ahora veremos el caso de las funciones holomorfas. Trabajaremos sobre  $\mathbb{C}$ , por lo que por comodidad, cuando escribamos  $D$ , nos referiremos a  $D(\mathbb{C})$ . Sea  $U \subseteq \mathbb{C}$  un abierto. El conjunto  $\mathcal{H}(U)$  de las funciones holomorfas definidas en  $U$  contiene al anillo de polinomios  $\mathbb{C}[z]$ . Veámoslo como un  $D$ -módulo a la izquierda, denotando los generadores de  $D$  como  $z$  y  $\partial = d/dz$ , extendiendo su acción usual a  $\mathcal{H}(U)$ .

Vamos a probar que  $\mathcal{H}(U)$  no es irreducible. Lo primero que necesitaremos es probar que  $e^x$  no es algebraica en  $\mathbb{C}(x)$ , es decir, que no existe  $G(x, y)$  en  $\mathbb{C}[x, y]$  no nulo tal que  $G(z, e^z)$  se anule para todo  $z$  de  $U$ .

**Proposición 3.1.6.** La función  $e^x$  no es algebraica sobre  $\mathbb{C}(x)$ .

**Demostración.** Llamemos  $y = e^x$ , que verifica que  $y' = y$ , y consideremos la extensión  $\mathbb{C}(x)(y)$  de  $\mathbb{C}(x)$ .

Veamos primero que  $y \notin \mathbb{C}(x)$ . Los elementos de este espacio son de la forma  $p/q$  con  $p, q \in \mathbb{C}(x)$ . Entonces,  $(p/q)' = \frac{p'q - pq'}{q^2}$ ; si tuviéramos que  $(p/q)' = p/q$  entonces por la igualdad de antes  $p'q - pq' = pq$ . Si nos fijamos en el grado de los polinomios de esta igualdad tenemos que el miembro de la izquierda tiene grado  $\deg(p) + \deg(q) - 1$  (o menor), mientras que el miembro de la derecha tiene grado  $\deg(p) + \deg(q)$ , por lo que no se puede dar la igualdad. Concluimos que  $y \notin \mathbb{C}(x)$ .

Supongamos ahora que  $y$  fuese algebraico sobre  $\mathbb{C}(x)$ , y sea  $f$  el polinomio mínimo de la extensión, es decir, el polinomio de menor grado que verifica que  $f(x, y) = y^n + a_{n-1}(x)y^{n-1} + \dots + a_1(x)y + a_0(x) = 0$ . Derivando respecto de  $x$ , y recordando que  $y' = y$ , llegamos a  $ny^n + (n-1)a_{n-1}(x)y^{n-1} + a'_{n-1}(x)y^{n-1} + \dots + a_1(x)y + a'_1(x)y + a'_0(x) = 0$ . Si ahora restamos  $n$  veces la primera expresión a la expresión de la derivada, obtenemos  $(a'_{n-1}(x) - a_{n-1}(x))y^{n-1} + \dots + a'_0(x) - na_0(x) = 0$ . Pero ya hemos visto que  $z = z'$  no tiene solución no nula en  $\mathbb{C}(x)$ , por lo que pueden ocurrir dos cosas, o bien  $a_{n-1} - a'_{n-1} \neq 0$ , con lo que tendríamos un polinomio de grado  $n-1$  que es anulado por  $y$ , lo que contradice la minimalidad de  $f$ , o bien  $a_{n-1} = a'_{n-1} = 0$ , por lo que pasaríamos al siguiente coeficiente de la derivada, hasta encontrar uno no nulo. El hecho de que exista uno no nulo viene de que si no fuese así, entonces  $f(x, y) = y^n$ , y este polinomio no anula a  $e^x$ , llegando a una contradicción. |

Una vez tengamos esto, ya estamos en disposición de probar que  $\mathcal{H}(U)$  no es irreducible.

**Proposición 3.1.7.**  $\mathcal{H}(U)$  no es irreducible.

**Demostración.** Razonemos por reducción al absurdo, supongamos que  $\mathcal{H}(U)$  es irreducible. Por 3.1.3,  $\mathcal{H}(U)$  será un módulo cíclico sobre  $D_n$ , llamemos  $u$  a su generador.

Supongamos en primer lugar que  $\text{ann}_{D_n}(u) = 0$ . En este caso, sea  $f$  un polinomio cualquiera en  $z$ , existen  $P \in D_n$  y  $N$  entero, tales que  $f = P \bullet u$  y  $\partial^N(f) = 0$ , de esta forma,  $(\partial^N P) \bullet u = 0$ , lo que contradice que  $\text{ann}_{D_n}(u) = 0$ .

Entonces, el anulador de  $u$  no es trivial. Veamos que la dimensión  $\mathcal{H}(U) \otimes_{\mathbb{C}[z]} \mathbb{C}(z)$  es finita. Como el anulador no es nulo, existirá un  $Q = \sum_{j=0}^{\text{ord}(Q)} Q_j \partial^j$  tal que  $Q \bullet u = 0$ .

Es este caso, dividiendo por  $Q_{\text{ord}(Q)}$ , podremos escribir  $\partial^{\text{ord}(Q)} \bullet u$  como combinación lineal, sobre  $\mathbb{C}(z)$ , de  $u, u', \dots, u^{(\text{ord}(Q))}$ , y, de igual forma, podríamos expresar el resto de derivadas de  $u$  como combinación de estas. Entonces, suponiendo que cualquier función holomorfa se puede expresar como  $P \bullet u$ , entonces será una combinación lineal de las primeras  $\text{ord}(Q)$  derivadas de  $u$ , es decir, la dimensión de  $\mathcal{H}(U) \otimes_{\mathbb{C}[z]} \mathbb{C}(z)$  será menor o igual que  $\text{ord}(Q)$ .

En este caso, si consideramos las potencias de  $e^z$ , son todas funciones holomorfas, por lo que tenemos una familia infinita, en un espacio vectorial de dimensión finita, lo que implica que debemos tener una relación de dependencia  $\mathbb{C}(z)$ -lineal entre cualquier cantidad finita de potencias de  $e^z$ . Pero eso no es más que una ecuación polinómica en  $e^z$  con coeficientes en  $\mathbb{C}(z)$ , es decir,  $e^z$  es algebraica, y esto no es cierto por la proposición 3.1.6. Podemos concluir que  $\mathcal{H}(U)$  no es irreducible. |

Por otro lado, un elemento de torsión de  $\mathcal{H}(U)$  es una función que satisface una ecuación diferencial con coeficientes polinómicos. Ejemplo de esto es la función  $e^z$ , que es solución de  $\partial - 1$ . Aún así,  $\mathcal{H}(U)$  no es un módulo de torsión. Vamos a ver que  $\exp(e^z)$  no satisface ninguna ecuación diferencial. Veamos primero un lema que va a ser útil en la prueba.

**Lema 3.1.8.** Sea  $h(z) = \exp(e^z) \in \mathcal{H}(U)$ . Para todo entero positivo  $m$  existe un polinomio  $F_m(x) \in \mathbb{C}[x]$  de grado  $m$  tal que

$$\partial^m(h) = F_m(e^z)h(z) = F_m(e^z)\exp(e^z).$$

**Demostración.** Tenemos que  $\partial(h) = \partial(\exp(e^z)) = e^z h(z)$ , por lo que el resultado es cierto para  $m = 1$  con  $F_1(z) = z$ . Supongamos por inducción que se verifica para cierto  $m \geq 1$ . Usando que  $\partial(\partial^m(h)) = \partial^{m+1}(h)$  y la hipótesis de inducción, llegamos a que

$$\partial^{m+1}(h) = \partial((F_m(e^z)h(z))) = F'_m(e^z)h(z)e^z + F_m(e^z)e^z h(z),$$

donde  $F'_m(z)$  es la derivada de  $F_m$ .

Tomando  $F_{m+1}(z) = F'_m(z)z + F_m(z)z$  obtenemos la formula deseada. Y además tenemos que  $\deg(F_{m+1}) = \max\{\deg(F'_m z), \deg(F_m z)\} = \deg(F_m z) = \deg(F_m) + \deg(z) = m + 1$ . |

Esto, junto con la proposición 3.1.6, nos bastará para probar que  $\exp(e^z)$  no satisface ninguna ecuación diferencial con coeficientes polinómicos.

**Proposición 3.1.9.** La función  $h(z) = \exp(e^z)$  no es un elemento de torsión del  $D$ -módulo  $\mathcal{H}(U)$ .

**Demostración.** Sea  $P = \sum_{i=0}^r f_i(z) \partial^i \in D$ , y asumamos que  $f_r \neq 0$ . Si  $0 = P \bullet h = \sum_{i=0}^r f_i(z) \partial^i(h)$ , usando el Lema 3.1.8, existen  $F_1, \dots, F_r$  tales que  $\partial^i(h) = F_i h$ , por lo que  $0 = P \bullet h = (\sum_{i=0}^r f_i(z) F_i(e^z))h$ , pero dado que  $h$  no se anula en  $\mathbb{C}$ , esto es equivalente a que  $\sum_{i=0}^r f_i(z) F_i(e^z) = 0$ . Si definimos ahora  $G(x, y) = \sum_{i=0}^r f_i(z) F_i(y)$ , donde los  $F_i$  son de grado  $i$ , y por tanto polinomios en  $y$  no nulos, tenemos que  $G(x, y) \neq 0$  y  $G(z, e^z) = 0$ . De ahí concluiríamos que  $e^z$  sería algebraica, con lo que llegaríamos a un absurdo, por la proposición 3.1.6. |

## 3.2 Soluciones de una ecuación diferencial

Ya estamos listos para justificar que la teoría de  $D_n$ -módulos ofrece un acercamiento algebraico a las ecuaciones diferenciales lineales. Sea  $P \in D_n$  un operador, podemos expresarlo como  $\sum_{\alpha} g_{\alpha} \partial^{\alpha}$  donde  $\alpha \in \mathbb{N}^n$  y  $g_{\alpha} \in K[x]$ . Este operador nos da la ecuación

$$P \bullet f = \sum_{\alpha \in \mathbb{N}^n} g_{\alpha} \partial^{\alpha}(f) = 0,$$

donde  $f$  será un polinomio o una función holomorfa en cierto abierto  $U \subseteq \mathbb{C}$  si  $K = \mathbb{C}$ , o, si  $K = \mathbb{R}$ , una función  $C^{\infty}$ . Más generalmente, si  $P_1, \dots, P_m \in D_n$  son operadores diferenciales, tenemos el sistema de ecuaciones diferenciales

$$P_1 \bullet f = \dots = P_m \bullet f = 0. \quad (3.1)$$

En esta sección queremos asociarle a este sistema un  $D_n$ -módulo finitamente generado de forma canónica. Esto permitirá dar una descripción puramente algebraica de las soluciones del sistema (3.1).

El  $D_n$ -módulo asociado al sistema es  $D_n / D_n \langle P_1, \dots, P_m \rangle$ , esta definición viene justificada por el próximo teorema. Una solución de (3.1) es un polinomio  $f \in K[x]$  que satisface  $P_i \bullet f = 0$  para todo  $i$ , y tenemos que el espacio de las soluciones de (3.1) es un  $K$ -espacio vectorial, dado por

$$\text{Sol}_{K[x]}(I) = \{m \in K[x] : P \bullet m = 0, \forall P \in I\},$$

donde  $I = \langle P_1, \dots, P_m \rangle$ , aunque podemos generalizar esta definición para  $M$ , un  $D_n$ -módulo cualquiera, como

$$\text{Sol}_M(I) = \{m \in M : P \bullet m = 0, \forall P \in I\}.$$

**| Teorema 3.2.1.** *El espacio vectorial de los polinomios solución del sistema (3.1) es isomorfo a  $\text{Hom}_{D_n}(D_n/I, K[x])$ . Pero más generalmente, si  $M$  es un  $D_n$ -módulo, tenemos el isomorfismo*

$$\text{Hom}_{D_n}(D_n/I, M) \cong \text{Sol}_M(I).$$

**Demostración.** Sea  $M$  un  $D_n$ -módulo. Para probar este resultado vamos a dar explícitamente dos homomorfismos,  $\sigma : \text{Sol}_M(I) \rightarrow \text{Hom}_{D_n}(D_n/I, M)$  y  $\tau : \text{Hom}_{D_n}(D_n/I, M) \rightarrow \text{Sol}_M(I)$ , veremos que son  $K$ -lineales e inversos uno del otro.

Empecemos viendo  $\sigma$ . Sea  $m \in \text{Sol}_M(I)$ , definimos la aplicación como  $\sigma(m) = \sigma_m \in \text{Hom}_{D_n}(D_n/I, M)$ , que es el homomorfismo que lleva la clase del 1 a  $m$ , es decir, para  $P + I \in D_n/I$ ,  $\sigma_m(P + I) = P \cdot \sigma_m(1) = P \cdot m \in M$ . Este homomorfismo verifica que si  $Q \in I$ , entonces  $\sigma_m(Q) = Q \cdot m = 0$ , ya que  $m \in \text{Sol}_M(I)$ .

Veamos que  $\sigma$  es  $K$ -lineal, es decir, tenemos que ver que  $\sigma(m + m') = \sigma(m) + \sigma(m')$  y que  $\sigma(\lambda m) = \lambda \sigma(m)$  para todo  $m, m' \in \text{Sol}_M(I)$  y  $\lambda \in K$ . Sea  $P + I \in D_n/I$ , tenemos que

$$\sigma_{m+m'}(P + I) = P \cdot (m + m') = P \cdot m + P \cdot m' = \sigma_m(P + I) + \sigma_{m'}(P + I).$$

Por último,  $\sigma_{\lambda m}(P + I) = \lambda P \cdot m = \lambda \sigma_m(P + I)$ , por lo que ya tenemos que  $\sigma$  es  $K$ -lineal.

Definamos ahora  $\tau$ . Sea  $\varphi \in \text{Hom}_{D_n}(D_n/I, M)$  un homomorfismo que mande la clase del 1 al elemento  $m \in M$ , definimos  $\tau(\varphi) = \varphi(1 + I) \in \text{Sol}_M(I)$ . Tenemos que comprobar que verdaderamente  $\varphi(1 + I) \in \text{Sol}_M(I)$ , sea  $P \in I$ ,

$$P \cdot \varphi(1 + I) = \varphi(P + I) = \varphi(0 + I) = 0.$$

Veamos ahora que  $\tau$  es  $K$ -lineal, sean  $\phi, \psi \in \text{Hom}_{D_n}(D_n/I, M)$  y  $\lambda \in K$ .

$$\tau(\phi + \psi) = (\phi + \psi)(1 + I) = \phi(1 + I) + \psi(1 + I) = \tau(\phi) + \tau(\psi),$$

de igual forma,

$$\tau(\lambda \phi) = (\lambda \phi)(1 + I) = \lambda \phi(1 + I) = \lambda \tau(\phi),$$

por lo que  $\tau$  es  $K$ -lineal.

Ya solo nos queda ver que son inversas la una de la otra. Sea  $\varphi \in \text{Hom}_{D_n}(D_n/I, M)$ .

$$(\sigma \circ \tau)(\varphi)(1 + I) = \sigma(\tau(\varphi))(1 + I) = \sigma(\varphi(1 + I))(1 + I) = \varphi(1 + I),$$

es decir, por  $D_n$ -linealidad,  $\sigma \circ \tau = \text{id}_{\text{Hom}_{D_n}(D_n/I, M)}$ . A su vez, para todo  $m \in \text{Sol}_M(I)$ ,

$$(\tau \circ \sigma)(m) = \tau(\sigma(m)) = \tau(\sigma_m) = \sigma_m(1 + I) = m,$$

por lo que  $\tau \circ \sigma = \text{id}_{\text{Sol}_M(I)}$ . Esto concluye la prueba. |

Hay que prestar atención al hecho de que  $\text{Hom}_{D_n}(D_n/I, K[x])$  no es un  $D_n$ -módulo, ni tampoco un  $K[x]$ -módulo, es “simplemente” un  $K$ -espacio vectorial. Es más, podría tener dimensión infinita.

### 3.3 Filtraciones y graduaciones

Estudiar anillos simples es muy complejo, dado que la mayoría de técnicas de estudio sobre anillos requieren de anillos biláteros, pero usando el concepto de orden que desarrollamos anteriormente, podemos estudiar más a fondo las propiedades del álgebra de Weyl.

En esta sección nos centraremos en introducir de forma general los conceptos de graduación y filtración, de cara a poder utilizarlos en el caso particular del álgebra de Weyl en la siguiente sección.

Los anillos de polinomios nos permiten definir una función grado. Nuestra idea es intentar generalizar y formalizar dicho concepto al caso de un álgebra, lo que nos lleva a la siguiente definición:

**Definición 3.3.1 ( $K$ -álgebra graduada).** Sea  $R$  una  $K$ -álgebra. Diremos que  $R$  es graduada si existen  $R_i$ ,  $i \in \mathbb{N}$ ,  $K$ -subespacios vectoriales tales que:

- (1)  $R = \bigoplus_{i \in \mathbb{N}} R_i$ ,
- (2)  $R_i \cdot R_j \subseteq R_{i+j}$ .

Llamamos componentes homogéneas de  $R$  a los  $R_i$  y sus elementos serán los elementos homogéneos de grado  $i$ . Si  $R_i = 0$  para  $i < 0$  diremos que la graduación es positiva.

Supondremos que las álgebras graduadas con las que trabajaremos serán de graduación positiva mientras no se diga lo contrario.

El ejemplo más importante de álgebra graduada es el anillo de polinomios  $K[x_1, \dots, x_n]$ , donde los monomios  $x_1^{k_1} \cdots x_n^{k_n}$ , con  $k_1 + \cdots + k_n = m$ , forman una base de la componente homogénea de grado  $m$ .



Los anillos graduados más importantes dentro de la geometría algebraica son los cocientes de anillos de polinomios. Sea  $R$  una  $K$ -álgebra graduada, un ideal bilátero  $I$  de  $R$  se dice *graduado* si  $I = \bigoplus_{i \geq 0} (I \cap R_i)$ ; por definición, los ideales graduados están generados por elementos homogéneos. La otra implicación también es cierta, si un ideal está generado por elementos homogéneos entonces será graduado.

Sea ahora  $S = \bigoplus_{i \geq 0} S_i$  otra  $K$ -álgebra graduada. Un homomorfismo de  $K$ -álgebras  $\phi : R \rightarrow S$  es graduado si  $\phi(R_i) \subseteq S_i$ , o equivalentemente, un homomorfismo graduado es aquel que preserva el grado.

