数理解析研究所講究録 1456 巻 2005 年 117-125

Mora の割り算アルゴリズムと多項式の local b 関数の計算

中山洋将
NAKAYAMA HIROMASA
神戸大学大学院自然科学研究科
GRADUATE SCHOOL OF SCIENCE AND TECHNOLOGY, KOBE UNIVERSITY

1 Introduction

次のように記号を定めておく。

$$x=(x_1,\cdots,x_n),\quad \partial=(\partial_1,\cdots,\partial_n), s=(s)$$

$$D[s]=\{\sum_{k,\beta}a_{k,\beta}(x)s^k\partial^\beta \mid a_{k,\beta}(x)\in\mathbb{C}[x]\},\quad \mathbb{C}[x]_{\langle x\rangle}=\{\frac{f(x)}{g(x)}\mid f(x),g(x)\in\mathbb{C}[x],g(0)\neq 0\}$$

$$\mathcal{D}_{alg}[s]=\{\sum_{k,\beta}a_{k,\beta}(x)s^k\partial^\beta \mid a_{k,\beta}(x)\in\mathbb{C}[x]_{\langle x\rangle}\},\quad \widehat{\mathcal{D}}[s]=\{\sum_{k,\beta}a_{k,\beta}(x)s^k\partial^\beta \mid a_{k,\beta}(x)\in\mathbb{C}[[x]]\}$$

多項式 $f\in\mathbb{C}[x]$ について、f の global b 関数と、f の原点での local b 関数の定義と性質を簡単に説明する。f の global b 関数とは、$Pf^{s+1}=\tilde{b}(s)f^s$ となる微分作用素 $P\in D[s]$ が存在する、最小次数で monic な s の多項式 $\tilde{b}(s)\in\mathbb{C}[s]$ のことである。f の原点での local b 関数とは、$Pf^{s+1}=b(s)f^s$ となる微分作用素 $P\in\mathcal{D}_{alg}[s]$ が存在する、最小次数で monic な s の多項式 $b(s)\in\mathbb{C}[s]$ のことである。

簡単な例を挙げる。

例 1 ($f=x_1(x_1+x_2+1)$ の b 関数)
global b 関数は、$b(s)=(s+1)^2$ であって、実際次のような式が成り立つ。

$$(-\partial_2^2+\partial_1\partial_2)f^{s+1}=(s+1)^2f^s$$

local b 関数は、$b(s)=s+1$ であって、次の式が成り立つ。

$$\frac{1}{1+2x_1+x_2}\partial_1f^{s+1}=(s+1)f^s$$

このような多項式の b 関数を求めるアルゴリズムは今までに知られている。多項式の global, local b 関数を求めるアルゴリズムは、大阿久氏の [2], [8], [9] 等で与えられている。global b 関数を求める効率的な方法は、野呂氏の [7] で与えられている。また、 Risa/Asir 上で global b 関数を計算するライブラリ `bfct` があり、多項式だけ与えれば、`bfct` や `bfunction` という命令を用いて計算することができるようになっている。

新たに、D 上の Mora の割り算アルゴリズムが見つけられた。([5], [6]) これは、$\mathcal{D}_{alg}$ での割り算を与えている。local b 関数の計算は、$\mathcal{D}_{alg}[s]$ での計算であった。そこでその割り算アルゴリズムを用いて、今までとは別の local b 関数を計算するアルゴリズムを得た。またそれを Risa/Asir 上に実装し、様々な計算を行った。

2 global b 関数の計算アルゴリズム

global b 関数の計算は、次のアルゴリズムで計算できることが知られている。([2],[8])

アルゴリズム 1 (多項式 f の global b 関数の計算)

1. $\mathrm{Ann}_{D[s]}f^s$ の生成元 G の計算
2. J を $G \cup \{f\}$ の生成する $D[s]$ のイデアルとして、$J \cap \mathbb{C}[s]$ の生成元の計算 (この生成元が f の global b 関数である)

1. の $\mathrm{Ann}_{D[s]}f^s$ を計算する D_{n+1} におけるグレブナ基底を用いた方法が知られている。([2], [8], [9]) 2. の $J \cap \mathbb{C}[s]$ を計算するには、グレブナ基底計算を使って s 以外の変数を消去する方法と、J のグレブナ基底 G による s の巾乗の正規形を使った方法がある。([7])