**Proposición 3.3.2.** Sean  $R = \bigoplus_{i \geq 0} R_i$  y  $S = \bigoplus_{i \geq 0} S_i$  dos  $K$ -álgebras graduadas.

- (1) El núcleo de un homomorfismo de  $K$ -álgebras  $\phi : R \rightarrow S$  es un ideal graduado bilátero de  $R$ .
- (2) Si  $I$  es un ideal graduado bilátero de  $R$  entonces  $R/I$  es una  $K$ -álgebra graduada.

**Demostración.** Veamos primero (1), sea  $\phi$  un homomorfismo graduado, y sea  $a = a_0 \oplus \cdots \oplus a_s$  un elemento del núcleo de  $\phi$ , donde los  $a_i \in R_i$  para todo  $0 \leq i \leq s$ . Entonces, dado que  $\phi$  es un homomorfismo y además graduado (mantiene el grado), se verifica que

$$\phi(a) = \phi(a_0) \oplus \cdots \oplus \phi(a_s) = 0,$$

pero al ser cada elemento de grado distinto, cada uno de ellos debe ser nulo, es decir,  $a_i \in \ker(\phi)$  para  $0 \leq i \leq s$ . Por lo que el núcleo de  $\phi$  es graduado, y como ya habíamos visto, es un ideal bilátero, esto prueba (1).

Probemos ahora (2), sea  $I$  un ideal graduado bilátero. Podemos descomponer el cociente  $R/I$  como  $\bigoplus_{i \geq 0} (R_i / (I \cap R_i))$ . Para terminar de probar que es graduado debemos comprobar que  $(R_i / (I \cap R_i)) \cdot (R_j / (I \cap R_j)) \subseteq R_{i+j} / (I \cap R_{i+j})$ , pero esto es fácil de ver, ya que si  $a_i \in R_i$  y  $a_j \in R_j$ , entonces  $(a_i + I) \cdot (a_j + I) = a_i a_j + I$  se corresponde con un elemento de  $R_{i+j} / (I \cap R_{i+j})$  en la proyección, por lo que  $R/I$  es graduado. |

Esto nos brinda una manera de crear anillos graduados; para ciertos polinomios homogéneos  $F_1, \dots, F_s \in K[x]$ , la proposición anterior nos dice que  $K[x]/(F_1, \dots, F_n)$  es un anillo graduado.

Las álgebras graduadas nos permiten a su vez definir un concepto similar para los módulos.

**| Definición 3.3.3 (Módulo graduado).** Sea  $R = \bigoplus_{i \geq 0} R_i$  un  $K$ -álgebra graduada. Un  $R$ -módulo a la izquierda  $M$  se dirá graduado si existen  $K$ -espacios vectoriales  $M_i$  para  $i \geq 0$  tales que:

- (1)  $M = \bigoplus_{i \geq 0} M_i$ ,
- (2)  $R_i \cdot M_j \subseteq M_{i+j}$ .

Las  $M_i$  son las componentes homogéneas de grado  $i$  de  $M$ .

También podemos definir los conceptos de *submódulo graduado* y *homomorfismo graduado entre módulos*, de la misma forma que las definimos anteriormente. Sea  $R$  una  $K$ -álgebra graduada y sean  $M, M'$  dos  $R$ -módulos a la izquierda graduados. Un submódulo  $N$  de  $M$  se dirá graduado si  $N = \bigoplus_{i \geq 0} (N \cap M_i)$ . A su vez, un homomorfismo de  $R$ -módulos  $\theta : M \rightarrow M'$  es graduado si  $\theta(M_i) \subseteq M'_i$ , para todo  $i \geq 0$ .

Análogamente, se verifica que  $\ker(\theta)$  es un submódulo graduado y que el módulo cociente  $M/N$  es un  $R$ -módulo graduado; la prueba de este resultado sigue los mismos pasos que la demostración de la proposición 3.3.2.

Nuestra actual definición de grado de un operador no nos permite convertir al álgebra de Weyl en un álgebra graduada, ya que aunque el elemento  $\partial_1 x_1$  es homogéneo de grado 2, es igual a  $x_1 \partial_1 + 1$ , que no es homogéneo. Por ello recurrimos a las filtraciones. Empecemos dando una definición de filtración.

**| Definición 3.3.4 (Filtración).** Sea  $R$  una  $K$ -álgebra. Una familia  $\Gamma = \{\Gamma_i\}_{i \geq 0}$  de  $K$ -espacios vectoriales es una filtración de  $R$  si:

- (1)  $\Gamma_0 \subseteq \Gamma_1 \subseteq \cdots \subseteq R$ ,
- (2)  $R = \bigcup_{i \geq 0} \Gamma_i$ ,
- (3)  $\Gamma_i \cdot \Gamma_j \subseteq \Gamma_{i+j}$ .

Si un álgebra tiene una filtración se dirá que es un álgebra filtrada. Tomaremos por convenio y comodidad que  $\Gamma_j = \{0\}$  si  $j < 0$ .

El concepto de álgebra graduada es mucho más restrictivo que el de álgebra filtrada, por lo que tenemos la siguiente relación.

**Lema 3.3.5.** Toda álgebra graduada es un álgebra filtrada.

**Demostración.** Sea  $G = \bigoplus_{i \geq 0} G_i$  una álgebra graduada. Consideremos los espacios vectoriales definidos como  $\Gamma_k = \bigoplus_{i=0}^k G_i$ , es inmediato que  $\Gamma_k \subseteq \Gamma_{k+1}$ , y por cons-

trucción la suma total es  $G$ . Quedaría por ver la tercera propiedad de la definición, pero esta se deduce de que  $G_i G_j \subseteq G_{i+j}$  por ser  $G$  graduado, entonces

$$\Gamma_k \cdot \Gamma_m = \bigoplus_{i+j \leq k+m} G_i G_j \subseteq \bigoplus_{l \leq k+m} G_l = \Gamma_{k+m}.$$

Por tanto,  $\{\Gamma_k\}_{k \geq 0}$  es una filtración de  $G$ . |

Podemos usar una filtración para construir una álgebra graduada, ese será nuestro objetivo. Sea  $R$  una  $K$ -álgebra, y supongamos que  $\Gamma = \{\Gamma_i\}_{i \geq 0}$  es una filtración sobre  $R$ . Lo primero que vamos a necesitar definir para llegar a una álgebra graduada es la aplicación símbolo, que es

$$\sigma_k : \Gamma_k \rightarrow \Gamma_k / \Gamma_{k-1},$$

la proyección natural de  $\Gamma_k$  en  $\Gamma_k / \Gamma_{k-1}$ , por lo que  $\ker(\sigma_k) = \Gamma_{k-1}$ .

Consideremos ahora el  $K$ -espacio vectorial

$$\text{gr}^\Gamma R = \bigoplus_{i \geq 0} (\Gamma_i / \Gamma_{i-1}).$$

Este es nuestro candidato a anillo graduado. Por construcción tenemos que la suma directa de las componentes homogéneas es el total, por lo que solo tenemos que definir una multiplicación que verifique que  $\Gamma_i / \Gamma_{i-1} \cdot \Gamma_j / \Gamma_{j-1} \subseteq \Gamma_{i+j} / \Gamma_{i+j-1}$ . Los elementos homogéneos de grado  $k$  de  $\text{gr}^\Gamma R$  son los  $\sigma_k(a)$  para cierto  $a \in \Gamma_k$ . Sean entonces  $a \in \Gamma_k$ ,  $b \in \Gamma_m$ , entonces definimos

$$\sigma_k(a) \sigma_m(b) := \sigma_{k+m}(ab).$$

Con esta definición, está claro que se verifica la condición que necesitábamos para que  $\text{gr}^\Gamma R$  fuera una álgebra graduada, su componente homogénea de grado  $i$  será  $\Gamma_i / \Gamma_{i-1}$ . Esta álgebra graduada,  $\text{gr}^\Gamma R$ , será el *álgebra graduada sobre  $R$  asociada a  $\Gamma$* .

### 3.4 Filtraciones en D-módulos

Las dos filtraciones que nos van a resultar de mayor utilidad durante el trabajo son las asociadas al grado de un operador y al orden de un operador. La primera es la filtración  $F = \{F_k\}_{k \geq 0}$  donde  $F_k$  es el conjunto de operadores de  $D_n$  con orden menor o igual que  $k$ . A la segunda se la conoce como filtración de Bernstein, y es  $B = \{B_k\}_{k \geq 0}$  donde  $B_k$  es el conjunto de operadores de  $D_n$  con grado menor o igual que  $k$ . Es fácil

comprobar que ambas familias son, en efecto, filtraciones, las dos primeras propiedades de la definición de filtración son inmediatas de comprobar, mientras que la tercera es consecuencia del apartado (2) de los teoremas 2.2.4 y 2.2.1, respectivamente.

Veamos a continuación que las álgebras asociadas a ambas filtraciones son isomorfas al anillo de polinomios en  $2n$  variables sobre  $K$ . En particular tendremos que las álgebras asociadas a ambas filtraciones son isomorfas entre ellas.

**| Teorema 3.4.1.** *Sea  $S_n = \text{gr}^A D_n$  el álgebra graduada asociada a la filtración  $A$ ; es isomorfa al anillo de polinomios sobre  $K$  en  $2n$  variables. Donde  $A$  es la filtración  $F$  o  $B$ .*

**Demostración.** Haremos la prueba para el caso de la filtración  $F$ , para  $B$  la demostración es análoga usando grado en vez de orden cuando sea necesario.

Primero definamos  $y_i = \sigma_1(x_i)$  y  $\xi_i = \sigma_1(\partial_i)$ , para todo  $1 \leq i \leq n$ . Dividamos la demostración en tres pasos:

**Primer paso:** Veamos que  $S_n$  está generada por  $y_1, \dots, y_n, \xi_1, \dots, \xi_n$  como  $K$ -álgebra.

Un elemento de orden  $k$  en  $S_n$  será de la forma  $\sigma_k(P)$ , para cierto  $P \in D_n$ . Ahora bien,  $P$  será una combinación lineal de elementos  $x^\alpha \partial^\beta$  con  $\alpha, \beta \in \mathbb{N}^n$  y  $|\beta| \leq k$ . Si  $|\beta| < k$ , entonces  $\sigma_k(x^\alpha \partial^\beta) = 0$ , ya que pertenecería a  $F_{k-1}$ . Si por el contrario,  $|\beta| = k$ , entonces

$$\sigma_k(x^\alpha \partial^\beta) = \sigma_1^{\alpha_1}(x_1) \cdots \sigma_1^{\alpha_n}(x_n) \cdot \sigma_1^{\beta_1}(\partial_1) \cdots \sigma_1^{\beta_n}(\partial_n) = y_1^{\alpha_1} \cdots y_n^{\alpha_n} \cdot \xi_1^{\beta_1} \cdots \xi_n^{\beta_n},$$

por lo que  $\sigma(P)$  será una combinación lineal de monomios en las  $y_i$  y las  $\xi_i$  de orden  $k$  (redefiniendo el orden para  $\xi$ , en vez de para  $\partial$ ).

**Segundo paso:** Probemos que  $S_n$  es un anillo conmutativo.

Tenemos que comprobar que los generadores conmutan entre sí. Para  $i$  distinto de  $j$ , tenemos claro que  $\sigma_2(y_i \partial_j) = x_i \xi_j = \xi_j x_i = \sigma_2(\partial_j y_i)$ , ya que las  $x_i$  y  $\partial_j$  conmutan, lo mismo ocurriría para  $\sigma_2(x_i x_j)$  y  $\sigma_2(\partial_i \partial_j)$ . Solo nos queda ver que  $\sigma_2(x_i \partial_i) = \sigma_2(\partial_i x_i)$ , pero esto se tiene por la linealidad de cada  $\sigma_k$ :

$$\sigma_2(x_i \partial_i) = \sigma_2(\partial_i x_i + 1) = \sigma_2(\partial_i x_i) + \sigma_2(1) = \sigma_2(\partial_i x_i).$$

**Tercer paso:** Definir un homomorfismo  $\phi$  inyectivo.

Como ambos espacios están generados por la misma cantidad de generadores, y los generadores van en generadores, si encontramos un homomorfismo entre ellos que

sea inyectivo, inmediatamente tendremos la sobreyectividad y con ella la biyectividad, por lo que los anillos serán isomorfos.

Vamos a definir  $\phi : K[\gamma_1, \dots, \gamma_n][z_1, \dots, z_n] \rightarrow S_n$  como  $\phi(\gamma_i) = y_i$  y  $\phi(z_i) = \xi_i$ , para  $1 \leq i \leq n$ . Con el concepto de orden que tenemos definido, asignemos a las  $\gamma_i$  orden 0, al igual que las  $y_i$ , mientras que las  $z_i$  tendrán orden 1 como las  $\xi_i$ , de modo que  $\phi$  sea un homomorfismo graduado.

Sea  $G \in K[\gamma_1, \dots, \gamma_n][z_1, \dots, z_n]$  y supongamos que  $G \in \ker(\phi)$ . Dado que  $\phi$  es graduado, podemos asumir que  $G$  es homogéneo de orden  $k$ , dado por

$$G(\gamma_1, \dots, \gamma_n, z_1, \dots, z_n) = \sum_{\alpha, \beta \in \mathbb{N}^n, |\beta|=k} c_{\alpha\beta} \gamma_1^{\alpha_1} \cdots \gamma_n^{\alpha_n} \cdot z_1^{\beta_1} \cdots z_n^{\beta_n},$$

por lo que podemos definir un operador  $P \in D_n$  como

$$P = \sum_{\alpha, \beta \in \mathbb{N}^n, |\beta|=k} c_{\alpha\beta} x_1^{\alpha_1} \cdots x_n^{\alpha_n} \cdot \partial_1^{\beta_1} \cdots \partial_n^{\beta_n},$$

que verifica que  $\sigma_k(P) = \phi(G)$ . Si  $\sigma_k(P) = \phi(G) = 0$ , entonces  $P \in F_{k-1}$ . Pero como está expresado como combinación lineal de elementos de orden  $k$ , y tiene orden menor que  $k$ , eso implica que los  $c_{\alpha\beta}$  son todos nulos, por lo que  $G$  también sería nulo, con lo que la inyectividad de  $\phi$  quedaría probada, y con ello el teorema. |

Para definir una filtración sobre un módulo, necesitaremos definir el concepto de filtración inducida por otra filtración. Nos centraremos en definir las asociadas a las dos filtraciones anteriores.

**| Definición 3.4.2 (F-filtración).** Sea  $M$  un  $D_n$ -módulo a la izquierda. Una familia  $\Gamma = \{\Gamma_i\}_{i \geq 0}$  de  $K$ -espacios vectoriales es una filtración de  $M$  respecto de  $F$  si verifica que:

1.  $\Gamma_0 \subseteq \Gamma_1 \subseteq \cdots \subseteq M$ ,
2.  $\bigcup_{i \geq 0} \Gamma_i = M$ ,
3.  $F_i \Gamma_j \subseteq \Gamma_{i+j}$ ,
4.  $\Gamma_i$  es un  $K[x]$ -módulo finitamente generado para  $i \geq 0$ .

De igual forma, podemos dar esta definición para la filtración de Bernstein.

**| Definición 3.4.3 (B-filtración).** Sea  $M$  un  $D_n$ -módulo a la izquierda. Una familia  $\Gamma = \{\Gamma_i\}_{i \geq 0}$  de  $K$ -espacios vectoriales es una  $B$ -filtración de  $M$  si verifica que:

1.  $\Gamma_0 \subseteq \Gamma_1 \subseteq \cdots \subseteq M$ ,
2.  $\bigcup_{i \geq 0} \Gamma_i = M$ ,
3.  $B_i \Gamma_j \subseteq \Gamma_{i+j}$ ,
4.  $\Gamma_i$  es un  $K$ -espacio vectorial de dimensión finita, para  $i \geq 0$ .

En general, diremos que  $M$  es un  $D_n$ -módulo filtrado si admite alguna filtración, respecto de la del orden o la de Bernstein.

Sea  $A$  la filtración  $F$  o  $B$ , podemos definir un módulo graduado asociado a un módulo filtrado respecto de  $A$ . Se construye de la misma manera que una álgebra graduada, a excepción de la definición de producto, sea  $\mu$  el símbolo de  $\Gamma$ , entonces

$$\sigma_k(a) \cdot \mu_i(u) = \mu_{i+k}(au),$$

donde  $a \in A_k$ ,  $u \in \Gamma_i$  y  $\sigma$  es el símbolo de la filtración  $A$ .

Sea  $M$  un  $D_n$ -módulo a la izquierda con una  $A$ -filtración  $\Gamma$ . Supongamos que  $N$  es un submódulo de  $M$ . Vamos a construir filtraciones para  $N$  y  $M/N$  a partir de  $\Gamma$ , estas serán las filtraciones inducidas por  $\Gamma$ .

Primero tomemos  $\Gamma' = \{N \cap \Gamma_i\}_{i \geq 0}$ , que es una filtración para  $N$ , esto es cierto, porque hereda las condiciones de que  $\Gamma$  sea una filtración para  $M$ . Dado que  $N \subset M$ , podemos definir una aplicación inyectiva y  $K$ -lineal

$$\phi_k : \Gamma'_k / \Gamma'_{k-1} = (N \cap \Gamma_k) / (N \cap \Gamma_{k-1}) \rightarrow \Gamma_k / \Gamma_{k-1}.$$

Con estas aplicaciones podemos definir una aplicación lineal  $\phi : \text{gr}^{\Gamma'} N \rightarrow \text{gr}^{\Gamma} M$ . Además, esta aplicación es  $S_n$ -lineal: sean  $a \in A_i D_n$ ,  $b \in \Gamma'_j N$ , con  $i+j = k$ , entonces  $ab \in \Gamma'_k$ ,

$$\phi_k(ab + \Gamma'_{k-1}) = ab + \Gamma_{k-1} M = (a + F_{i-1} D_n)(b + \Gamma_{j-1} M) = (a + F_{i-1} D_n) \phi_j(b),$$

por lo que  $\phi$  es  $S_n$ -lineal, tomando  $a \in A_0 D_n$ , que será  $K[x]$  si  $A = F$  o  $K$  si  $A = B$ , tendríamos que las  $\phi_k$  son  $K[x]$ -lineales, en el caso de  $F$  y  $K$ -lineales para  $B$ .

Ahora tomemos  $\Gamma''_k = \Gamma_k / (\Gamma_k \cap N)$ , que por el segundo teorema de isomorfía podemos ver como  $\Gamma''_k = (\Gamma_k + N) / N$ , que es una filtración sobre  $M/N$ , ya que como  $\Gamma$  es una filtración, esta  $\Gamma''$  hereda algunas propiedades, obviamente  $\Gamma''_0 \subseteq \Gamma''_1 \subseteq \cdots \subseteq M/N$ , por lo que la propiedad 1 se cumple. La segunda es inmediata a partir de la segunda definición,

$$\bigcup_{i \geq 0} (\Gamma_i + N) / N = \left( \bigcup_{i \geq 0} (\Gamma_i + N) \right) / N = M / N.$$

La tercera propiedad es cierta porque  $F_i \cdot (\Gamma_k + N) = F_i \Gamma_k + F_i N \subseteq \Gamma_{k+i} + N$ , de forma que usando el tercer teorema de isomorfía  $F_i \cdot \Gamma_k'' = F_i \cdot (\Gamma_k + N)/N \subseteq (\Gamma_{k+i} + N)/N = \Gamma_{k+i}''$ . La cuarta se cumple ya que como  $\Gamma_k$  es  $K[x]$ -finitamente generado o de dimensión finita sobre  $K$ , entonces  $\Gamma_k/J$  es finitamente generado para todo sub- $K[x]$ -módulo,  $J$ .

Notemos ahora que

$$\Gamma_k''/\Gamma_{k-1}'' \simeq \Gamma_k/(\Gamma_{k-1} + (\Gamma_k \cap N)),$$

ya que por el tercer teorema de isomorfía,

$$\Gamma_k/(\Gamma_{k-1} + (\Gamma_k \cap N)) \stackrel{3^\circ}{\simeq} (\Gamma_k/(\Gamma_k \cap N))/((\Gamma_{k-1} + (\Gamma_k \cap N))/\Gamma_k \cap N)$$

y por el segundo,

$$(\Gamma_{k-1} + (\Gamma_k \cap N))/(\Gamma_k \cap N) \stackrel{2^\circ}{\simeq} \Gamma_{k-1}/(\Gamma_{k-1} \cap \Gamma_k \cap N) = \Gamma_{k-1}/(\Gamma_{k-1} \cap N).$$

Sea  $\pi_k : \Gamma_k/\Gamma_{k-1} \rightarrow \Gamma_k''/\Gamma_{k-1}''$  la proyección canónica. Al igual que con  $\phi$ , podemos definir una aplicación a partir de las  $\pi_k$  y obtenemos la aplicación  $K$ -lineal

$$\pi : \text{gr}^\Gamma M \rightarrow \text{gr}^{\Gamma''} M/N,$$

que no es más que la proyección inducida por la inclusión de  $S_n$ -módulos  $\text{gr}^{\Gamma'} N \xrightarrow{\phi} \text{gr}^\Gamma M$ . La prueba de que  $\text{gr}^{\Gamma''} M/N \simeq \text{gr}^\Gamma M / \text{gr}^{\Gamma'} N$  se puede ver componente a componente del mismo modo que la discusión después de la definición 3.3.3.





## 4 | Funciones holónomas

En este capítulo introduciremos a las protagonistas principales del trabajo, las funciones holónomas, para ello deberemos ver primero la definición de variedad característica y el teorema fundamental del análisis algebraico, que nos permitirán definir el concepto de ideal holónomo. Después de esto introduciremos brevemente algunos conceptos sobre polinomios de Hilbert que nos permitirán dar una caracterización sobre funciones holónomas muy útil. Terminaremos el capítulo comprobando que la holonomía se mantiene bajo operaciones como la suma, el producto, la restricción, derivación e integración, para ello necesitaremos unas pinceladas de teoría de Fourier.

### 4.1 Ideales y módulos holónomos

Como hemos dicho, nuestro objetivo en esta sección es definir lo que es un ideal holónomo, y usar el polinomio de Hilbert para dar una caracterización sobre ser ideal holónomo.

Usaremos  $\sigma(P)$  para referirnos al símbolo de un operador,  $P \in D_n$ , respecto de la filtración por el orden,  $F$ .

**| Definición 4.1.1 (Variedad característica).** Sea  $I$  un ideal de  $D_n$ . La variedad característica de  $I$ , denotada por  $\text{Char}(I)$ , es el conjunto de los ceros del ideal característico,  $\text{ch}(I) := \sigma(I) := \{\sigma(P) | P \in I\}$  en  $\mathbb{C}^{2n}$ . Es decir,  $\text{Char}(I) = \mathcal{V}(\text{ch}(I)) = \mathcal{V}(\sigma(I))$ .

**| Teorema 4.1.2 (Teorema fundamental del análisis algebraico).** Sea  $I$  un ideal propio de  $D_n$ . Toda componente irreducible de su variedad característica tiene dimensión al menos  $n$ .

La demostración de este teorema no la realizaremos en el trabajo por su complejidad,

pero una prueba puede encontrarse en [SKK73].

Para la siguiente definición necesitaremos primero dejar clara un poco de notación, denotaremos por  $R$  al álgebra de Weyl racional sobre los complejos, es decir

$$R = \mathbb{C}(x_1, \dots, x_n) \langle \partial_1, \dots, \partial_n \rangle.$$

**| Definición 4.1.3 (Ideal holónimo).** *Un ideal  $I$  de  $D_n$  se dirá holónimo si  $\dim(\text{ch}(I)) = n$ , es decir, si la dimensión de su variedad característica en  $\mathbb{C}^{2n}$  es minimal. Consideremos ahora  $\mathbb{C}(\underline{x})$  y  $\mathbb{C}(\underline{x})[\underline{\xi}]$ , el rango holónimo de  $I$  se define como  $\text{rang}(I) = \dim_{\mathbb{C}(\underline{x})}(R/RI)$ .*

La definición de módulo holónimo se puede ver mediante una  $B$ -filtración  $\Gamma$  como aquel que verifica que el grado de su polinomio de Hilbert respecto de  $\Gamma$  es  $n$ . Para poder afirmar esto formalmente deberemos introducir algunos resultados sobre polinomios de Hilbert. Seguiremos el esquema presentado en [CJ10, Subsection 2.4].

**| Definición 4.1.4 (Polinomio de Hilbert).** *Sea  $M$  un módulo sobre  $D_n$  finitamente generado y dotado de una  $B$ -filtración  $\Gamma = \{M_k\}_{k \geq 0}$ , denotaremos por*

$$\text{gHF}_{\text{gr}^\Gamma(M)} : \mathbb{N} \rightarrow \mathbb{N}$$

*al polinomio de Hilbert del  $\text{gr}^B(D_n)$ -módulo  $\text{gr}^\Gamma(M)$ , y lo definimos como*

$$\text{gHF}_{\text{gr}^\Gamma(M)}(v) = \dim_{\mathbb{C}} \left( \frac{M_v}{M_{v-1}} \right),$$

*para todo  $v \in \mathbb{N}$ .*

**| Teorema 4.1.5 (Hilbert, Serre).** *Existe un único polinomio  $\text{gHP}_{\text{gr}^\Gamma(M)}(t) \in \mathbb{Q}[t]$  tal que  $\text{gHF}_{\text{gr}^\Gamma(M)}(v) = \text{gHP}_{\text{gr}^\Gamma(M)}(v)$  para  $v \in \mathbb{N}$  lo suficientemente grande. El grado de  $\text{gHP}$  es  $d - 1$ , donde  $d = \dim(\mathcal{V}(\text{ann}_{\text{gr}^B(D_n)}(\text{gr}^\Gamma(M))))$ , es más, es menor o igual que  $2n - 1$ .*

Una prueba de este teorema se puede encontrar en [Har77, Theorem 7.5]. Al polinomio  $\text{gHP}_{\text{gr}^\Gamma(M)}(t)$  se le conoce como polinomio de Hilbert del módulo graduado  $\text{gr}^\Gamma(M)$ .

Denotaremos por  $\text{HF}_{M,\Gamma} : \mathbb{N} \rightarrow \mathbb{N}$  a la aplicación definida por

$$\text{HF}_{M,\Gamma}(v) = \sum_{k=0}^v \text{gHF}_{\text{gr}^\Gamma(M)}(k) = \dim_{\mathbb{C}} M_v,$$

para  $v \in \mathbb{N}$ .

Ahora enunciaremos un corolario de [Har77, Proposition 7.3].

**Corolario 4.1.6.** Sea  $M$  un módulo finitamente generado sobre  $D_n$  dotado de una  $B$ -filtración  $\Gamma = \{M_k\}_{k \geq 0}$ . Existe un único polinomio  $\text{HP}_{M,\Gamma}(t) \in \mathbb{Q}[t]$  tal que  $\text{HF}_{M,\Gamma}(v) = \text{HP}_{M,\Gamma}(v)$  para  $v \in \mathbb{N}$  lo suficientemente grande. El grado de dicho polinomio es igual a la dimensión de Krull de  $\mathcal{V}(\text{ann}_{\text{gr}^B(D_n)}(\text{gr}^\Gamma(M)))$  y es menor o igual que  $2n$ .

El polinomio  $\text{HP}_{M,\Gamma}$  es el polinomio de Hilbert de  $M$  respecto de la  $B$ -filtración  $\Gamma$ . Se puede probar que el grado de dicho polinomio no depende de la  $B$ -filtración respecto de la cual lo tomemos.

**| Definición 4.1.7 (Módulo holónimo).** Sea  $M$  un módulo sobre  $D_n$  finitamente generado y  $\Gamma$  una  $B$ -filtración de  $M$ , definimos la dimensión de  $M$  como  $\dim(M) = \deg(\text{HP}_{M,\Gamma})$ . Diremos que un módulo es holónimo si  $\dim(M) = n$ .

El siguiente teorema nos resultará muy útil, dicho teorema afirma que  $I$  será un ideal holónimo si y solo si  $D_n/I$  es un módulo holónimo.

**| Teorema 4.1.8 ([Ber71, Theorem 3.1]).** Sea  $I \subseteq D_n$ ,  $M = D_n/I$  y  $\Gamma$  una  $B$ -filtración de  $M$ , entonces  $\dim(M) = \dim(I)$ , por lo que  $I$  será holónimo si y solo si  $\deg(\text{HP}_{M,\Gamma}) = n$ .

Veamos algunos ejemplos de por qué esto es útil. Sea  $k \geq 0$ , recordemos que  $B_k$  es el espacio de operadores con grado menor o igual que  $k$ . Como veremos en la demostración de la proposición 4.2.1,  $\dim(B_k \cap \mathbb{C}[\underline{x}]) = \binom{n+k}{k}$ , es decir, es un polinomio de grado  $n$  en  $k$ , por lo que,  $\mathbb{C}[\underline{x}]$  es un ideal holónimo. Como vimos en la proposición 3.1.4,  $\mathbb{C}[\underline{x}] \simeq D_n/\langle \partial_1, \dots, \partial_n \rangle$ , de forma que  $D_n/\langle \partial_1, \dots, \partial_n \rangle$  será un módulo holónimo, pero por el teorema 4.1.8, tendremos que  $\langle \partial_1, \dots, \partial_n \rangle$  es un ideal holónimo. De igual forma, la  $\dim(B_k \cap D_n) = \dim(B_k) = \binom{2n+k}{k}$ , es un polinomio de grado  $2n$  en  $k$ , de donde concluimos que  $D_n$  no es un ideal holónimo, aplicando el mismo razonamiento que antes llegamos a que  $\{0\}$  tampoco lo es.

## 4.2 Definición y caracterización

En esta sección veremos la caracterización más importante para que una función sea holónoma, ya que será el método que más usaremos, por ser el más simple, para

afirmar que una función es holónoma.

Sea  $M$  un  $D_n$ -módulo y sea  $f \in M$ , diremos que  $f$  es holónoma si  $\text{ann}_{D_n}(f)$  es un ideal holónomo. Si podemos ver  $f$  como una función infinitamente diferenciable en un abierto de  $\mathbb{R}^n$  o  $\mathbb{C}^n$ , entonces la llamaremos función holónoma. A partir de ahora, cuando digamos que un módulo es libre de torsión, querremos decir libre de torsión sobre  $K[\underline{x}]$ .

**Proposición 4.2.1.** Sea  $f$  un elemento de un  $D_n$ -módulo  $M$  libre de torsión. Entonces, las siguientes condiciones son equivalentes:

1.  $f$  es holónoma,
2.  $\text{rang}(\text{ann}_{D_n}(f)) < \infty$ ,
3. para cada  $i \leq n$  existe un operador  $P_i \in \mathbb{C}[\underline{x}]\langle \partial_i \rangle \setminus \{0\}$  que anula a  $f$ .

*Demostración.*

$1 \Rightarrow 3$  Es consecuencia directa de [Zei90, Lemma 4.1], que dice que si  $L$  es un ideal a la izquierda de  $D_n(\mathbb{C})$  tal que  $D_n(\mathbb{C})/L$  es un módulo holónomo, entonces para cada  $n+1$  generadores entre  $\langle x_1, \dots, x_n, \partial_1, \dots, \partial_n \rangle$ , hay un elemento no nulo de  $L$  que depende de estos generadores.

Probemos este resultado para  $x_1, \dots, x_n, \partial_1$ , como veremos, la prueba no depende de los generadores elegidos, solo de la cantidad de ellos, que siempre será  $n+1$ . Consideremos la aplicación

$$\phi : \mathbb{C}[\underline{x}]\langle \partial_1 \rangle \rightarrow D_n/L,$$

dada por  $\phi(P) = P + L$ . Sean ahora  $B_m^1$  y  $B_m^L$  las filtraciones de Bernstein inducidas en  $\mathbb{C}[\underline{x}]\langle \partial_1 \rangle$  y  $D_n/L$ , respectivamente. Por la definición de módulo holónomo usando la  $B$ -filtración, como  $D_n/L$  es holónomo,  $\dim(B_m^L)$  viene dada por un polinomio de grado  $n$ , para  $m$  lo suficientemente grande. Veamos qué es  $\dim(B_m^1)$ .

Tenemos que  $B_m^1 = \{\lambda x^\alpha \partial_1^\beta \mid \lambda \in K, \alpha \in \mathbb{N}^n, \beta \in \mathbb{N}, |\alpha| + \beta \leq m\}$ , por lo que la dimensión será el cardinal del conjunto de generadores, que es  $\dim(B_m^1) = \sum_{k=0}^m \binom{n+k}{n} \stackrel{(*)}{=} \binom{n+m+1}{m}$ , por lo que es un polinomio de grado  $n+1$  en  $m$ .

Veamos antes de continuar que la identidad  $(*)$  es cierta. Por inducción sobre  $m$ , si  $m = 0$  es inmediato. Veamos el caso  $m = 1$ ,  $\dim(B_1^1) = \binom{n}{n} + \binom{n+1}{n} = 1 + n + 1 = n + 2 = \binom{n+2}{1}$ . Supongamos que es cierto para todo  $m$  menor que

cierto  $k$ , veamos el caso  $m = k$ :

$$\begin{aligned} \dim(B_k^1) &= \sum_{j=0}^k \binom{n+j}{n} = \sum_{j=0}^{k-1} \binom{n+j}{n} + \binom{n+k}{n} = \binom{n+k}{k-1} + \binom{n+k}{n} = \\ &= \binom{n+k}{n+1} + \binom{n+k}{n} = \binom{n+k+1}{n+1}. \end{aligned}$$

Entonces tenemos que  $\phi$ , restringida a  $B_m^1$ , es un homomorfismo filtrado que va de un espacio de dimensión mayor a uno de dimensión menor, por lo que el núcleo de dicha aplicación no puede ser cero, así que existirá un operador  $P \in B_m^1$  no nulo que va al cero, es decir, pertenece al ideal  $L$ , porque el núcleo es  $L \cap \mathbb{C}[\underline{x}]\langle \partial_1 \rangle$ .

Si ahora aplicamos el resultado recién probado, tomando  $L = \text{ann}_{D_n}(f)$  y los mismos generadores, concluye la prueba.

3  $\Rightarrow$  2 Sabemos que  $\text{rang}(\text{ann}_{D_n}(f)) = \dim_{\mathbb{C}(\underline{x})}(R/R \text{ann}_{D_n}(f)) < \infty$ . Sean, para todo  $i = 1, \dots, n$ ,  $P_i \in \mathbb{C}[\underline{x}]\langle \partial_i \rangle$ , que anulen a  $f$ , con  $N_i = \text{ord}(P_i)$ , entonces  $P_i = \sum_{k=0}^{N_i} p_{ik} \partial_i^k$ , con  $p_{ik} \in \mathbb{C}[\underline{x}]$ . Sean ahora

$$Q_i := \sum_{k=0}^{N_i} \frac{p_{ik}}{p_{iN_i}} \partial_i^k = \partial_i^{N_i} + \dots \in R \text{ann}_{D_n}(f).$$

Tomando los distintos  $Q_i$  podemos definir el ideal  $J = \langle Q_1, \dots, Q_n \rangle \subseteq R \text{ann}_{D_n}(f) \subseteq R$ .

Teniendo esto en cuenta, podemos definir el homomorfismo sobreyectivo de  $R$ -módulos

$$R/J \rightarrow R/R \text{ann}_{D_n}(f),$$

por lo que  $\dim_{\mathbb{C}(\underline{x})} R/R \text{ann}_{D_n}(f) \leq \dim_{\mathbb{C}(\underline{x})} R/J$ , veamos que esto es finito.

$$\begin{aligned} \dim_{\mathbb{C}(\underline{x})}(R/J) &= \dim_{\mathbb{C}(\underline{x})}(R/\langle Q_1, \dots, Q_n \rangle) = \\ &= \dim_{\mathbb{C}(\underline{x})}(R/\langle \partial_1^{N_1} + \dots, \dots, \partial_n^{N_n} + \dots \rangle) \leq N_1 \dots N_n < \infty \end{aligned}$$

2  $\Rightarrow$  1 No daremos la prueba de este apartado por la dificultad del mismo, pero se puede consultar una indicación de la prueba en [SST00, Theorem 1.4.15].

**Corolario 4.2.2.** Los polinomios en  $n$  variables son funciones holónomas.

**Demostración.** Sea  $f$  un polinomio en  $n$  variables, si definimos  $\alpha_i$  como el grado en la variable  $x_i$  de  $f$ , tenemos que  $\partial_i^{\alpha_i+1}(f) = 0$ . Por lo que aplicando el apartado (3) de la proposición anterior, tendremos que  $f$  es una función holónoma.