3 local b 関数の計算アルゴリズムその 1

$D[y]$ における Mora の割り算 (定理 4) を紹介し、それを用いた local b 関数の計算アルゴリズム (アルゴリズム 2) を導く。

3.1 $D[y]$ 上の Mora の割り算アルゴリズム

今まで $D[s]$ で考えてきたが、あとの都合上、パラメータ変数を s から y にかえて、$D[y]$ を考える。またパラメータ変数 s はある斉次化のために用いるものとしておく。

以下の内容は、[5],[6] に書かれている微分作用素の Mora の割り算アルゴリズムの内容を少し変形したものである。(具体的には、斉次化微分作用素上の Mora の割り算アルゴリズムであったものを普通の微分作用素上の Mora の割り算アルゴリズムにした。また、パラメータ y がついている場合を考えた。)

$D[y]$ 上で次の単項式順序 $<_1$ についての割り算を行いたい。

x_1	$\cdots$	x_n	y	ξ_1	$\cdots$	ξ_n
0	$\cdots$	0	1	1	$\cdots$	1
-1	$\cdots$	-1	0	0	$\cdots$	0

(適当な項順序 $<'$)

$<_1$ は項順序でないので ($x_1,\cdots,x_n$ について局所順序である)、普通の割り算アルゴリズムでは一般に無限回つづいてしまう場合がある。そこで用いられるのが、Mora の割り算アルゴリズムである。このアルゴリズムでは、新たな変数 s を使って、元に $(-1,1)$-斉次化を行う。$<_1$ に付随して決まる $D[y][s]$ 上の項順序 $<_1^s$ を使い、割り算を行う。最後に結果を非斉次化して (s に 1 を代入して)、$<_1$ についての割り算の結果が得られるというものである。

定義 1 ($(-1,1)$-斉次化)
$P \in D[y]$ は、$P = \sum a_{\alpha\beta\gamma}x^\alpha\partial^\beta y^\gamma$ $(\alpha \in (\mathbb{Z}_{\geq 0})^n, \beta \in (\mathbb{Z}_{\geq 0})^n, \gamma \in \mathbb{Z}_{\geq 0})$ と表される。この時、

$$m = \min\{|\beta| - |\alpha| + |\gamma| \mid a_{\alpha\beta\gamma} \neq 0\} \quad (\text{ただし } |v| \text{ はベクトル } v \text{ の各成分の総和を表す})$$

とおいて、P の $(-1,1)$ -斉次化 $P^{(s)}$ を

$$P^{(s)} = \sum a_{\alpha\beta\gamma}x^\alpha\partial^\beta y^\gamma s^{-|\alpha|+|\beta|+|\gamma|-m}$$

と定義する。また、$P \in D[y][s]$ について、$P = \sum a_{\alpha\beta\gamma\nu} x^\alpha \partial^\beta y^\gamma s^\nu$ と表されているとする。$a_{\alpha\beta\gamma\nu} \neq 0$ であるすべての $(\alpha, \beta, \gamma, \nu)$ について、$d = -|\alpha| + |\beta| + |\gamma| - |\nu|$ が成り立つ時、P は d 次の $(-1,1)$-斉次であるという。

定義 2 ($<_1$ に付随する項順序)
$D[y][s]$ 上の項順序 $<_1^s$ を次のように定める。

x_1	$\cdots$	x_n	y	s	ξ_1	$\cdots$	ξ_n
1	$\cdots$	1	0	1	0	$\cdots$	0
0	$\cdots$	0	1	0	1	$\cdots$	1
-1	$\cdots$	-1	0	0	0	$\cdots$	0

(適当な項順序 $<'$)

この順序は、x と s の全次数比較の次に $<_1$ の比較を行うものである。最初に x と s の全次数比較が来るので、うまく項順序になっている。

補題 3 ($<_1$ と $<_1^s$ の LM)
$P \in D[y][s]$ として、P は $(-1,1)$-斉次元とする。この時、

$$\mathrm{LM}_{<_1}(P|_{s=1}) = (\mathrm{LM}_{<_1^s}(P))|_{s=1}$$

が成り立つ。また、$P, Q \in D[y][s]$ について、P, Q ともに 同じ次数の $(-1,1)$-斉次元であるとする。この時、

$$\mathrm{LM}_{<_1^s}(P) <_1^s \mathrm{LM}_{<_1^s}(Q) \Leftrightarrow \mathrm{LM}_{<_1}(P|_{s=1}) <_1 \mathrm{LM}_{<_1}(Q|_{s=1})$$

が成り立つ。

$P, Q \in D[y][s]$ が $(-1,1)$-斉次元であり、P が Q で簡約できる場合 (即ち、$\mathrm{LM}_{<_1^s}(P)$ が $\mathrm{LM}_{<_1^s}(Q)$ で割り切れる場合) を考える。すると簡約結果 R も P と同じ次数の $(-1,1)$-斉次元であり、$\mathrm{LM}_{<_1^s}(R) <_1^s \mathrm{LM}_{<_1^s}(P)$ が成り立つが、先の補題から、$\mathrm{LM}_{<_1}(R|_{s=1}) <_1 \mathrm{LM}_{<_1}(P|_{s=1})$ が成り立つ。この事を利用して、$D[y]$ 上の Mora の割り算アルゴリズムが導かれる。

定理 4 (微分作用素の Mora の割り算アルゴリズム, [5] [6])
$P, P_1, \cdots, P_m \in D[y]$ が与えられている。この時、$a(x) \in \mathbb{C}[x], Q_1, \cdots, Q_m \in D[y], R \in D[y]$ が存在して、次のことが成り立つ。

- $a(x)P = Q_1 P_1 + \cdots + Q_m P_m + R$
- $a(0) \neq 0$、 すなわち $a(x)$ は単元
- $Q_i \neq 0$ ならば、$\mathrm{LM}_{<_1}(Q_i P_i) \leq_1 \mathrm{LM}_{<_1}(P)$
- $R \neq 0$ ならば、$\mathrm{LM}_{<_1}(R)$ は $\mathrm{LM}_{<_1}(P_i)$ のいづれでも割り切れない

また、この $a(x), Q_1, \cdots, Q_m, R$ を計算するアルゴリズムが存在し、それを *Mora* の割り算アルゴリズムと呼ぶ。(具体的なアルゴリズムについては、*[5], [6]* に与えられている。)

$D[y]$ の元で生成される $\mathcal{D}_{alg}[y], \widehat{D}[y]$ 上のイデアル $\mathcal{I}$ についてグレブナ基底を計算するのに、Mora の割り算アルゴリズムを使うことができる。すなわち、$D[y]$ における Mora の割り算アルゴリズムを用いた Buchberger アルゴリズムを適用することでグレブナ基底を求めることができる。また、$D[y]$ の元が $\mathcal{D}_{alg}[y], \widehat{D}[y]$ のあるイデアルに属するかどうかの判定を行うことができる。

3.2 local b 関数の計算その1

$\mathcal{I}$ を、$\mathrm{Ann}_{D[s]}f^s$ と f の生成する $\mathcal{D}_{alg}[s]$ のイデアルとする。$\mathcal{I}\cap\mathbb{C}[s]$ の生成元が local b 関数になるのだが、この生成元の求めるには次のようにする。$\mathcal{I}$ の $\mathcal{D}_{alg}[s]$ のグレブナ基底を G とする。この G をもちいた $\mathcal{D}_{alg}[s]$ での割り算で、余りが 0 になるかどうかで $\mathcal{I}$ に元が入るかどうかを判定することができる。local b 関数 $b(s)$ は global b 関数 $\tilde{b}(s)$ の約元なので、$\tilde{b}(s)$ の各約元について $\mathcal{I}$ に入るかどうか判定して、そのうちで最小次数の元が local b 関数 $b(s)$ になる。

ここで問題となってくるのは、global b 関数の次数が大きくなるほど約元の数も莫大になるということである。しかし、全ての約元に対してイデアルに入るかどうかの判定をしなくてもよい。

$$g(s)\in\mathcal{I}\cap\mathbb{C}[s]\Leftrightarrow\exists(g(s)\text{ の約元})\in\mathcal{I}\cap\mathbb{C}[s]$$

だから、$g(s)\notin\mathcal{I}$ なることが確定したら、$g(s)$ の約元については調べる必要がないことがわかる。

global b 関数が

$$\tilde{b}(s)=b_1(s)^{e_1}b_2(s)^{e_2}\cdots b_l(s)^{e_l}\quad(b_i(s)=s+a_i\quad a_i\in\mathbb{Q}_{>0})$$