### 4.3 Conservación de la holonomía

En esta sección veremos como se comporta la holonomía mediante las operaciones usuales, como composición, suma y producto, restricción, derivación e integración. Trabajaremos con funciones  $C^\infty(U)$  o analíticas en un abierto  $U \subseteq \mathbb{C}^n$ . Los cálculos que llevaremos a cabo en esta sección son formales, es decir, supondremos que las operaciones que vamos a realizar tienen sentido.

Lo primero que veremos es que la holonomía se mantiene por transformaciones lineales, esto será de vital importancia para probar el resto de propiedades.

**Proposición 4.3.1.** Sea  $\varphi : D_n \rightarrow D_n$  el homomorfismo de  $\mathbb{C}$ -álgebras dado por  $x_i \mapsto a_i x_i + b_i$  y  $\partial_i \mapsto a_i^{-1} \partial_i$ , con  $a_i, b_i \in \mathbb{C}$  y  $a_i \neq 0$  para todo  $i$ . Entonces:

- a)  $\varphi$  es un automorfismo filtrado de  $D_n$  con respecto a la filtración de Bernstein.
- b)  $f(x)$  es una función holónoma si y solo si  $(f \circ \varphi)(x)$  lo es.
- c)  $D_n/I$  es un módulo holónomo si y solo si  $\varphi(D_n/I) \simeq D_n/\varphi(I)$  es holónomo.

**Demostración.** El apartado a) de la proposición es directo ya que el automorfismo  $\varphi$  no varía el grado de un operador, por lo que respeta grados, que es la definición de aplicación filtrada. Que es una biyección también está bastante claro, si tomamos  $\varphi'$  verificando que  $\varphi'(x_i) = a_i^{-1} x_i - a_i^{-1} b_i$  y  $\varphi'(\partial_i) = a_i$ , de esta forma  $(\varphi \circ \varphi')(x_i) = \varphi(\varphi'(x_i)) = \varphi(a_i^{-1} x_i - a_i^{-1} b_i) = a_i^{-1}(a_i x_i + b_i) - a_i^{-1} b_i = x_i$  y  $\varphi'(\varphi(x_i)) = \varphi'(a_i x_i + b_i) = a_i(a_i^{-1} x_i - a_i^{-1} b_i) + b_i = x_i$ , con las  $\partial_i$  ocurre igual y al ser  $\varphi$  lineal esto es suficiente. Ya solo nos queda ver que verifica las relaciones que caracterizan a  $D_n$ , veamos que se verifica que para  $i \neq j$ ,  $1 = [\partial_i, x_i] = \varphi([\partial_i, x_i])$ ,

$$\begin{aligned} \varphi([\partial_i, x_i]) &= \varphi(\partial_i)\varphi(x_i) - \varphi(x_i)\varphi(\partial_i) = a_i^{-1}\partial_i(a_i x_i + b_i) - (a_i x_i + b_i)a_i^{-1}\partial_i = \\ &= \partial_i x_i - x_i \partial_i = [\partial_i, x_i] = 1, \end{aligned}$$

por lo que ya podemos afirmar que  $\varphi$  es un automorfismo de  $D_n$  filtrado.

Para el b), solo tenemos que probar la implicación de izquierda a derecha, ya que de derecha a izquierda pasamos tomando otro automorfismo  $\varphi'$  que lleve  $x_i \mapsto a_i^{-1} x_i - a_i^{-1} b_i$ , de forma que  $(f \circ \varphi \circ \varphi')(\underline{x}) = f(\underline{x})$ . Supongamos ahora que  $f$  es holónoma, entonces por la proposición 4.2.1 para todo  $i \leq n$  existe  $P_i \in \mathbb{C}[\underline{x}]\langle \partial_i \rangle$  tal que  $P_i \bullet f = 0$ . Veamos que  $\varphi(P_i \bullet f) = \varphi(P_i) \bullet \varphi(f) = \varphi(P_i) \bullet (f \circ \varphi)$ , y como esto es nulo y  $\varphi(P_i) \in \mathbb{C}[\underline{x}]\langle \partial_i \rangle$ , tendremos que  $f \circ \varphi$  es holónoma. Por comodidad, definamos el producto  $*_{\mathbb{C}}$ , dado por  $\underline{a} *_{\mathbb{C}} \underline{b} = \underline{a} \cdot \text{Id}_n \cdot \underline{b}^t \in \mathbb{C}^n$  para cualesquiera  $\underline{a}, \underline{b} \in \mathbb{C}^n$ . Apliquemos ahora

$\varphi$  a  $f$  y  $P_i$ :

$$f = \sum_{\alpha} f_{\alpha} \underline{x}^{\alpha}, \text{ entonces } \varphi(f) = \sum_{\alpha} f_{\alpha} (\underline{a} *_{\underline{c}} \underline{x} + \underline{b})^{\alpha},$$

$$P_i = \sum_{\beta\gamma} P_{\beta\gamma}^i \underline{x}^{\beta} \partial_i^{\gamma}, \text{ entonces } \varphi(P_i) = \sum_{\beta\gamma} P_{\beta\gamma}^i (\underline{a} *_{\underline{c}} \underline{x} + \underline{b})^{\beta} (a_i^{-1} \partial_i)^{\gamma},$$

por otro lado,

$$P_i \bullet f = \sum_{\beta\gamma} P_{\beta\gamma}^i \underline{x}^{\beta} \partial_i^{\gamma} (f) = \sum_{\alpha\beta} f_{\alpha} \underline{x}^{\beta} \sum_{\gamma \leq \alpha_i} P_{\beta\gamma}^i \frac{\alpha_i!}{(\alpha_i - \gamma)!} \underline{x}^{\alpha - \gamma e_i},$$

$$\varphi(P_i \bullet f) = \sum_{\alpha\beta} f_{\alpha} (\underline{a} *_{\underline{c}} \underline{x} + \underline{b})^{\beta} \sum_{\gamma \leq \alpha_i} P_{\beta\gamma}^i \frac{\alpha_i!}{(\alpha_i - \gamma)!} (\underline{a} *_{\underline{c}} \underline{x} + \underline{b})^{\alpha - \gamma e_i}.$$

Por último,

$$\begin{aligned} \varphi(P_i) \bullet \varphi(f) &= \sum_{\beta\gamma} P_{\beta\gamma}^i (\underline{a} *_{\underline{c}} \underline{x} + \underline{b})^{\beta} (a_i^{-1} \partial_i)^{\gamma} \left( \sum_{\alpha} f_{\alpha} (\underline{a} *_{\underline{c}} \underline{x} + \underline{b})^{\alpha} \right) = \\ &= \sum_{\alpha\beta} f_{\alpha} (\underline{a} *_{\underline{c}} \underline{x} + \underline{b})^{\beta} \sum_{\gamma \leq \alpha_i} P_{\beta\gamma}^i a_i^{-\gamma} \partial_i^{\gamma} ((\underline{a} *_{\underline{c}} \underline{x} + \underline{b})^{\alpha}) = \\ &= \sum_{\alpha\beta} f_{\alpha} (\underline{a} *_{\underline{c}} \underline{x} + \underline{b})^{\beta} \sum_{\gamma \leq \alpha_i} P_{\beta\gamma}^i a_i^{-\gamma} a_i^{\gamma} \frac{\alpha_i!}{(\alpha_i - \gamma)!} (\underline{a} *_{\underline{c}} \underline{x} + \underline{b})^{\alpha - \gamma e_i} = \varphi(P_i \bullet f) = 0. \end{aligned}$$

Por lo que podemos concluir que  $\varphi(f) = f \circ \varphi$  es holónoma.

El último apartado es consecuencia directa de que  $\varphi$  sea un automorfismo, luego  $D_n/I \simeq \varphi(D_n/I) \simeq D_n/\varphi(I)$ . Y además graduado, por lo que si  $\Gamma = \{\Gamma_k\}_{k \geq 0}$  es una  $B$ -filtración de  $D_n/I$ , entonces para cada  $k$ ,  $\Gamma_k \simeq \varphi(\Gamma_k)$  como  $K$ -espacio vectorial, y será una  $B$ -filtración para  $D_n/\varphi(I)$ . Como el grado del polinomio de Hilbert no depende de la  $B$ -filtración elegida, el polinomio de Hilbert de  $D_n/I$  respecto de  $\Gamma$  tendrá grado  $n$  si y solo si el polinomio de Hilbert de  $D_n/\varphi(I)$  respecto de  $\varphi(\Gamma)$  tiene grado  $n$ , es decir,  $D_n/I$  será un módulo holónomo si y solo si  $\varphi(D_n/I)$  lo es. |

**Proposición 4.3.2.** Sean  $n = 1$ ,  $f(x)$  holónoma y  $g(x)$  algebraica, entonces, su composición  $(f \circ g)(x) = f(g(x))$  es holónoma.

**Demostración.** Definamos  $h(x) := f(g(x))$ . Si derivamos  $h$  sucesivamente, por la regla de la cadena, todas sus derivadas van a ser una combinación lineal de las  $f^{(i)}(g)$  para  $i \geq 0$ , con coeficientes en  $\mathbb{C}[g, g', \dots]$ . Por hipótesis,  $g$  es algebraica, por lo que existirá un polinomio  $G(s, t) \in \mathbb{C}[s, t]$  tal que  $G(x, g) = 0$ . Derivando esta ecuación

respecto de  $x$ , podríamos expresar los  $g^{(i)}$  en función de  $x$  y  $g$ . De esta forma tenemos que  $\mathbb{C}[g, g', \dots]$  está contenido en  $\mathbb{C}(x, g)$ .

Llamemos  $W$  al espacio vectorial generado sobre  $\mathbb{C}(x, g)$  por todas las composiciones  $f^{(i)}(g)$  para  $i \geq 0$  y  $V$  al espacio vectorial generado sobre  $\mathbb{C}(x)$  por las derivadas  $f^{(i)}$ . Dado que  $f$  es holónoma, existe un  $P \in \mathbb{C}(x)\langle\partial\rangle$  tal que  $P \bullet f = 0$ , por lo que si  $P = \sum_{i=0}^N P_i \partial^i$ , con  $P_i \in \mathbb{C}(x)$  y  $P_N = 1$ , tenemos que  $P \bullet f = \sum_{i=0}^N P_i \partial^i f = \sum_{i=0}^N P_i f^{(i)} = 0$ , es decir, tenemos una relación de dependencia lineal entre las derivadas sucesivas de  $f$ , por lo que la dimensión de  $V$  sobre  $\mathbb{C}(x)$  será menor que  $N < \infty$ . De igual forma,  $f^N(g) = \sum_{i=0}^{N-1} P_i f^{(i)}(g)$ , por lo que  $\{f^{(i)}(g)\}_{0 \leq i \leq N}$  no es linealmente independiente, entonces la dimensión de  $W$  sobre  $\mathbb{C}(x, g)$  es finita. Por último, como  $g$  es algebraico,  $\mathbb{C}(x)[g] = \mathbb{C}(x, g)$  es un espacio vectorial de dimensión finita sobre  $\mathbb{C}(x)$ . Dado que  $W$  es un  $\mathbb{C}(x, g)$ -espacio vectorial de dimensión finita y  $\mathbb{C}(x, g)$  también lo es, respecto de  $\mathbb{C}(x)$ , entonces  $W$  será un  $\mathbb{C}(x)$ -espacio vectorial de dimensión finita. Por último, como  $Z = \{(f \circ g)^{(i)}\} \subseteq W$ ,  $\dim(Z) < \infty$ , de forma que existen  $h_{k_1}, \dots, h_{k_r} \in \mathbb{C}(x)$  no nulos, tales que  $\sum_{i=1}^r h_{k_i} (f \circ g)^{(k_i)} = 0$ . La proposición 4.2.1 nos asegura que  $f \circ g$  es holónoma. |

**Proposición 4.3.3.** Sean  $f, g$  dos funciones holónomas, entonces tanto  $f + g$  como  $fg$  son funciones holónomas.

**Demostración.** Como  $f$  y  $g$  son holónomas, para cada  $i \leq n$  hay unos operadores  $P_i, Q_i \in \mathbb{C}[x_1, \dots, x_n]\langle\partial_i\rangle$  tales que  $P_i \bullet f = Q_i \bullet g = 0$ . Sean  $n_i = \text{ord}(P_i)$  y  $m_i = \text{ord}(Q_i)$ , entonces el espacio vectorial generado por  $\{\partial_i^k(f)\}_{k=0, \dots, n_i}$  sobre  $\mathbb{C}(x)$ , que llamaremos  $V$ , tiene dimensión menor o igual que  $n_i$ . De igual forma el espacio vectorial generado por los  $\{\partial_i^k g\}_{k=0, \dots, m_i}$  sobre  $\mathbb{C}(x)$ , que llamaremos  $W$ , tiene dimensión menor o igual que  $m_i$ .

Consideremos ahora el espacio generado por  $\{\partial_i^k(f + g)\}_{k=0, \dots, n_i+m_i} = \{\partial_i^k(f) + \partial_i^k(g)\}_{k=0, \dots, n_i+m_i}$  sobre  $\mathbb{C}(x)$ , está contenido en  $V + W$ , que tiene dimensión menor o igual que  $n_i + m_i$ , por lo que tendrá dimensión menor o igual que  $n_i + m_i$  también. Esto nos asegura que existirá un operador no nulo,  $S_i \in \mathbb{C}[\underline{x}]\langle\partial_i\rangle$  tal que  $S_i \bullet (f + g) = 0$ . Dado que esto es cierto para todo  $1 \leq i \leq n$ , entonces  $f + g$  es holónoma.

Sea  $i \in \{1, \dots, n\}$ . Si consideramos ahora el espacio vectorial generado por  $\{\partial_i^k(fg)\}_{k=0, \dots, n_i m_i}$ , al igual que antes la dimensión de nuestro espacio vectorial es menor o igual que  $m_i n_i$ , porque el espacio generado por  $\{\partial_i^k(fg)\}_{k=0, \dots, n_i m_i} = \{g \partial_i^k(f) + f \partial_i^k(g)\}_{k=0, \dots, n_i m_i}$  está contenido en el generado por  $\{h \cdot v \mid h \in W, v \in V\}$ , que como  $\dim(V) \leq n_i$  y  $\dim(W) \leq m_i$ , tendrá dimensión menor o igual que el producto  $n_i m_i$ ,



por lo que habrá un operador no nulo de  $T_i \in \mathbb{C}[\underline{x}]\langle\partial_i\rangle$  tal que  $T_i \bullet (fg) = 0$ . Como esto se verifica para todo  $i$  podemos concluir que  $fg$  es holónoma. |

**Proposición 4.3.4.** Sea  $f$  una función holónoma en  $n$  variables y sea  $m < n$ . Entonces, la restricción de  $f$  al subespacio  $\{x_{m+1} = x_{m+2} = \cdots = x_n = 0\}$  es una función holónoma en  $x_1, \dots, x_m$ .

**Demostración.** Como  $f$  es holónoma, por la proposición 4.2.1 para todo  $i \leq n$ , existe un  $P_i \in \mathbb{C}[\underline{x}]\langle\partial_i\rangle$  tal que  $P_i \bullet f = 0$ . Este  $P_i$  lo podemos expresar como

$$P_i = Q_i - x_{m+1}R_{m+1}^i - \cdots - x_n R_n^i,$$

donde  $Q_i \in \mathbb{C}[x_1, \dots, x_m]\langle\partial_i\rangle$  y  $R_j^i \in \mathbb{C}[\underline{x}]\langle\partial_i\rangle$  para todo  $j$ . Teniendo esto en cuenta,  $Q_i = P_i + \sum_{j=m+1}^n x_j R_j^i$ , por lo que

$$Q_i \bullet f = P_i \bullet f + \sum_{j=m+1}^n x_j R_j^i \bullet f.$$

Si tomamos la restricción al plano  $H = \{x_{m+1} = x_{m+2} = \cdots = x_n = 0\}$ , tenemos que  $Q_i \bullet f|_H = P_i \bullet f = 0$ .

Hemos encontrado un operador en  $\mathbb{C}[x_1, \dots, x_m]\langle\partial_i\rangle \subset \mathbb{C}[\underline{x}]\langle\partial_i\rangle$  que anula a  $f$  para todo  $i \leq m$ , usando otra vez la proposición 4.2.1 concluimos que  $f|_H$  es holónoma. |

**Corolario 4.3.5.** Sea  $f$  una función holónoma, su restricción a cualquier variedad afín  $\{x_i = a_i \mid i > m\}$  es una función holónoma.

**Demostración.** Ya hemos visto en la proposición anterior que la restricción a la variedad afín  $\{x_{m+1} = x_{m+2} = \cdots = x_n = 0\}$ , con  $m < n$  es holónoma, pero para pasar de esta variedad a la variedad  $\{x_{m+1} = a_{m+1}, x_{m+2} = a_{m+2}, \dots, x_n = a_n\}$ , solo tenemos hacer un cambio de variables, que no es más que aplicar una transformación lineal  $\varphi$  que lleve  $x_i \mapsto a_i$  para  $m+1 \leq i \leq n$ . Y por la proposición 4.3.1 sabemos que esto no afecta a la holonomía. |

**Proposición 4.3.6.** Una función es holónoma si y solo si sus derivadas parciales lo son.

**Demostración.** Sea  $f$  holónoma, por la proposición 4.2.1 sabemos que para todo  $i \leq n$  existe un  $P_i \in \mathbb{C}[\underline{x}]\langle\partial_i\rangle \setminus \{0\}$  que verifica que  $P_i \bullet f = 0$ . Estos  $P_i$  podemos escribirlos

como  $P_i = \hat{P}_i \partial_i + a_i(x)$  con  $a_i \in \mathbb{C}[\underline{x}]$ . Si  $a_i$  es nulo, entonces  $\hat{P}_i$  anula a la derivada parcial respecto de  $x_i$  de  $f$ , con lo que quedaría probado el resultado. Si  $a_i$  no fuera nulo, entonces como  $f$  y  $a_i$  son holónomas, el segundo por ser un polinomio, entonces por la proposición 4.3.3  $a_i f$  es holónoma, y existirá un  $Q_i \in \mathbb{C}[\underline{x}] \langle \partial_i \rangle$  que anule a  $a_i f$ , por lo que si tomamos  $Q_i \hat{P}_i$ , tenemos que

$$(Q_i \hat{P}_i) \cdot (\partial f / \partial x_i) = (Q_i (\hat{P}_i \partial_i)) \cdot f = (Q_i (P_i - a_i)) \cdot f = Q_i \cdot (P_i \cdot f) - Q_i \cdot (a_i f) = 0,$$

por lo que  $Q_i \hat{P}_i$  anula a la derivada parcial respecto de  $x_i$  de  $f$ .

Veamos ahora el recíproco, si  $\partial_i(f)$  es holónoma para todo  $i \in \mathbb{N}^n$ , entonces  $f$  es holónoma. Esto es fácil de ver, si  $\partial_i(f)$  es holónoma, por la proposición 4.2.1 sabemos que para  $0 \leq j \leq n$  existe  $P_j \in \mathbb{C}[\underline{x}] \langle \partial_j \rangle \setminus \{0\}$  que anula a  $\partial_i(f)$ , en particular, esto es cierto para  $j = i$  por lo que existe un  $P_i = \sum_{k=0}^{\text{ord}(P_i)} P_{ik} \partial_i^k$  que verifica que

$$0 = P_i \cdot \partial_i(f) = \sum_{k=0}^{\text{ord}(P_i)} P_{ik} \partial_i^k (\partial_i(f)) = \sum_{k=0}^{\text{ord}(P_i)} P_{ik} \partial_i^{k+1}(f) = 0,$$

de forma que si tomamos  $P'_i = P_i \partial_i$ , anula a  $f$  y está en  $\mathbb{C}[\underline{x}] \langle \partial_i \rangle$ . Como hemos tomado un  $i$  arbitrario, esto es cierto para  $0 \leq i \leq n$ , por lo que usando la proposición 4.2.1, podemos concluir que  $f$  es holónoma. |

Ahora necesitamos definir la transformada de Fourier y enunciar algunas propiedades de esta, esto se puede consultar en el capítulo 1 de [Duo01] y en los capítulos 5 y 6 de [SS03]. Recordemos la definición del espacio de funciones  $L^1(\mathbb{R}^n) = \{f : \mathbb{R}^n \rightarrow \mathbb{R} \mid \int_{\mathbb{R}^n} |f| < \infty\}$ .

**| Definición 4.3.7 (Transformada de Fourier).** Sea  $f \in L^1(\mathbb{R}^n)$ , definimos la transformada de Fourier de  $f$  como

$$\hat{f}(\xi) = \int_{\mathbb{R}^n} f(x) e^{-2\pi i x \xi} dx.$$

En ocasiones denotaremos a la transformada de Fourier de  $f$  como  $F(f)$  por comodidad, y denotaremos por  $F(I) = \{\hat{f} \mid f \in I\}$  a la transformada de Fourier de un ideal.

El interés en introducir la transformada de Fourier radica en la riqueza de sus propiedades, estas nos ayudarán a probar el resto de propiedades sobre las funciones holónomas que nos interesan.

**Proposición 4.3.8 (Propiedades de la transformada de Fourier).** Sean  $f, g$  funciones, se verifica que:

1.  $\widehat{\partial_j(f)} = 2\pi i x_j \widehat{f}$ .
2.  $\widehat{x_j f} = -\frac{1}{2\pi i} \partial_j(\widehat{f})$ .
3.  $\widehat{\widehat{f}}(x) = f(-x)$ . (Fórmula de inversión de Fourier).

**| Definición 4.3.9 (Transformada de Fourier-Laplace de un operador diferencial).** Definimos la transformada de Fourier-Laplace como el automorfismo de  $K$ -álgebras  $FL : D_n \rightarrow D_n$  tal que:

$$x_i \mapsto -\partial_i \text{ y } \partial_i \mapsto x_i.$$

**Proposición 4.3.10.** Sea  $I \subseteq D_n$  un ideal (a la izquierda). Entonces,  $I$  es un ideal holónomo (equivalentemente  $D_n/I$  es un módulo holónomo) si y solo si  $FL(I)$  es un ideal holónomo (o equivalentemente  $D_n/FL(I)$  es un módulo holónomo).

**Demostración.** La demostración de esta proposición es análoga a la demostración de la tercera afirmación de la proposición 4.3.1, aunque tenemos que ver un par de detalles necesarios. Primero,  $FL$  es un automorfismo filtrado de  $D_n$ , con la filtración de Bernstein, por la fórmula de inversión sabemos que es biyectiva y como intercambia  $x_i$  con  $\partial_i$ , el grado no se ve alterado, por lo que es una aplicación filtrada. Veamos entonces que es realmente un automorfismo de  $D_n$ , viendo que verifica las relaciones que caracterizan a  $D_n$ , es decir, veamos que  $FL([\partial_i, x_j]) = 1$ , para  $i \neq j$ ,

$$FL([\partial_i, x_j]) = FL(\partial_i x_j) - FL(x_j \partial_i) = -x_i \partial_j + \partial_j x_i = [\partial_j, x_i] = 1,$$

por lo que ya podemos afirmar que  $FL$  es un automorfismo de  $D_n$ .

Como hemos dicho antes, la implicación hacia la derecha es análoga a la prueba de la tercera afirmación de la proposición 4.3.1, pero en este caso si tenemos que hacer la prueba hacia la izquierda. En el caso de la transformada de Fourier-Laplace, al hacer  $FL(FL(f))$  no obtenemos  $f(x)$ , sino  $f(-x)$ , pero como ya vimos en la misma proposición 4.3.1, esta también es holónoma. |

El siguiente paso que daremos será probar que una función es holónoma si y solo si su transformada de Fourier lo es, pero para ello necesitaremos primero un lema de carácter técnico.

**Lema 4.3.11.** Sean  $\varphi$  el automorfismo de  $K$ -álgebras de  $D_n$  que verifica que  $\varphi(x_i) = 2\pi i x_i$  y  $\varphi(\partial_i) = \frac{1}{2\pi i} \partial_i$  y  $f$  una función. Un operador  $P \in D_n$  anula a  $f$  si y solo si  $(\varphi \circ FL)(P)$  anula a  $\hat{f}$ .

**Demostración.** Sean  $P = \sum_{\alpha\beta} P_{\alpha\beta} \underline{x}^\alpha \underline{\partial}^\beta \in D_n$  y una función  $f$ , tales que  $P \bullet f = 0$ . Por la proposición 4.3.8, tenemos que  $\hat{\hat{f}}(-x) = f(x)$ , por lo que

$$\begin{aligned} 0 &= \sum_{\alpha\beta} P_{\alpha\beta} \underline{x}^\alpha \underline{\partial}^\beta (\hat{\hat{f}}(-\underline{x})) = \left[ \begin{array}{l} \underline{x}' := -\underline{x} \\ \underline{\partial}' := -\underline{\partial} \end{array} \right] = \sum_{\alpha\beta} P_{\alpha\beta} (-1)^{|\alpha|+|\beta|} \underline{x}^\alpha \underline{\partial}^\beta (\hat{\hat{f}}(\underline{x})) = \\ & \quad (\text{por 4.3.8}) = \sum_{\alpha\beta} P_{\alpha\beta} (-1)^{|\alpha|+|\beta|} (-2\pi i)^{|\beta|} \underline{x}^\alpha F \left( \underline{x}^\beta \hat{\hat{f}}(\underline{x}) \right) = \\ & \quad (\text{por 4.3.8}) = \sum_{\alpha\beta} P_{\alpha\beta} (-1)^{|\alpha|+|\beta|} (-2\pi i)^{|\beta|} \left( \frac{1}{2\pi i} \right)^{|\alpha|} F \left( \underline{\partial}^\alpha \underline{x}^\beta \hat{\hat{f}}(\underline{x}) \right) = \\ &= \left[ \begin{array}{l} \underline{x}' := 2\pi i \underline{x} \\ \underline{\partial}' = \frac{1}{2\pi i} \underline{\partial} \end{array} \right] = \sum_{\alpha\beta} P_{\alpha\beta} F \left( (-\underline{\partial})^\alpha \underline{x}^\beta \hat{\hat{f}} \left( \frac{\underline{x}}{2\pi i} \right) \right) = F \left( \sum_{\alpha\beta} P_{\alpha\beta} (-\underline{\partial})^\alpha \underline{x}^\beta \hat{\hat{f}} \left( \frac{\underline{x}}{2\pi i} \right) \right) \end{aligned}$$

Podemos deducir que  $FL(P) = \sum_{\alpha\beta} P_{\alpha\beta} (-\underline{\partial})^\alpha \underline{x}^\beta \in \text{ann}_{D_n}(\varphi^{-1}(\hat{\hat{f}})) = \varphi^{-1}(\text{ann}_{D_n}(\hat{\hat{f}}))$ , que es verdad si y solo si  $(\varphi \circ FL)(P) \in \text{ann}_{D_n}(\hat{f})$ . |

| **Teorema 4.3.12.**  $f$  es holónoma si y solo si su transformada de Fourier es holónoma.

**Demostración.** Si  $f$  es holónoma, como vimos en la prueba de la proposición 4.2.1, para todo  $i \leq n$  existe un  $P_i \in \mathbb{C}[x_i] \langle \partial_1, \dots, \partial_n \rangle$  tal que  $P_i \bullet f = 0$ . Por la proposición anterior,  $(\varphi \circ FL)(P_i)$  anula a  $\hat{\hat{f}}$ , y además,  $(\varphi \circ FL)(P_i) \in \mathbb{C}[\underline{x}] \langle \partial_i \rangle$ , por lo que aplicando otra vez la proposición 4.2.1,  $\hat{\hat{f}}$  es holónoma.

Para la otra implicación podemos hacer exactamente lo mismo, y llegaríamos, por la proposición 4.3.8, a que  $f(-x)$  es holónoma, pero por la proposición 4.3.1  $f$  es holónoma. |

Con esto ya probado, podremos probar el recíproco del primer apartado de la proposición 4.2.1.

**Corolario 4.3.13 (Corolario proposición 4.2.1).**  $f$  es una función holónoma si y solo si para todo  $i \leq n$  existe un  $P_i \in \mathbb{C}[x_i] \langle \partial_1, \dots, \partial_n \rangle$  tal que  $P_i \bullet f = 0$ .

**Demostración.** La implicación de izquierda a derecha esta probada en la proposición 4.2.1. Solo tendremos que probar la otra. Supongamos que para todo  $i \leq n$  existe un  $P_i \in \mathbb{C}[x_i]\langle \partial_1, \dots, \partial_n \rangle$  tal que  $P_i \bullet f = 0$ , entonces como  $FL(P_i) \in \mathbb{C}[\underline{x}]\langle \partial_i \rangle$  y anula a  $\hat{f}$ , tenemos que  $\hat{f}$  es holónoma, y por el teorema anterior,  $f$  es holónoma. |

Ya tenemos todos los resultados y consecuencias de la transformada de Fourier que necesitábamos, ahora podemos volver a ver cómo se conserva la holonomía.

**Lema 4.3.14.** Sea  $n = 1$ . La integral indefinida de una función holónoma es holónoma.

**Demostración.** Sea  $f$  holónoma. Por el teorema fundamental del análisis tenemos que existirá un  $F(x) = \int_0^x f(t)dt$  que verificará que  $F'(x) = f(x)$ . Dado que estamos en una variable, esto es equivalente a decir que todas las derivadas de  $F$  son holónomas, por lo que la proposición 4.3.6 nos afirma que  $F$  será holónoma. |

**Proposición 4.3.15.** Sea  $f : \mathbb{R}^n \rightarrow \mathbb{C}$ . Su integral definida,

$$F(x_1, \dots, x_{n-1}) = \int_a^b f(x_1, \dots, x_n) dx_n$$

es una función holónoma en  $n - 1$  variables (asumiendo que la integral existe).

**Demostración.** Sea  $f$  holónoma, entonces, como vimos en la demostración de la proposición 4.2.1, para todo  $i \leq n$  existirá un  $P_i \in \mathbb{C}[x_i]\langle \partial_1 \dots \partial_n \rangle$  tal que  $P_i \bullet f = 0$ . Este  $P_i$  lo podemos escribir como  $P_i = Q_i - \partial_n R_i$  donde  $Q_i \in \mathbb{C}[x_i]\langle \partial_1, \dots, \partial_{n-1} \rangle$  y  $R_i \in \mathbb{C}[x_i]\langle \partial_1, \dots, \partial_n \rangle$ . De esta forma, tenemos que

$$Q_i \bullet F = \int_a^b Q_i \bullet f dx_n = \int_a^b P_i \bullet f dx_n + \int_a^b \partial_n R_i \bullet f dx_n = [R_i \bullet f]_{x_n=a}^{x_n=b}.$$

Por las proposiciones 4.3.3 y 4.3.4  $[R_i \bullet f]_{x_n=a}^{x_n=b}$  es una función holónoma en las variables  $x_1, \dots, x_{n-1}$ , por ser la diferencia de dos restricciones, por lo que existe  $\tilde{Q}_i \in \mathbb{C}[x_i]\langle \partial_1, \dots, \partial_{n-1} \rangle$  que anula a  $[R_i \bullet f]_{x_n=a}^{x_n=b}$ . Si consideramos  $\tilde{Q}_i Q_i \in \mathbb{C}[x_i]\langle \partial_1, \dots, \partial_{n-1} \rangle$ ,

$$(\tilde{Q}_i Q_i) \bullet F = \tilde{Q}_i \bullet (Q_i \bullet F) = \tilde{Q}_i \bullet ([R_i \bullet f]_{x_n=a}^{x_n=b}) = 0.$$

Hemos encontrado un operador en  $\mathbb{C}[x_i]\langle \partial_1, \dots, \partial_{n-1} \rangle$  que anula a  $F$ , para todo  $i = 1, \dots, n - 1$ , por lo que  $F$  es holónoma. |



## 5 | Cálculos

En este capítulo utilizaremos el desarrollo teórico que hemos hecho hasta ahora para calcular el área (o volumen) de ciertas regiones delimitadas por polinomios. Estudiaremos el caso del área de una pantalla de televisión, que se corresponde con [SS19, Example 2.5], calcularemos el volumen del toro de radio interior 1 y radio exterior 2, siguiendo los pasos de [LMD19], y por último calcularemos el volumen de un astroide y de un *pretzel*, ambos ejemplos proporcionados por Marc Mezzarobba.

Por no recargar el trabajo de código, solo incluiremos en el mismo los programas para la resolución del área de la pantalla de televisión y el volumen del *pretzel*. El código de SageMath en su integridad se puede consultar en [MB21].

Como detalle anecdótico, el tiempo de ejecución de todo el archivo de SageMath es de aproximadamente 7 minutos y 30 segundos, es decir, SageMath tarda 7 minutos en hacer todos los cálculos que explicaremos en este capítulo. Estos cálculos los hemos hecho en mi portátil personal, en otro ordenador y otro momento, los tiempos que reflejamos en el trabajo podrían no coincidir.

### 5.1 Área de la pantalla de una televisión

Sea

$$q(x, y) = x^4 + y^4 + \frac{1}{100}xy - 1, \quad (5.1)$$

nuestro objetivo es calcular el área del conjunto  $S = \{(x, y) \in \mathbb{R}^2 \mid q(x, y) \leq 0\}$ . Si dibujamos el borde de nuestro conjunto  $S$ , podemos ver que  $S$  está contenido en el cuadrado  $-1, 2 \leq x, y \leq 1, 2$ . Sabiendo esto, una primera aproximación para calcular el área es hacer un muestreo aleatorio y ver que porcentaje del área representa nuestra región. En nuestro caso, hemos hecho un muestreo con 20000 puntos elegidos aleatoriamente en SageMath, y hemos obtenido que el área es aproximadamente

3,700512. En la figura 5.1 están representados en cian los puntos de  $S$  y en coral los que no.

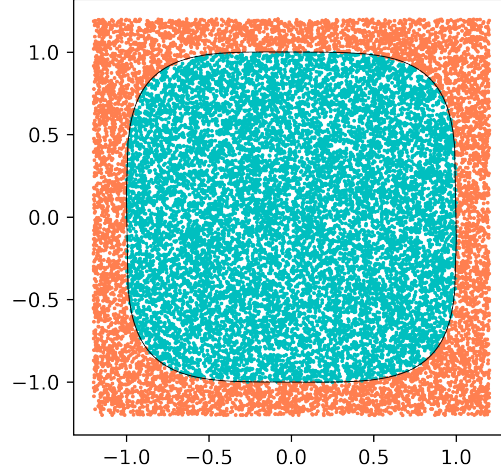


Figura 5.1: Muestreo aleatorio

Nuestro objetivo ahora es calcular el área de  $S$  usando  $D$ -módulos. Definimos  $\pi : S \rightarrow \mathbb{R}$ , la proyección de los puntos de  $S$  sobre el eje  $x$ , y  $v(x) = \ell(\pi^{-1}(x))$ , es decir, la longitud del segmento  $\pi^{-1}(x)$ . Esta función,  $v$ , es holónoma, ya que si definimos  $y_1(x)$  e  $y_2(x)$  como la mayor y la menor solución entre las dos soluciones reales de  $q(x, y) = 0$  respectivamente, ambas son holónomas, por la proposición 4.3.2, y  $v(x) = y_1(x) - y_2(x)$ , por lo que también es holónoma, por la proposición 4.3.3.