と表されるとせよ。

$$f_i=\tilde{b}(s)/b_i(s)$$

とおく。$\mathcal{J}$ のグレブナ基底 G と Mora の割り算の剰余を用いて、つぎのようにイデアル所属判定ができたとする。

$$f_{i_1}(s),\cdots,f_{i_m}(s)\in\mathcal{I}\quad f_{j_1},\cdots,f_{j_{l-m}}\notin\mathcal{I}$$

もし、f_i の内で、$\mathcal{I}$ に入るものがなければ、この時点で $\mathcal{I}\cap\mathbb{C}[s]$ の生成元は $\tilde{b}(s)$ と確定する。それ以外の場合、すなわち $m>0$ の場合を考える。ここで、$g(s)=\gcd(f_{i_1}(s),\cdots,f_{i_m}(s))$ とせよ。$g(s)\in\mathcal{I}$ である。上の $\tilde{b}(s)$ の部分を $g(s)$ に置き換えて、同じ操作を終わるまで繰り返せばよい。これをまとめれば、次のようになる。

アルゴリズム 2 (local-b 関数の計算 (総当たり))

$\boxed{input}$ $f\in\mathbb{C}[x]$
$\boxed{output}$ f の local b 関数
$H\leftarrow\mathrm{Ann}_{D[s]}f^s$ の生成系
$\mathcal{I}\leftarrow(H\cup\{f\}$ の生成する $\mathcal{D}_{alg}[s]$ のイデアル $)$
$G\leftarrow\mathcal{I}$ のグレブナ基底
$BF\leftarrow f$ の global b 関数
$LF\leftarrow BF$
while (true) {
　　$LF=b_1(s)^{e_1}\cdots b_l(s)^{e_l}\quad(b_i(s)=s+a_i,a_i\in\mathbb{Q}_{>0})$ と表されたとする。(LF の因数分解)
　　$f_i\leftarrow LF/b_i(s)\quad(1\leq i\leq l,$ ただし $b_i(s)|LF$ が成り立つ i に限る$)$
　　f_i が $\mathcal{I}$ に入るかどうかの判定を行う。
　　(f_i を G で Mora の割り算アルゴリズムにより割って、余りが 0 かどうかで判定する。)
　　if $(f_i\notin\mathcal{I}(1\leq\forall i\leq l))$
　　　　return LF
　　$f_{i_1},\cdots,f_{i_m}\in\mathcal{I}$ であったとする。
　　$LF\leftarrow\gcd(f_{i_1},\cdots,f_{i_m})$
}

例 2 ($f=(x-1)^3+(y+1)^2$ の local b 関数)
$f=(x-1)^3+(y+1)^2$ の *global* b 関数は $\tilde{b}(s)=(s+1)(s+\frac{5}{6})(s+\frac{7}{5})$ である。また、この場合 f は原点で非特異なので、計算するまでもなく *local* b 関数は $s+1$ とわかるのであるが、上の方法で計算してみる。$\mathrm{Ann}_{D[s]}f^s$ の生成系は $\{g_1=-6s-2\partial_x+3\partial_y+2x\partial_x+3y\partial_y, g_2=2\partial_x-3\partial_y+2y\partial_x+6x\partial_y-3x^2\partial_y\}$ となる。$\mathcal{I}$ を $\mathcal{D}_{alg}[s]\cdot\{f,g_1,g_2\}$ として、$\mathcal{I}$ のグレブナ基底 G は $\{f,g_1,g_2\}$ となる。

$f\to g$ と書くと、f を G で Mora の割り算アルゴリズムを用いて割った余りが g であることを表すとする。次は自明であるが、計算してみると確かに

$$\tilde{b}(s)=(s+1)(s+\frac{5}{6})(s+\frac{7}{5})\to 0$$

$\tilde{b}(s)$ の因子の指数を一つ下げた約元について

$$f_1=(s+1)(s+\frac{5}{6})\to 0,\quad f_2=(s+1)(s+\frac{7}{5})\to 0,\quad f_3=(s+\frac{5}{6})(s+\frac{7}{5})\to \textit{non-zero}$$

$(s+1)=\gcd(f_1,f_2)$ であり、$s+1\to 0$ となって、結局 *local* b 関数 $b(s)=s+1$ となることがわかる。

4 local b 関数の計算アルゴリズムその2

$\widehat{\mathcal{D}}[s]$ における割り算定理 (定理 6) とその近似割り算アルゴリズム (アルゴリズム 4) を紹介し、それを用いた local b 関数計算アルゴリズムを導く。

この Section で使う記号の説明をしておく。$\mathrm{Rest}_<(f)$ は f の先頭項を除いた部分を表す。$\mathrm{LE}_<(f)$ は f の先頭項の指数ベクトルを表す。$\mathrm{Exps}(f)$ は f の各項の指数ベクトルを全て集めた集合を表す。$\mathrm{Mono}(A)$ ($A=\{\alpha_1,\cdots,\alpha_n \mid \alpha_i\in(\mathbb{Z}_{\geq 0})^n\}$) は指数ベクトルの集合 A のモノイデアルを表す。すなわち、$\mathrm{Mono}(A)=\cup_{i=1}^n(\alpha_i+(\mathbb{Z}_{\geq 0})^n)$ である。$\mathrm{in}_e(P)$ (微分作用素 P, 重みベクトル e) は、微分作用素 P の重みベクトル e による主部を表す。$\mathrm{ord}_e(P)$ は重みベクトル e についての P の階数 (すなわち、最大重み) を表す。

4.1 $\mathbb{C}[[x]]$ の割り算定理とその近似アルゴリズム

$\widehat{\mathcal{D}}$ の近似割り算アルゴリズムを与えるために、その準備として $\mathbb{C}[[x]]$ の割り算である Weierstrass-Hironaka の割り算 (WH 割り算) について述べる。またその近似である $\mathbb{C}[[x]]$ の近似割り算アルゴリズムを導く。

$\{x^\alpha\}$ $(\alpha\in(\mathbb{Z}_{\geq 0})^n)$ 上の全順序 $<_r$ を次の行列で定義されるものとする。

x_1	$\cdots$	x_n
-1	$\cdots$	-1
(	適当な項順序 $<$	)

すなわち、$<_r$ は $\mathbb{C}[x]$ 上の逆次数順序である。$<_r$ は項順序でないため、普通の多項式の割り算アルゴリズムでは割り算が有限回で終わるとは限らないことに注意する。この $<_r$ における割り算について、無限回の操作を許すと次のような定理が成り立つ。

定理 5 (**Weierstrass-Hironaka の割り算定理, [1], [3]**)
$f\in\mathbb{C}[[x]], f\neq 0$ と $G=\{g_1,\cdots,g_s\}\subset\mathbb{C}[[x]]$ に対して、ある $q_1,\cdots,q_s,r\in\mathbb{C}[[x]]$ が存在して次の条件を満たす。

- $f=q_1g_1+\cdots+q_sg_s+r$

- $q_i \neq 0$ ならば $\mathrm{LM}_{<_r}(q_i g_i) \leq_r \mathrm{LM}_{<_r}(f)$
- $\mathrm{Exps}(r) \cap \mathrm{Mono}(\{\mathrm{LE}_{<_r}(g_1), \cdots, \mathrm{LE}_{<_r}(g_s)\}) = \phi$

しかし、この計算は一般に無限回の計算を必要とする。そのため、ある次数で計算打ち切ることにより有限回の計算でおさめる。ある次数までは正確な商と余りが得られるような近似割り算が得られる。

アルゴリズム 3 (WH 割り算の近似)

$\boxed{\textit{Input}}$ $f \in \mathbb{C}[x], G = \{g_1, \cdots, g_s\} \subset \mathbb{C}[x], N \in \mathbb{Z}_{>0}$

$\boxed{\textit{Output}}$ $\overline{Q'_1}, \cdots, \overline{Q'_s}, \overline{R'}, \overline{F} \in \mathbb{C}[x]$ で次を満たす。

$f = \overline{Q'_1} g_1 + \cdots + \overline{Q'_s} g_s + \overline{R'} + \overline{F}$

$\overline{Q'_i} \neq 0$ ならば $\mathrm{LM}_{<_r}(\overline{Q'_i} g_i) \leq_r \mathrm{LM}_{<_r}(f)$

$\mathrm{Exps}(\overline{R'}) \cap \mathrm{Mono}(\{\mathrm{LE}_{<_r}(g_1), \cdots, \mathrm{LE}_{<_r}(g_s)\}) = \phi$