Para calcular los puntos extremos de la figura usaremos que son los valores mínimo y máximo de  $x$ . Si expresamos  $x$  en función de  $y$ , lo que buscamos son puntos en los que  $x' = 0$ , consideremos ahora  $x = x(y)$  y derivemos la expresión de  $q(x(y), y) = 0$  para ver que obtenemos

$$\partial q / \partial y = 4x'(y)x^3(y) + 4y^3 + x(y)/100 + x'(y)y/100 = 0,$$

y como queremos que  $x'(y) = 0$ , entonces llegamos a la condición

$$4y^3 + \frac{1}{100}x = 0 \rightarrow x = -400y^3.$$

Sustituyendo ahora en  $q(x, y)$ ,  $y$  por  $T$  y  $x$  por  $-400T^3$ , solo nos queda resolver la ecuación  $q(T) = 0$  y ver cuánto vale  $T$ , esto podemos calcularlo directamente con



SageMath que nos devuelve que las soluciones reales son

$$\pm 1,0002544658502588454785457666435667500801962761590\dots$$

Tomemos  $x_0$  como la solución negativa, y  $x_1$  la positiva. El área que queremos calcular es  $\text{vol}(S) = w(x_1)$ , donde  $w$  viene definida por

$$w(x) = \int_{x_0}^x v(t) dt.$$

Busquemos un operador que anule a este  $w$ , de la forma  $P\partial$  donde  $P$  anule a  $v$ . Este  $P$  se puede calcular cómodamente usando el paquete `HolonomicFunctions`, para Mathematica, creado por C. Koutschan [Kou10], de la siguiente forma:

```
In [1]: << RISC`HolonomicFuntions`
Out[1]: HolonomicFunctions Package version 1.7.3 (21-Mar-2017)
        written by Christoph Koutschan
        Copyright Research Institute for Symbolic Computation
        (RISC), Johannes Kepler University, Linz, Austria

In [2]: q = y^4 + x^4 + x*y/100 - 1;
        ann = FullSimplify[Annihilator[Root[q,y,1],Der[x]]]
Out[2]: {(1+2x^4)^2(-25600000000+76799999973x^4+
        25600000000x^8(-3+x^4))D_x^3+6x^3(1+2x^4)(179199999973-
        307199999946x^4+25600000000x^8(3+2x^4))D_x^2+3x^2
        (307199999953+4x^4(-876799999861+723199999893x^4+
        25600000000x^8(2+x^4)))D_x-3x(51199999993+4x^4
        (-262399999957+364799999949x^4+25600000000x^8(2+x^4)))}
```

Entonces tomando

$$\begin{aligned} P = & (1 + 2x^4)^2(-25600000000 + 76799999973x^4 + 25600000000x^8(-3 + x^4))\partial^3 \\ & + 6x^3(1 + 2x^4)(179199999973 - 307199999946x^4 + 25600000000x^8(3 + 2x^4)) \\ & \partial^2 + 3x^2(307199999953 + 4x^4(-876799999861 + 723199999893x^4 + \\ & 25600000000x^8(2 + x^4)))\partial - 3x(51199999993 + 4x^4(-262399999957 \\ & + 364799999949x^4 + 25600000000x^8(2 + x^4))), \end{aligned}$$

tenemos que  $P\partial$  anula a  $w$ . Para conseguir un problema de contorno del que  $w$  sea solución necesitamos algunas condiciones iniciales. Como  $\text{ord}(P) = 3$  entonces

$\text{ord}(P\partial) = 4$ , por lo que necesitaremos 4 condiciones. Por definición de  $w$  tenemos que  $w(x_0) = 0$ , esa será nuestra primera condición inicial. El resto de condiciones las impondremos a partir de la derivada,  $w'(0) = v(0) = 2$ , y las otras dos las podemos obtener dando valores a  $q(x, y)$ , si  $x = 1$  entonces  $\pi^{-1}(1) = \{-\frac{1}{\sqrt[3]{100}}, 0\}$  por lo que  $w'(1) = v(1) = 0 - (-1/\sqrt[3]{100}) = 1/\sqrt[3]{100}$ . Si  $x = -1$ ,  $\pi^{-1}(-1) = \{\frac{1}{\sqrt[3]{100}}, 0\}$  entonces  $w'(-1) = v(-1) = 1/\sqrt[3]{100} - 0 = 1/\sqrt[3]{100}$ . Ya tenemos nuestras cuatro condiciones iniciales:

$$w(x_0) = 0, \quad w'(0) = 2 \quad \text{y} \quad w'(1) = w'(-1) = 1/\sqrt[3]{100}.$$

Nuestro problema se reduce ahora a buscar un  $w \in \text{Sol}(P\partial)$  que verifique estas condiciones, para ello, hallemos una base de  $\text{Sol}(P\partial)$ . En realidad lo que hallaremos será una base local de  $\text{Sol}(P\partial)$  con respecto a un  $x_{ord} \in \mathbb{R}$ , un punto que no anule al coeficiente líder de  $P$ . Este proceso puede programarse usando el paquete `ore_algebra` [KJJ13] en SageMath, obteniendo que una base local centrada en  $x_{ord}$  será de la forma:

$$\begin{aligned} s_{x_{ord},0}(x) &= 1 + \mathcal{O}((x - x_{ord})^4), \\ s_{x_{ord},1}(x) &= (x - x_{ord}) + \mathcal{O}((x - x_{ord})^4), \\ s_{x_{ord},2}(x) &= (x - x_{ord})^2 + \mathcal{O}((x - x_{ord})^4), \\ s_{x_{ord},3}(x) &= (x - x_{ord})^3 + \mathcal{O}((x - x_{ord})^4). \end{aligned} \tag{5.2}$$

Teniendo esta base, ya podemos afirmar que, localmente en  $x_{ord}$ , nuestra solución será de la forma

$$w(x) = c_{x_{ord},0} \cdot s_{x_{ord},0}(x) + c_{x_{ord},1} \cdot s_{x_{ord},1}(x) + c_{x_{ord},2} \cdot s_{x_{ord},2}(x) + c_{x_{ord},3} \cdot s_{x_{ord},3}(x) + \mathcal{O}((x - x_{ord})^4)$$

El problema que se nos plantea ahora es que nuestras condiciones iniciales están definidas en puntos distintos, por este motivo, vamos a necesitar el uso de matrices de transición. Con estas matrices vamos a poder pasar nuestro polinomio respecto de  $x_{ord}$  a otro punto que nos interese. Estas matrices podemos calcularlas con el comando `numerical_transition_matrix` del paquete `ore_algebra`.

La matriz de transición de  $x_{ord}$  a  $\hat{x}$  nos va a permitir obtener una base local sobre la cual podemos expresar nuestro polinomio en un entorno de  $\hat{x}$ , a partir de la base que tenemos en  $x_{ord}$ . Es decir, si conocemos los coeficientes  $c_{x_{ord},i}$ , podemos obtener los coeficientes  $c_{\hat{x},i}$  multiplicando los primeros por la matriz de transición de  $x_{ord}$  a  $\hat{x}$ . Nuestro algoritmo se basará en partir de un  $x_{ord} \in \mathbb{R}$ , aplicarle la matriz de transición de dicho  $x_{ord}$  a un cierto  $\hat{x} \in \mathbb{R}$ , para el cual tengamos una condición de contorno, que

podremos imponer tras aplicar la matriz de transición, pues obtenemos una expresión local en  $\hat{x}$  de nuestra  $w$ .

En este caso, tomando  $x_{ord} = 0$ , dado que  $w'(0) = 2$ , tenemos que

$$w'(x) = c_{0,1} + 2c_{0,2}x + 3c_{0,3}x^2 + \mathcal{O}(x^3), \text{ entonces } w'(0) = c_{0,1} = 2.$$

De igual forma para el resto de condiciones  $c_{x_0,0} = 0$  y  $c_{1,1} = c_{-1,1} = 1/\sqrt[3]{100}$ . En nuestro caso, calcularemos las matrices de transición de 0 al resto de puntos, 0 jugará el papel de  $x_{ord}$ , y nuestras incógnitas serán los coeficientes  $c_{0,i}$ . Una vez hayamos calculado estos, solo tendremos que hallar la matriz de transición de 0 a  $x_1$  y podremos expresar  $w$  localmente en  $x_1$ , es más dado que  $s_{x_1,1}(x_1) = s_{x_1,2}(x_1) = s_{x_1,3}(x_1) = 0$ , entonces  $w(x_1) = c_{x_1,0}$ .

Con este procedimiento podemos obtener el área de nuestra región con la precisión que queramos, y obtenemos

$$w(x_1) \approx 3,70815994474216228834822556114586537124306581991393470943857 \\ 21323867607641628396904239116597636019645$$

Con una precisión de 100 dígitos, podemos asegurar que todos ellos son correctos. Todo el código utilizado para estos cálculos se puede consultar en la sección 5.3.1.

Algo que merece la pena recalcar es que este procedimiento es verdaderamente interesante desde el punto de vista computacional, ya que un aumento de la cantidad de dígitos de precisión no aumenta el tiempo de cálculo del área. Para ilustrar esto hemos calculado el área de nuestra región con diferente cantidad de dígitos de precisión en SageMath y en Mathematica, en este segundo mediante el comando

```
Area[ImplicitRegion[x^4 + y^4 + x*y/100 <= 1, {x, y}]], digitos]
```

donde *digitos* es la cantidad de dígitos de precisión que queríamos obtener, y hemos anotado tanto el tiempo de ejecución como el error absoluto, las figuras 5.2 y 5.3 reflejan la comparativa.

Como podemos ver en ambas gráficas, el proceso descrito en este trabajo es mucho más preciso y bastante más práctico a la hora del cálculo con el menor error posible la solución de nuestro problema. Como dato interesante, para calcular 550 dígitos nuestro proceso tarda unos 6 minutos, el otro dato curioso es que el error que cometemos es como mucho del orden de  $10^{-(\text{digitos}-1)}$ , donde *digitos* es la cantidad de dígitos de precisión que exigimos.

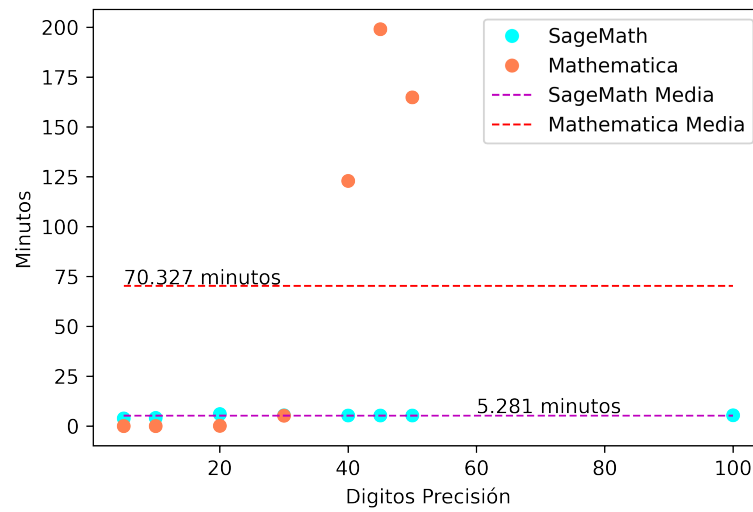


Figura 5.2: Tiempo de ejecución (en minutos) por dígitos de precisión.

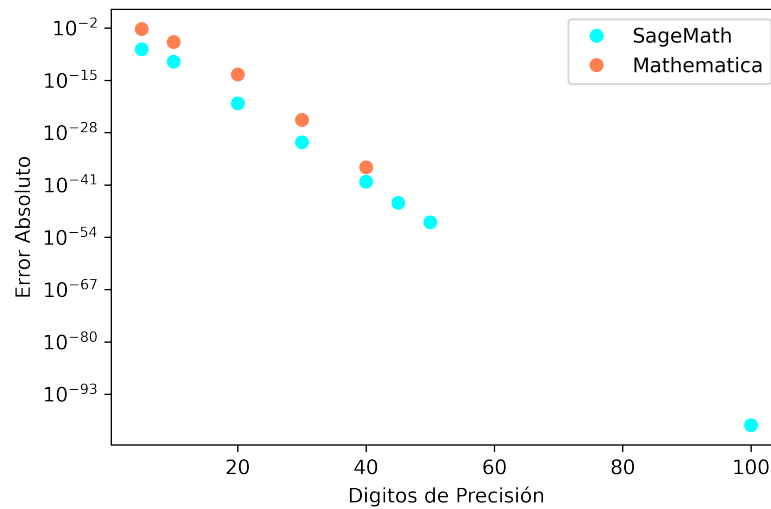


Figura 5.3: Error absoluto por dígitos de precisión.

El procedimiento descrito en esta sección es totalmente general, dada una región representada por una función holónoma, podremos hallar su área siguiendo el mismo procedimiento, particularizando los datos de acuerdo a nuestra nueva región.

## 5.2 Volúmenes

Ya hemos visto cómo calcular el área de una región bidimensional, ahora usaremos eso para calcular el volumen de una región en tres dimensiones. El razonamiento que seguiremos será exactamente el mismo, tendremos una ecuación, en este caso en tres variables, que delimita una región de  $\mathbb{R}^3$ , y querremos calcular el volumen que engendra dicha región. Supongamos que nuestra región viene dada por

$$S = \{(x, y, z) \in \mathbb{R}^3 \mid q(x, y, z) \leq 0\},$$

al igual que antes definimos  $\pi : S \rightarrow \mathbb{R}$  como la proyección de un valor  $x$  sobre el eje  $y = z = 0$ , y tomamos  $v(x) = a(\pi^{-1}(x))$ , que en este caso en vez de ser la longitud de un segmento, es el área de un corte de nuestra región. Entonces el volumen que buscamos será  $w(x_1)$  donde

$$w(x) = \int_{x_0}^x v(t) dt,$$

y  $x_0$  y  $x_1$  son el máximo y mínimo de  $x$ , respectivamente.

Si  $v$  es holónoma, existirá un operador diferencial que la anule, llamémoslo  $P$ , como  $P \bullet v = 0$ , tendremos que  $P\partial$  anula a  $w$ . Podemos transformar nuestro problema en buscar una solución  $w \in \text{Sol}(P\partial)$  verificando unas condiciones iniciales que debaremos calcular e imponer. Siempre tendremos que  $w(x_0) = 0$ , el resto las obtendremos de que  $w'(x) = v(x)$ , y aplicando el procedimiento que describimos al calcular el área de la pantalla de la televisión, podremos calcular estos  $v(x)$  para la cantidad de valores que necesitemos, que será el orden de  $P$ .

Una vez tengamos las condiciones de contorno necesarias, tendremos que calcular las matrices de transición para llevar una base local de  $w$  en  $x_{ord}$  a los puntos en los que hayamos impuesto las condiciones y obtendremos una base local de  $w$  en  $x_1$ , donde el coeficiente de grado 0 será el valor del volumen que buscamos. Como podemos apreciar para el cálculo del volumen de una región tridimensional procedemos de la misma forma que para el cálculo del área de una región bidimensional, con la única diferencia de que en nuestro caso actual las condiciones de contorno son áreas de secciones y en el caso anterior eran longitudes de curvas, pero implementando el algoritmo descrito anteriormente, computacionalmente será el mismo proceso.

En esta sección calcularemos el volumen de un toro, tomando como referencia [LMD19], el volumen de un astroide y un *pretzel*, ambos ejemplos proporcionados por Marc Mezzarobba, y comprobaremos los resultados obtenidos tanto con

Mathematica, como con el programa desarrollado por Pierre Lairez, Marc Mezzarobba y Mohab Safey El Din en [LMD19].

### 5.2.1 Volumen de un toro

El primer volumen que calcularemos será el de un toro con radio interior 1 y radio exterior 2, la región vendrá dada por la función

$$q(x, y, z) = (x^2 + y^2 + z^2 + 3)^2 - 16(x^2 + y^2),$$

es decir, la región será

$$T = \{(x, y, z) \in \mathbb{R}^3 \mid q(x, y, z) \leq 0\}.$$



Figura 5.4: Toro  $T$ .

Como hemos explicado en la introducción a la sección, ahora deberemos definir  $v(x) = a(\pi^{-1}(x))$ , donde  $\pi$  será la proyección del toro sobre el eje  $y = z = 0$ , y debemos comprobar que esta función es holónoma. Por la proposición 4.2.1, si encontramos un operador diferencial en  $\mathbb{R}[x, y, z]\langle \partial_x \rangle$  que anule a  $v$  habremos acabado, este operador lo podemos obtener de [LMD19], y es

$$P = (x - 3)(x + 3)(x - 1)^2(x + 1)^2x^2\partial_x^3 + (x^2 + 9)(x - 1)^2(x + 1)^2x\partial_x^2 - (2x^4 + 11x^2 - 9)(x - 1)(x + 1)\partial_x + 2(x^2 + 3)x^3.$$

Sean  $x_0$  y  $x_1$  el mínimo y el máximo valor de  $x$ , respectivamente, en  $T$ , que sabemos que son  $-3$  y  $3$  respectivamente, tomamos

$$w(x) = \int_{x_0}^x v(t)dt,$$

lo que queremos calcular es  $w(x_1)$ . Como  $P$  anula a  $v$ , entonces  $P \bullet \partial_x$  anula a  $w$ . Ya tenemos la ecuación diferencial que queremos resolver,  $(P\partial_x) \bullet w = 0$ , este operador es de orden 4, porque  $\text{ord}(P) = 3$ , por lo que el siguiente paso será encontrar 4 condiciones de contorno. La primera está clara,  $w(x_0) = 0$ . Impondremos el resto sobre la derivada, ya que  $w'(x) = v(x)$ , y aplicando el razonamiento visto para calcular el área de la pantalla de la televisión, podremos calcular estos  $v(x)$ .

Como hemos dicho, siguiendo los pasos anteriores, podemos hallar las condiciones en  $x = 0$ ,  $x = 1/2$  y  $x = -1/2$ , estas dos últimas coincidirán, no incluimos los resultados obtenidos por no recargar la explicación, se pueden consultar en [MB21]. Una vez hemos hallado el área de ambas secciones, las llamaremos  $A_0$  y  $A_{1/2}$ , respectivamente, llegamos al sistema diferencial siguiente:

$$\left\{ \begin{array}{l} (P\partial_x) \bullet w = 0 \\ w(x_0) = 0 \\ w'(0) = A_0 \\ w'(\pm 1/2) = A_{1/2}. \end{array} \right.$$

Si calculamos los puntos singulares de nuestro operador diferencial, veremos que 0 es uno de ellos. Aún así, es necesario considerarlo como nuestro  $x_{ord}$ , ya que, de no hacerlo, la solución que obtenemos no es correcta, esto lo comprobamos en el código, que se puede consultar en [MB21]. No sabemos exactamente por qué ocurre esto, no hemos sido capaces de encontrar una solución a este problema tampoco.

Aunque el cero sea singular, sí verifica que una base local en un entorno de él tiene la forma que buscamos, es decir, obtenemos que nuestra función, localmente en 0, es de la forma

$$w(x) = c_{0,0} + c_{0,1}x + c_{0,2}x^2 + c_{0,3}x^3 + \mathcal{O}(x^4),$$

y aplicando matrices de transición de la misma forma que antes llegamos a un sistema lineal cuyas soluciones son los  $c_{0,i}$ , y con ellos y la matriz de transición de 0 a  $x_1$  obtenemos los coeficientes de la base local en  $x_1$ , y con ello el volumen de  $T$ . Aplicando

este procedimiento en SageMath obtenemos que

$$w(x_1) \approx 39,4784176043574344753379639995046045412547976289631625056533 \\ 97504880179289676820972007093614874208925,$$

con 100 dígitos de precisión, podemos comprobar que coincide con  $4\pi^2$  que es el volumen que sabemos que tiene este toro. Nuestro programa tarda unos 22 segundos en hacer el cálculo completo, incluyendo las condiciones de contorno.