$\overline{Q'_i}$ の全次数が $N - |\mathrm{LE}_{<_r}(g_i)| - 1$ 次の項 までは正確

$\overline{R'}$ の全次数が $N-1$ 次の項までは正確

$\boxed{\textit{WH-approximate-division}(f, G, N)}$

$\overline{F} \leftarrow f, \overline{Q'_i} \leftarrow 0, \overline{R'} \leftarrow 0, \beta \leftarrow 0$

while ($\overline{F} \neq 0$ *and* $|\beta| < N$) {

  $[Q, R] \leftarrow$ *mono-div*$(\overline{F}, G)$ ($\overline{F}$ を $\mathrm{LT}_{<_r}(G)$ で完全簡約する操作で、その商を Q、余りを R に)

  $\beta \leftarrow \max_{<_r}(\{\mathrm{LE}_{<_r}(Q_1 g_1), \cdots, \mathrm{LE}_{<_r}(Q_s g_s), \mathrm{LE}_{<_r}(R)\})$

  $\overline{Q'_i} \leftarrow \overline{Q'_i} + Q_i$

  $\overline{R'} \leftarrow \overline{R'} + R$

  $\overline{F} \leftarrow -\sum_{i=1}^{s} Q_i \mathrm{Rest}_{<_r}(g_i)$

}

return $[\overline{Q'}, \overline{R'}, \overline{F}]$

この近似割り算アルゴリズムは、$\widehat{\mathcal{D}}$ における近似割り算アルゴリズム (アルゴリズム 4) で用いられる。

4.2 $\widehat{\mathcal{D}}[y]$ の割り算定理とその近似アルゴリズム

Castro は [4] において、$\widehat{\mathcal{D}}$ の割り算定理を与えた。この節では、この割り算定理により求められる商と余りを与えられた次数まで正確に求める近似割り算アルゴリズムを導く。$y = (y)$ を 1 変数のパラメータとしておく。

$<_1$ を section 3.1 で用いられている単項式順序とする。$\{x^\alpha \xi^\beta y^\gamma\}$ 上に $<_r$ を改めて定義する。$<_r$ は次の行列で表される。

x_1	$\cdots$	x_n	y	ξ_1	$\cdots$	ξ_n
-1	$\cdots$	-1	-1	-1	$\cdots$	-1
(		$<_1$ で用いられている項順序 $<'$				)

また、重みベクトル e を次のように定義する。

x_1	$\cdots$	x_n	y	ξ_1	$\cdots$	ξ_n
0	$\cdots$	0	1	1	$\cdots$	1

ここで $<_r$ について次の事が成り立つことに注意する。$|\beta|+|\gamma| = |\beta'|+|\gamma'|$ の時 (すなわち e についての階数が等しい時), $x^\alpha \xi^\beta y^\gamma <_1 x^{\alpha'} \xi^{\beta'} y^{\gamma'} \Leftrightarrow x^\alpha \xi^\beta y^\gamma <_r x^{\alpha'} \xi^{\beta'} y^{\gamma'}$ である。だから、$\mathrm{LM}_{<_1}(P) = \mathrm{LM}_{<_r}(\mathrm{in}_e(P))$ が成り立つ。

$\widehat{\mathcal{D}}[y]$ での割り算定理として次のものが知られている。$<_1$ が項順序でないので、一般に無限回の操作が必要になる。

定理 6 ($\widehat{\mathcal{D}}[y]$ における割り算定理 ,[4],[1])
$P, P_1, \cdots, P_s \in \widehat{\mathcal{D}}[y]$ に対して、次の条件を満すような $Q_1, \cdots, Q_s, R \in \widehat{\mathcal{D}}[y]$ が存在する。

- $P = Q_1P_1 + \cdots + Q_sP_s + R$
- $\mathrm{Exps}(R) \cap \mathrm{Mono}(\{\mathrm{LE}_{<_1}(P_1), \cdots, \mathrm{LE}_{<_1}(P_s)\}) = \phi$
- $Q_k \neq 0$ ならば、$\mathrm{LM}_{<_1}(Q_kP_k) \leq_1 \mathrm{LM}_{<_1}(P)$

そこで、割る元、割られる元を $D[y]$ の元として、商、余りが与えられた次数まで正確に求めることのできる近似割り算アルゴリズムを考える。WH 割り算の近似アルゴリズム (アルゴリズム 3) を用いて、これを実現する。

アルゴリズム 4 ($\widehat{\mathcal{D}}[y]$ の割り算の近似)
$\boxed{input}$ $P, P_1, \cdots, P_s \in D[y], N \in \mathbb{Z}_{>0}$
$\boxed{output}$ $\overline{Q_1}, \cdots, \overline{Q_s}, \overline{R} \in D[y]$ s.t.
　$P = \overline{Q_1}P_1 + \cdots + \overline{Q_s}P_s + \overline{R}$
　$\overline{Q_i} \neq 0$ ならば $\mathrm{LM}_{<_1}(\overline{Q_i}P_i) \leq_1 \mathrm{LM}_{<_1}(P)$
　$\{\alpha \mid \alpha \in \mathrm{Exps}(\overline{R}), |\alpha| < N\} \cap \mathrm{Mono}(\{\mathrm{LE}_{<_1}(P_1), \cdots, \mathrm{LE}_{<_1}(P_s)\}) = \phi$
　$\overline{Q_i}$ の全次数が $N - |\mathrm{LE}_{<_1}(P_i)|$ 次未満の部分は正確である。
　$\overline{R}$ の全次数が N 次未満の部分は正確である。
$\boxed{\textit{D-approximate-division}(P, \{P_1, \cdots, P_s\}, N)}$
$\overline{Q_1} \leftarrow 0, \cdots, \overline{Q_s} \leftarrow 0, \overline{R} \leftarrow P$
$m_0 \leftarrow \mathrm{ord}_e(\overline{R})$
for $(k \leftarrow 0; k \leq m_0; k \leftarrow k+1)$
　$M_k \leftarrow \max(\{|\mathrm{LE}_{<_1}(P_i)| + 2(\max(k - \mathrm{ord}_e(P_i), 0) \mid 1 \leq i \leq s\})$
Bound $\leftarrow N + \sum_{i=0}^{m_0} M_i$
for $(k \leftarrow m_0; k \geq 0; k \leftarrow k-1)$ {
　$\overline{r} \leftarrow$ ($\overline{R}$ のうちの e 階数 k で全次数が *Bound* 未満の部分の全表象)
　$[\overline{q_i'}, \overline{r'}] \leftarrow$ *WH-approximate-division*$(\overline{r}, \{\mathrm{in}_e(P_1), \cdots, \mathrm{in}_e(P_s)\}, <_r,$ *Bound*$)$
　$\overline{Q_i'} \leftarrow$ ($\overline{q_i'}$ の ξ を ∂ にかえたもの)
　$\overline{R} \leftarrow \overline{R} - \sum \overline{Q_i'}P_i$
　$\overline{Q_i} \leftarrow \overline{Q_i} + \overline{Q_i'}$
　Bound $\leftarrow$ *Bound* $- M_k$
}
return $[\overline{Q}, \overline{R}]$

4.3 $\mathcal{I} \cap \mathbb{C}[s]$ の計算アルゴリズム

$\widehat{\mathcal{D}}[s]$ のイデアル $\mathcal{I}$ について、$\mathcal{I} \cap \mathbb{C}[s]$ の生成元を計算する方法を考える。P を グレブナ基底 G で割り算 (定理 6) を行った余りを正規形と呼び、$\mathrm{NF}(P, G, <_1)$ で表す。

$g(s)=a_l s^l+a_{l-1}s^{l-1}+\cdots+a_0\in\mathcal{I}$ について、次のことが成り立つ。G を $\mathcal{I}$ の $<_1$ についてのグレブナ基底としておく。$g(s)\in\mathcal{I}$ は $\mathrm{NF}(g(s),G,<_1)=0$ と同値。

$$\mathrm{NF}(g(s),G,<_1)=a_l\mathrm{NF}(s^l,G,<_1)+a_{l-1}\mathrm{NF}(s^{l-1},G,<_1)+\cdots+a_0\mathrm{NF}(1,G,<_1)=0$$

である。だから、あらかじめ $\mathrm{NF}(s^i,G,<_1)$ を計算しておき、$\mathrm{NF}(g(s),G,<_1)=0$ が成り立つような最小の l と $a_l,\cdots,a_0\in\mathbb{C}$ を計算してやれば、$\mathcal{I}\cap\mathbb{C}[s]$ の生成元 $a_l s^l+\cdots+a_0$ が得られるわけである。