### 5.2.2 Volumen de un astroide

El siguiente ejemplo también nos lo proporcionó Marc Mezzarobba. Se trata de un astroide delimitado por la función

$$q(x, y, z) = x^2 + y^2 + z^2 - (1 - 2^{10}(x^2y^2 + y^2z^2 + z^2x^2)),$$

por lo que nuestra región en este caso es

$$A = \{(x, y, z) \in \mathbb{R}^3 \mid q(x, y, z) \leq 0\}.$$

De forma idéntica a como procedimos anteriormente, definimos  $\pi$  como la proyección de  $x$  sobre el eje  $y = z = 0$ , definimos  $v(x) = a(\pi^{-1}(x))$  y  $w(x) = \int_{x_0}^x v(t)dt$ , donde  $x_0$  es el mínimo valor que toma  $x$  en  $A$ , y a su vez,  $x_1$  será el máximo. Para calcular  $x_0$  y  $x_1$ , vemos donde se anulan  $\text{Res}(q, \partial q / \partial_y)$ ,  $\text{Res}(q, \partial q / \partial_z)$  y  $q$  a la vez, y obtenemos que  $x_0 = -1$  y  $x_1 = 1$ .

Ahora queremos calcular el valor  $w(x_1)$ , y para ello primero necesitamos un operador diferencial que anule a  $w$ , este será de la forma  $P\partial_x$ , donde  $P \bullet v = 0$ . Para calcular dicho  $P$  necesitaríamos acceso al software Magma, el cual no tenemos, por lo que junto con el ejemplo Marc Mezzarobba, nos facilitó dicho  $P$ , no lo añadimos al trabajo por su longitud, el lector interesado lo puede encontrar en el archivo de SageMath de [MB21]. En este caso,  $\text{ord}(P) = 3$ , por lo que necesitaremos imponer 4 condiciones de contorno, la primera, como siempre es  $w(x_0) = 0$ , y las restantes las impondremos sobre la derivada.

Con la astroide ocurre lo mismo que con el toro, 0 es un punto singular de nuestro operador diferencial, pero necesitamos tomarlo como  $x_{ord}$  para obtener el resultado correcto. Como dijimos anteriormente, no sabemos por qué sucede esto. Aún así, en el origen volvemos a tener que una base local es de la forma que buscamos, por lo que podemos, *a priori*, proceder como describimos anteriormente.





Figura 5.5: Astroide A.

Impondremos las condiciones sobre  $x = 0$ ,  $x = 1/2$  y  $x = -1/3$ , obteniendo  $A_0$ ,  $A_{1/2}$  y  $A_{-1/3}$  respectivamente, al igual que antes, omitimos poner estos resultados por su longitud, pero se pueden consultar en [MB21]. De esta forma, obtenemos el problema de contorno

$$\left\{ \begin{array}{l} (P\partial_x) \cdot w = 0 \\ w(x_0) = 0 \\ w'(0) = A_0 \\ w'(1/2) = A_{1/2} \\ w'(-1/3) = A_{-1/3}. \end{array} \right.$$

Y al igual que antes, tomamos  $x_{ord} = 0$  y usando las matrices de transición necesarias llegamos a que

$$w(x_1) \approx 0,10857542146036093773950339599420761981091787444660747544447 \\ 582299328536067303292819494347441406406678,$$

con 100 dígitos de precisión, tardando en torno a 31 segundos en hacer todo el cálculo. Usando el programa desarrollado por los autores de [LMD19], obtenemos en segun-

dos, que el volumen es aproximadamente

0,10857542146036093773950339599420761981091787444660747544447  
582299328536067303292819494347441406406613,

por lo que podemos comprobar que nuestra solución coincide, con la suya, con un error del orden de  $10^{-98}$ . Por el contrario, Mathematica tarda unos 5 minutos en calcular la solución con una precisión de 25 dígitos, obteniendo como solución

0,1085754214603609562682925,

cometiendo un error del orden de  $10^{-16}$ , por lo que ambos métodos algebraicos son mucho más rápidos y proporcionan mayor precisión.

### 5.2.3 Volumen de un pretzel

El último ejemplo de este trabajo será hallar el área encerrada por un pretzel, nuestra



Figura 5.6: Pretzel  $B$ .

región está delimitada por la función

$$q(x, y, z) = (x^2(1 - x^2) - y^2)^2 + z^2 - 1/100,$$

y dicha región será

$$B = \{(x, y, z) \in \mathbb{R}^3 \mid q(x, y, z) \leq 0\}.$$

Como hasta ahora, definimos la proyección de  $x$  sobre  $y = z = 0$ ,  $\pi$ , definimos  $v(x) = a(\pi^{-1}(x))$  y  $w(x) = \int_{x_0}^x v(t)dt$ , donde  $x_0$  será el mínimo valor que toma  $x$ , que en este caso es  $x_0 = -\sqrt{1/2 + \sqrt{35}/10}$  y  $x_1 = -x_0$  el máximo valor que alcanza  $x$  en  $B$ . Queremos calcular  $w(x_1)$ , para ello buscamos un operador diferencial que anule a  $w$  de la forma  $P\partial_x$ , donde  $P \bullet v = 0$ . Este  $P$  también nos lo facilitó Marc Mezzarobba, se puede consultar en 5.3.2. Vuelve a ser de orden 3, por lo que necesitamos 4 condiciones de contorno, la primera, como siempre es  $w(x_0) = 0$ , y el resto las obtenemos de que  $w'(x) = v(x)$ . En este caso tenemos que en  $1/3$  y  $-1/3$  vale  $A_{1/3}$  y en  $1/5$  vale  $A_{1/5}$ , esto se puede consultar con más detalle en la sección 5.3.2.

Imponiendo las condiciones que acabamos de mencionar, y aplicando las matrices de transición necesarias, obtenemos que

$$w(x_1) \approx 0,19716703100601195200543797720540697966824110671375940268322685211275869905250724231224199323252392090,$$

con una precisión de 100 dígitos, tardando entorno a 34 segundos en el cálculo total. De igual forma, usando el programa desarrollado por [LMD19], en menos de un minuto afirma que el volumen es aproximadamente

$$0,19716703100601195200543797720540697966824110671375940268322685211275869905250724231224199323252392089,$$

por lo que nuestro resultado tiene un error del orden de  $10^{-99}$ . Por otro lado Mathematica tarda unos 6 minutos y medio en devolvernos un valor con una precisión de 25 dígitos,

$$0,1971670302742182190175848,$$

que tiene un error del orden de  $10^{-9}$ . Una vez más comprobamos que el método algebraico es mucho más interesante que el mero cálculo.

## 5.3 Código

Como hemos dicho anteriormente, solo incluiremos el código del área de la pantalla de televisión y del volumen del pretzel, por no recargar el trabajo demasiado. Los otros dos ejemplos son análogos al del pretzel y se pueden consultar en [MB21].

### 5.3.1 Área de la pantalla de una televisión

Primero debemos importar algunos paquetes que necesitaremos durante esta parte del trabajo.

```
[1]: from matplotlib.pyplot import plot, savefig, subplots
      from random import random
      from numpy import linspace
      from time import time
      from sympy.solvers import solve as ssolve
      from sympy import Symbol

[2]: t0_archivo = time() # Calcularemos el tiempo total de ejecución
      ↪ del archivo
```

#### Cálculo del área por muestreo

La primera función que vamos a calcular es la que dada el tamaño de una muestra nos hace un muestreo y nos calcula el área relativa de nuestra región

$$S = \{(x, y) \in \mathbb{R}^2 \mid x^4 + y^4 + \frac{1}{100}xy - 1 \leq 0\}.$$

```
[3]: def muestreo(tamaño):
      t0 = time()
      contador = 0 # El contador va a ser el número de muestras
      ↪ que hallamos obtenido
      correctos = 0 # Serán el número de puntos que hemos obtenido
      ↪ en S
      while contador < tamaño:
          x = random()*2.4-1.2
          y = random()*2.4-1.2
          if x^4 + y^4 + (1/100)*x*y - 1 <= 0: # Comprobamos si
          ↪ están en el interior de S
              plot(x,y,'co',markersize=1) # Si están los pintamos
          ↪ de cian
              correctos += 1 # Y añadimos que hemos obtenido un
          ↪ acierto
          else:
```

```

        # Si lo anterior no ocurre, el punto estaba fuera,
        ↪ lo pintamos de coral entonces
        plot(x,y,'o',color = 'coral',markersize=1)
        contador += 1 # Por último aumentamos el tamaño de la
        ↪ muestra
        fraccion = correctos/tamaño # Calculamos la fracción de la
        ↪ muestra que ha acertado
        area = (2.4^2)*fraccion # Y lo multiplicamos por el área
        ↪ total del cuadrado
        t1 = time()
        # Finalmente hacemos un "pretty print" que nos devuelva los
        ↪ datos que hemos obtenido con la muestra aleatoria
        print(f""Con una muestra de tamaño {tamaño}, han acertado
        ↪ {correctos} puntos, que corresponde con un
        {round(fraccion*100,2)}% del area del cuadrado, por lo que
        ↪ obtenemos una aproximación de que el área
        de la TV es {area}.
        Calculado en {t1-t0} segundos.""")

```

Además de pintar la muestra nos gustaría poder dibujar la frontera de la curva con la que estamos trabajando, para ello necesitamos un dato que después calcularemos, que es  $x_0$ .

Sage no nos deja dibujar la curva a partir de la ecuación en forma  $q(x, y) = 0$ , por lo que tomaremos una muestra de la  $x$  con suficientes puntos como para la representación sea fiel, y resolveremos la ecuación en  $y$  dada por  $q(x, y) = 0$  para cada  $x$  fijo, con eso podremos dibujar la curva.

```

[4]: x0 = 1.
        ↪ 000254465850258845478545766643566750080196276158976351763236
        ↪ # Tomamos el x0 que conseguiremos un poco más tarde
xp = linspace(-x0,x0,200) # Tomamos una muestra en el intervalo
        ↪ de 200 puntos que serán suficientes para la representación
y = Symbol('y')
yp1 = []
yp2 = []
for x in xp:

```

```

# Dado que es una ecuación de grado 4 tendrá 4 soluciones,
↳ de las cuales sabemos que 2 serán reales,
# que son las que nos interesan
ys = ssolve(x^4+y^4+(1/100)*x*y-1,y)
y1 = ys[0]
y2 = ys[1]
yp1 += [y1]
yp2 += [y2]

```

Ya solo nos queda dibujar todo lo que tenemos, en este caso tomamos una muestra de 20000 puntos para ser lo más precisos que podamos, y que el cálculo sea moderadamente rápido.

```

[5]: fig1, ax = subplots() # Estos dos primeros comandos son
↳ simplemente para que el dibujo quede bonito
ax.set_box_aspect(1)
muestreo(20000)
plot(xp,yp1,'black',linewidth=0.5)
plot(xp,yp2,'black',linewidth=0.5)
savefig('muestreoConBorde.png',dpi=600,transparent=True) # Y
↳ este último para guardar la imagen y poder añadirla al TFG

```

Con una muestra de tamaño 20000, han acertado 12876 puntos, que  
 ↳ corresponde con  
 un 64.38% del area del cuadrado, por lo que obtenemos una  
 ↳ aproximación de que el  
 área de la TV es 3.700512000000000.  
 Calculado en 17.388654470443726 segundos.

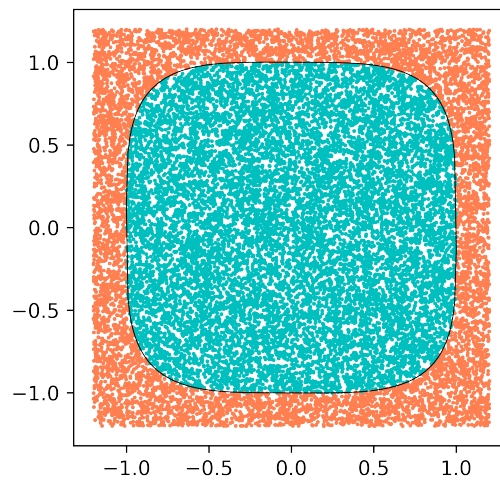
### Cálculo de los puntos extremos

```

[6]: R.<T> = QQ[] # Definimos el espacio donde trabajaremos
x,y = var('x,y') # Y nuestras variables

[7]: # Definimos las dos funciones para las que vamos a calcular la
↳ resultante
q = y^4 + x^4 + (1/100)*x*y - 1

```



```
dqy = diff(q,y)
dqy
```

```
[7]: 4*y^3 + 1/100*x
```

```
[8]: p = q.subs({x:-400*T^3, y:T}); # Dado que la derivada es nula,
    ↪despejamos y sustituimos en la ecuación de q
sols = [sol[0] for sol in R(p).roots(AA)] # Con esto conseguimos
    ↪el valor de la variable y
coord = [[-400*sol^3, sol] for sol in sols] # Coordenadas de las
    ↪soluciones
x0 = coord[1][0]
x1 = coord[0][0]
x0.n(digits=50)
```

```
[8]: -1.0002544658502588454785457666435667500801962761590
```

## Cálculo del área mediante operadores diferenciales

```
[9]: from ore_algebra import *
```

```
[10]: Dops, x, Dx = DifferentialOperators(QQ, 'x')
      digitos = 100
      precision = 10^(-digitos)
```

```
[11]: P1 = (1+2*x^4)^2*(-25600000000+76799999973*x^4+25600000000*x^8*
      (-3+x^4))*Dx^3
      P2 =  $\square$ 
       $\hookrightarrow 6*x^3*(1+2*x^4)*(179199999973-307199999946*x^4+25600000000*x^8$ 
       $* (3+2*x^4))*Dx^2$ 
      P3 = 3*x^2*(307199999953+4*x^4*(-876799999861+723199999893*x^4+
      256000000000*x^8*(2+x^4)))*Dx
      P4 = 3*x*(51199999993+4*x^4*(-262399999957+364799999949*x^4+
      256000000000*x^8*(2+x^4)))
      P = P1 + P2 + P3 - P4
```

```
[12]: (P*Dx).local_basis_expansions(1)
```

```
[12]: [1,
      (x - 1) - 3160493827/8*(x - 1)^4 - 82804938263/120*(x - 1)^5 -
      71427160505/432*(x - 1)^6,
      (x - 1)^2 + 1/12*(x - 1)^4 + 460799999941/180*(x - 1)^5 -  $\square$ 
       $\hookrightarrow 170666666573/648*(x -$ 
       $1)^6,$ 
      (x - 1)^3 + 1/2*(x - 1)^4 + 41/60*(x - 1)^5 + 155306666627/
       $\hookrightarrow 216*(x - 1)^6]$ 
```

Recordemos que tenemos expresado nuestra solución  $w$  localmente en un punto no singular  $x_{ord} \in \mathbb{R}$  como

$$w(x) = c_{x_{ord},0} + c_{x_{ord},1}(x - x_{ord}) + c_{x_{ord},2}(x - x_{ord})^2 + c_{x_{ord},3}(x - x_{ord})^3 + \mathcal{O}((x - x_{ord})^4).$$

Nuestro operador diferencial es de orden 4, por lo que necesitaremos 4 condiciones



iniciales, estas son:

$$w(x_0) = 0, \quad w'(0) = 2 \text{ y } w'(1) = w'(-1) = \frac{1}{\sqrt[3]{100}}.$$

El problema que nos presentan estas condiciones es que están en puntos distintos, por lo que necesitaremos recurrir al cálculo de matrices de transición para hallar los coeficientes de  $w$  con respecto a la base que hemos tomado. Nuestro objetivo es plantear un sistema de ecuaciones donde las soluciones sean los coeficientes en el punto  $x_{ord}$ .

Por comodidad, vamos a tomar  $x_{ord} = 0$ , veamos como se transforman las restricciones que tenemos. Para empezar, localmente en 0 nuestro polinomio es de la forma

$$w(x) = c_{0,0} + c_{0,1}x + c_{0,2}x^2 + c_{0,3}x^3.$$

La segunda condición podemos aplicarla directamente, ya que  $x_{ord} = 0$ , por lo que  $w'(0) = 2$  significa que  $c_{0,1} = 2$ . Como nuestras incógnitas son  $c_{0,0}, c_{0,1}, c_{0,2}, c_{0,3}$ , las llamaremos  $c0, c1, c2, c3$ , respectivamente, por comodidad.

Vamos a crear una función que nos calcule el área de nuestra superficie directamente una vez reciba una serie de variables necesarias:

- operador: el operador diferencial que anula a nuestra función.
- cond: una lista de tripletas  $(a, b, c)$  con las condiciones que debemos imponer, donde:
  - $a$  es la derivada de la función a la que le imponemos la condición (si  $a=0$  es la función sin derivar).
  - $b$  es el punto donde aplicamos la condición.
  - $c$  es la condición en el punto  $b$ .
- punto: es el punto en el que queremos calcular el valor de nuestra función.
- digitos: el número de dígitos de precisión que buscamos.

Este procedimiento asume que la base en cada  $x_{ord}$  está compuesta por monomios de grado ascendente, hasta el número de condiciones menos uno. Y que los coeficientes que calcularemos antes de pasar a nuestro punto es el primero de la lista `cond`, en nuestro caso el 0.

```
[13]: def area(operador,cond,punto,digitos):
```

```

t_ini = time() # Vamos a contar el tiempo que tarda en
↳ calcular la solución
precision = 10^(-digitos)
n = len(cond) # n será el número de variables con las que
↳ trabajamos
c = var(', '.join('c%s'%i for i in range(n))) # Necesitamos
↳ definir las como variables para que Sage trabaje bien con ellas
variables = vector([var for var in c]) # Pero más tarde
↳ también las necesitaremos como vector
func = [cond[i][0] for i in range(n)] # Esta lista contiene
↳ los órdenes de las derivadas
ptos = [cond[i][1] for i in range(n)] # Esta, los puntos en
↳ los que imponemos las condiciones
cond_ini = vector([cond[i][2] for i in range(n)]) # Esta,
↳ las condiciones que imponemos
x_ord = ptos[0] # Como hemos dicho antes, asumimos que
↳ queremos centrarlo todo en el primer punto
# La condición del punto en el que estamos centrados no
↳ necesita matrices de transición, así que la añadimos
↳ directamente
condiciones = [variables[func[0]]-cond_ini[0]]
# Para el resto de condiciones, como hemos explicado arriba,
↳ necesitamos una matriz de transición e imponer que
# el coeficiente correspondiente a la derivada de la función
↳ en la que imponemos la condición cumpla la condición
for j in range(1,n):
    # Cuando pasamos por un punto singular la función
↳ numerical_transition_matrix devuelve un error, una forma de
    # arreglarlo es tomar un camino en los complejos entre
↳ los puntos que queremos pero sin pasar por la singularidad
    # por ese motivo, intentamos (try) calcular la
↳ numerical_transition_matrix, y si no es posible (except) le
↳ indicamos
    # un punto del plano complejo por el cual evitaremos la
↳ singularidad
    try:

```

```

        m = operador.
↪numerical_transition_matrix([x_ord,ptos[j]],precision) #
↪Calculamos la matriz de transición
        except Exception as e:
            print(e)
            salto = complex(input("Inserte punto intermedio "))
            m = operador.
↪numerical_transition_matrix([x_ord,salto,ptos[j]],precision)
            m = m.apply_map(lambda x : x.mid()) # La ponemos de una
↪forma en la que podamos trabajar con ella
            # Añadimos la condición de que la matriz de transición
↪aplicada a nuestros coeficientes incógnita, tomando solo
            # el coeficiente que nos interesa, verifique la
↪condición que imponemos
            condiciones += [(m*variables)[func[j]]-cond_ini[j]]
            Mc = Matrix([[c.coeficient(var) for var in variables] for c
↪in condiciones]) # Obtenemos la matriz del sistema
            # Tenemos un sistema de la forma Mc*variables=cond_ini =>
↪variables = Mc-1*cond_ini
            coef_ord = (Mc.inverse())*cond_ini
            # Por último calculamos la matriz de transición de nuestro
↪punto referencia al punto al que queremos llegar
            try:
                M = operador.
↪numerical_transition_matrix([x_ord,punto],precision)
                except Exception as e:
                    print(e)
                    salto = complex(input("Inserte punto intermedio "))
                    M = operador.
↪numerical_transition_matrix([x_ord,salto,punto],precision)
                    M = M.apply_map(lambda x : x.mid())
                    # Y se la aplicamos a los coeficientes que hemos obtenido,
↪el área que buscamos será el primero
                    coef = M*coef_ord
                    t_fin = time()
                    print(f'El área de nuestra superficie es {coef[0]}.')
↪\nCalculado en {t_fin-t_ini} segundos.')

```

```
return coef[0]
```

Veamos como expresar los argumentos para que nuestra función los entienda: - Para empezar, nuestro operador es  $P * Dx$ , entonces simplemente

```
[14]: operador = P*Dx
```

- Las condiciones son

$$w(x_0) = 0, w'(0) = 2 \text{ y } w'(1) = w'(-1) = \frac{1}{\sqrt[3]{100}},$$

entonces:

- $w'(0) = 2$ , es la primera derivada en 0 y vale 2  $\Rightarrow (1, 0, 2)$ .
- $w'(1) = \frac{1}{\sqrt[3]{100}}$ , es la primera derivada en 1 y vale  $100^{-1/3} \Rightarrow (1, 1, (100)^{-1/3})$ .
- $w'(-1) = \frac{1}{\sqrt[3]{100}}$ , es la primera derivada en  $-1$  y vale  $100^{-1/3} \Rightarrow (1, -1, (100)^{-1/3})$ .
- $w(x_0) = 0$ , es la derivada 0 en  $x_0$  y vale 0  $\Rightarrow (0, x_0, 0)$ .

Ponemos la primera la de 0, porque queremos centrarnos en el 0 por comodidad, con el resto sería igual. Además vamos a calcular  $(100)^{-1/3}$  con un poco más de precisión para reducir al máximo el error cometido.

```
[15]: s = (T^3-1/100).roots(AA)[0][0]
```

Ya podemos crear el vector cond:

```
[16]: cond = [(1,0,2),(1,1,s),(1,-1,s),(0,x0,0)]
```

- El punto es  $x_1$ , que ya está calculado.
- Y por último, los dígitos que también los tenemos ya expresados.

Ya solo nos queda juntarlo todo y pedirle a Sage que calcule el área.

```
[17]: TV = area(operador,cond,x1,digitos)
```

El área de nuestra superficie es 3.

↪ 708159944742162288348225561145865371243065819913934709438572132

```
3867607641628396904239116597636019645 + 8.
↪ 878078476675739269332816726572014424907901075093661621946864721
2927823727098095784034723386936110064 e-625*I.
Calculado en 226.98939275741577 segundos.
```

### 5.3.2 Volúmenes

#### Funciones previas

```
[18]: Dop, t, D = DifferentialOperators(QQ, 't')
```

```
[19]: R.<T> = QQ[] # Definimos el espacio donde trabajaremos
      x,y,z = var('x,y,z') # Y nuestras variables
```

```
[20]: precision = 100
```

Primero creamos una función que nos devolverá los extremos de una función dada, esto nos hará mucho más cómodo la selección de los puntos en los que imponer las condiciones de nuestra ecuación diferencial.

```
[21]: def extremos(funcion):
      f = funcion
      dfz = diff(f,z)
      res = f.resultant(dfz,z)
      res = res.subs({y:T})
      sols = [root[0] for root in R(res).roots(AA)]
      y0 = min(sols)
      y1 = max(sols)
      print('y0 = ',y0,'\ny1 = ',y1)
      return y0,y1
```

Para el cálculo del área de una región, con el método que estamos aplicando, necesitamos calcular la longitud de la preimagen para ciertos valores. Por comodidad he creado una función que dado dichos valores nos halle la longitud de dicho segmento.

Esta función lo devuelve de una forma coherente con la función area que he construido anteriormente.

```
[22]: def condiciones(puntos,funcion):
    cond = []
    # Quitamos el último punto porque siempre será el
    # valor inicial de nuestra integral, por lo que
    # su condición será que el valor sea nulo
    for punto in puntos[:-1]:
        # Primero sustituimos el punto y hallamos los
        # valores de la preimagen
        fun = funcion.subs({y:punto,z:T})
        sols = [root[0] for root in R(fun).roots(AA)]
        n = len(sols)
        # Por último vamos restando dos a dos hasta
        # obtener el resultado
        valor = 0
        for r in range(n/2):
            valor += sols[2*r+1]-sols[2*r]
        cond += [(1,punto,valor)]
    cond += [(0,puntos[-1],0)]
    return cond
```

Como ya dijimos los anuladores están calculados en Mathematica con la función Annihilator del paquete RISC HolonomicFunctions. Como no podemos copiar y pegar directamente estos anuladores de Mathematica, la función clean se encarga de pasarlos a una forma que entienda SageMath.

```
[23]: def clean(text):
    text = text.replace(" y", "*y")
    text = text.replace(" ", "")
    text = text.replace("\n", "")
    text = text.replace("y", "t")
    for i in range(len(text)):
        if text[i] == "(":
            try:
                a = int(text[i-1])
                text = text[:i]+"*"+text[i:]
            except:
                continue
    return text
```

### Cálculo del volumen de un *Pretzel*

En este caso calcularemos el volumen dentro de un *pretzel*, dado por la ecuación

$$\{(x, y, z) \in \mathbb{R}^3 \mid (x^2(1 - x^2) - y^2)^2 + z^2 - 1/100 \leq 0\}.$$



```
[71]: t0_pret = time()
```

```
[72]: f = (x^2*(1 - x^2) - y^2)^2 + z^2 - 1/100
```

Al igual que antes, Marc Mezzarobba nos proporcionó el anulador calculado con Magma que anula a nuestra  $v$ .

```
[73]:
```

```

P_v = (t^24 - 6*t^22 + 61/4*t^20 - 85/4*t^18 + 514187/29400*t^16 -
↪ - 31081/3675*t^14 + 10271/4704*t^12 - 23453/117600*t^10 -
↪ 4563/245000*t^8 + 7787/2940000*t^6 + 19/2352000*t^4 - 19/
↪ 11760000*t^2)*D^3 + (-23/5*t^23 + 134/5*t^21 - 1293/20*t^19 +
↪ 414/5*t^17 - 8757701/147000*t^15 + 843439/36750*t^13 -
↪ 2134439/588000*t^11 - 11693/58800*t^9 + 212761/1837500*t^7 -
↪ 84001/7350000*t^5 - 437/19600000*t^3 + 19/14700000*t)*D^2 +
↪ (-21/5*t^22 + 86/5*t^20 - 631/20*t^18 + 38*t^16 - 1692939/
↪ 49000*t^14 + 1574899/73500*t^12 - 8241/1120*t^10 + 336649/
↪ 294000*t^8 - 87779/700000*t^6 + 3277/300000*t^4 + 19/
↪ 2352000*t^2 - 19/14700000)*D + 216/5*t^21 - 972/5*t^19 + 1836/
↪ 5*t^17 - 378*t^15 + 2811793/12250*t^13 - 403933/4900*t^11 +
↪ 39899/2450*t^9 - 1706/1225*t^7 + 209/19600*t^5 - 209/
↪ 196000*t^3
operador_v = P_v*D

```

Calculamos el mínimo y el máximo para  $x$ .

```

[74]: resy = f.resultant(diff(f,y),y)
      resz = f.resultant(diff(f,z),z)

```

```

[75]: %time s = ssolve([f,resy,resz],x,y,z,solution_dict=True)
      s

```

CPU times: user 625 ms, sys: 0 ns, total: 625 ms

Wall time: 625 ms

```

[75]: [...,
      {x: -sqrt(1/2 - sqrt(15)/10), y: 0, z: 0},
      ...,
      {x: sqrt(1/2 - sqrt(15)/10), y: 0, z: 0},
      ...,
      {x: -sqrt(1/2 - sqrt(35)/10), y: 0, z: 0},
      {x: sqrt(1/2 - sqrt(35)/10), y: 0, z: 0},
      ...,
      {x: -sqrt(sqrt(15)/10 + 1/2), y: 0, z: 0},
      ...,

```



```
{x: sqrt(sqrt(15)/10 + 1/2), y: 0, z: 0},
...]
```

Y nos quedamos solo con los puntos con coordenadas reales, en este caso vemos que tenemos tres pares de puntos, esto se debe a que el par de mayor módulo son el mínimo y máximo para la  $x$ , el segundo par de mayor módulo es el correspondiente al borde exterior de los agujeros del *pretzel*, y el último par es el borde interior de los agujeros.

```
[76]: xd = [-sqrt(1/2 - sqrt(15)/10), sqrt(1/2 - sqrt(15)/10), -sqrt(1/2 +
      ↪ sqrt(35)/10), sqrt(1/2 + sqrt(35)/10), -sqrt(sqrt(15)/10 + 1/
      ↪ 2), sqrt(sqrt(15)/10 + 1/2)]
      [x.n() for x in xd]
```

```
[76]: [-0.335710687019729,
      0.335710687019729,
      -1.04480044903798,
      1.04480044903798,
      -0.941965145119893,
      0.941965145119893]
```

Nos interesa conocer los tres pares, pero el volumen lo calcularemos entre el par de mayor módulo, así que les damos un nombre para que sea más cómodo usarlos.

```
[77]: x0 = xd[2]
      x1 = xd[3]
```

Una vez ya tenemos el anulador y los valores máximo y mínimo de  $x$ , ya solo nos queda calcular las condiciones iniciales suficientes, en este caso el operador  $_v$  es de orden 4 así que necesitaremos 3 más la condición de que en  $x_0$  valga 0.

### Cálculo de las condiciones iniciales

Calculemos primero los puntos singulares de nuestro operador, para ver que puntos no podemos tomar como  $x_{ord}$ . El coeficiente de mayor orden de operador  $_v$  es:

```
[78]:
```

```
g = t^24 - 6*t^22 + 61/4*t^20 - 85/4*t^18 + 514187/29400*t^16 -
↳ 31081/3675*t^14 + 10271/4704*t^12 - 23453/117600*t^10 - 4563/
↳ 245000*t^8 + 7787/2940000*t^6 + 19/2352000*t^4 - 19/
↳ 11760000*t^2
```

```
[79]: [root[0] for root in R(g.subs({t:T})).roots(AA)]
```

```
[79]: [-1.04480044903798?,
-1.012327152726?,
-1.00000000000000?,
-0.9868622089859?,
-0.94196514511990?,
-0.7071067811865475?,
-0.3357106870197288?,
-0.1615641682907841?,
0,
0.1615641682907841?,
0.3357106870197288?,
0.7071067811865475?,
0.94196514511990?,
0.9868622089859?,
1.00000000000000?,
1.012327152726?,
1.04480044903798?]
```

La primera condición la impondremos sobre  $x = 1/3$ . Como hemos hecho hasta ahora, todos los anuladores están calculados en Mathematica. Y procedemos de la misma manera que con los ejemplos anteriores. Como nuestra región es simétrica, la condición para  $x = -1/3$  coincidirá con la que calculemos ahora.

```
[80]: P = (-161-129600*t^2+656100*t^4)*D-16200*t*(-8+81*t^2)
operador = P*D
```

```
[81]: y0,y1 = extremos(f.subs({x:1/3}))
```

```
y0 = -0.4458311699497529?
y1 = 0.4458311699497529?
```

```
[82]: A1_3 = area(operador,condiciones([0,y0],f.subs({x:1/
↪3})),y1,precision)
```

El área de nuestra superficie es 0.

↪121930499785831521875572426418509317133891149

82417512311016741749054513029636938647987858967027065690.

Calculado en 0.6383311748504639 segundos.

La última condición la impondremos sobre  $x = 1/5$ .

```
[83]: P = (-13321-120000*t^2+1562500*t^4)*D+(120000*t-3125000*t^3)
operador = P*D
```

```
[84]: y0,y1 = extremos(f.subs({x:1/5}))
```

y0 = -0.3720215047547655?

y1 = 0.3720215047547655?

```
[85]: A1_5 = area(operador,condiciones([0,y0],f.subs({x:1/
↪5})),y1,precision)
```

El área de nuestra superficie es 0.

↪132882583958502236995021978391388571405290381

65282242442113863144535212336167906938329238605332164561.

Calculado en 0.31058740615844727 segundos.

### Cálculo del volumen

Por último solo debemos crear el vector cond y calcular el volumen con la función area definida anteriormente.

```
[86]: cond = [(1,1/3,A1_3),(1,-1/3,A1_3),(1,1/5,A1_5),(0,x0,0)]
```

```
[87]: pretzel = area(operador_v,cond,x1,precision)
```

Step 1/3 --> -1/3 passes through or too close to singular points

-0.1615641682907841?, 0, 0.1615641682907841? (to compute the

↪connection to a

singular point, make it a vertex of the path)

```

Inserte punto intermedio 0.16+0.05j
Step 1/3 --> -1/10*a^3 - a passes through or too close to
↳singular points
-1.012327152726?, -1, -0.986862208986?, -0.94196514511989?,
-0.7071067811865475?, -0.3357106870197288?, -0.1615641682907841?,
↳0,
0.1615641682907841? (to compute the connection to a singular
↳point, make it a
vertex of the path)
Inserte punto intermedio 0.16+0.05j
Step 1/3 --> 1/10*a^3 + a passes through or too close to singular
↳points
0.3357106870197288?, 0.7071067811865475?, 0.94196514511989?, 0.
↳986862208986?, 1,
1.012327152726? (to compute the connection to a singular point,
↳make it a vertex
of the path)
Inserte punto intermedio 0.33+0.05j
El área de nuestra superficie es 0.
↳197167031006011952005437977205406979668241106
71375940268322685211275869905250724231224199323252392090.
Calculado en 33.62344312667847 segundos.

```

```

[88]: t1_pret = time()
print(f'Área del Pretzel calculada en {t1_pret-t0_pret} segundos.
↳')

```

Área del Pretzel calculada en 36.77028274536133 segundos.

# Bibliografía

- [Ber71] I. N. Bernšteĭn, *Modules over a ring of differential operators. An investigation of the fundamental solutions of equations with constant coefficients*, Funkcional. Anal. i Priložen. **5** (1971), no. 2, 1–16.
- [CJ10] Francisco J. Castro Jiménez, *Modules over the Weyl algebra*, Algebraic approach to differential equations, World Sci. Publ., Hackensack, NJ, 2010, pp. 52–118.
- [Cou95] S. C. Coutinho, *A primer of algebraic D-modules*, London Mathematical Society Student Texts, vol. 33, Cambridge University Press, Cambridge, 1995.
- [Duo01] Javier Duoandikoetxea, *Fourier analysis*, Graduate Studies in Mathematics, vol. 29, American Mathematical Society, Providence, RI, 2001, Translated and revised from the 1995 Spanish original by David Cruz-Uribe.
- [Har77] Robin Hartshorne, *Algebraic geometry*, Springer-Verlag, New York-Heidelberg, 1977, Graduate Texts in Mathematics, No. 52.
- [KJJ13] Manuel Kauers, Maximilian Jaroschek, and Fredrik Johansson, *Ore polynomials in Sage*, [https://www3.risc.jku.at/research/combinat/software/ore\\_algebra/](https://www3.risc.jku.at/research/combinat/software/ore_algebra/), 2013.
- [Kou10] Christoph Koutschan, *HolonomicFunctions (user's guide)*, Johannes Kepler University, Linz, Austria, 2010, Technical Report 10-01, RISC Report Series.
- [LMD19] Pierre Lairez, Marc Mezzarobba, and Mohab Safey El Din, *Computing the volume of compact semi-algebraic sets*, prepublicación arXiv:1904.11705 [math.AG], 2019.
- [MB21] Isidro Matos Bellido, *Mi página de github*, <https://github.com/isimatosbe/FuncionesHolonomas>, 2021.

- [SKK73] Mikio Sato, Takahiro Kawai, and Masaki Kashiwara, *Microfunctions and pseudo-differential equations*, Hyperfunctions and pseudo-differential equations (Proc. Conf., Katata, 1971; dedicated to the memory of André Martineau), 1973, pp. 265–529. Lecture Notes in Math., Vol. 287.
- [SS03] Elias M. Stein and Rami Shakarchi, *Fourier analysis*, Princeton Lectures in Analysis, vol. 1, Princeton University Press, Princeton, NJ, 2003, An introduction.
- [SS19] Anna-Laura Sattelberger and Bernd Sturmfels, *D-Modules and Holonomic Functions*, prepublicación arXiv:1910.01395 [math.AG], pendiente de publicación en las actas de la *MATH+ Fall School on Algebraic Geometry*, 2019.
- [SST00] Mutsumi Saito, Bernd Sturmfels, and Nobuki Takayama, *Gröbner deformations of hypergeometric differential equations*, Algorithms and Computation in Mathematics, vol. 6, Springer-Verlag, Berlin, 2000.
- [Sta78] J. T. Stafford, *Module structure of Weyl algebras*, J. London Math. Soc. (2) **18** (1978).
- [Zei90] Doron Zeilberger, *A holonomic systems approach to special functions identities*, J. Comput. Appl. Math. **32** (1990), no. 3, 321–368.