しかし問題となるのは、今 $\widehat{\mathcal{D}}[s]$ で考えているので、正規形 $\mathrm{NF}(s^i,G,<_1)$ は一般には無限個の項を持ち、アルゴリズミックに求められない。そこで正規形の近似を導入する。

定義 7 (正規形の近似)
$P\in\widehat{\mathcal{D}}[s]$ と $\widehat{\mathcal{D}}[s]$ のあるイデアル $\mathcal{I}$ のグレブナ基底 G について、(アルゴリズム 4) の近似割り算アルゴリズムで P を G で全次数 $N-1$ 次まで割ることにより得られた余り $\overline{R}$ を、P の G による $<_1$ についての $N-1$ 次までの正規形の近似と呼び、$\mathrm{NF}(P,G,<_1,N)$ で表す。$\mathrm{NF}(P,G,<_1)$ と $\mathrm{NF}(P,G,<_1,N)$ は、全次数が $N-1$ 次まで一致する。

正規形の近似を用いた $\mathcal{I}\cap\mathbb{C}[s]$ の生成元の計算法について説明する。ここで問題となるのは、

$$a_l\mathrm{NF}(s^l,G,<_1,N)+a_{l-1}\mathrm{NF}(s^{l-1},G,<_1,N)+\cdots+a_0\mathrm{NF}(1,G,<_1,N)=0$$

が成り立っても、近似であるから、即 $a_l s^l+a_{l-1}s^{l-1}+\cdots+a_0\in\mathcal{I}$ とは言えないことである。$D[s]$ のある元 P が $\mathcal{I}$ に入るかどうかを判定するのに、Mora の割り算アルゴリズム (定理 4) を用いればよい。余りが 0 なら $P\in\mathcal{I}$ であり、そうでなければ $P\notin\mathcal{I}$ である。

$\mathcal{I}\cap\mathbb{C}[s]$ の生成元を計算するアルゴリズムについて述べる。準備として、次のような線形空間を考える。d を十分大きな自然数 ($\mathcal{I}\cap\mathbb{C}[s]$ の生成元の次数以上の数) とし、N を適当な自然数としておく。

$$L_{i,N}=\{(a_0,\cdots,a_i,0,\cdots,0)\in\mathbb{C}^{d+1}\mid a_i\mathrm{NF}_<(s^i,G,<_1,N)+\cdots+a_0\mathrm{NF}_<(1,G,<_1,N)=0\}\quad(1\le i\le d)$$

$$L_i=\{(a_0,\cdots,a_i,0,\cdots,0)\in\mathbb{C}^{d+1}\mid a_i\mathrm{NF}_<(s^i,G,<_1)+\cdots+a_0\mathrm{NF}_<(1,G,<_1)=0\}\quad(1\le i\le d)$$

この集合について、次のような包含関係があることに注意する。

$$\begin{array}{ccccccccc}
L_{d,N} & \supset & L_{d,N+1} & \supset & L_{d,N+2} & \supset & \cdots & \supset & L_d \\
\cup & & \cup & & \cup & & & & \cup \\
L_{d-1,N} & \supset & L_{d-1,N+1} & \supset & L_{d-1,N+2} & \supset & \cdots & \supset & L_{d-1} \\
\cup & & \cup & & \cup & & & & \cup \\
\vdots & & \vdots & & \vdots & & & & \vdots
\end{array}$$

$\mathcal{I}\cap\mathbb{C}[s]$ の生成元は、$L_i\ne\{0\},L_{i-1}=\{0\}$ なる i について、L_i の元 $(a_0,\cdots,a_i,0,\cdots,0)$ からつくられる $a_i s^i+\cdots+a_0$ である。このような元を得るために、次のような手順を行う。

アルゴリズム 5 ($\mathcal{I}\cap\mathbb{C}[s]$ の生成元の計算)

1. $L_{d,N},L_{d-1,N},\cdots$ と見ていき、$L_{i,N}\ne\{0\},L_{i-1,N}=\{0\}$ なる i を探す。(上の包含関係から、$L_{i-1}=L_{i-2}=\cdots=\{0\}$ が確定する。)

2. $(a_0,\cdots,a_i,0,\cdots,0)\in L_{i,N}$ をとってきて、$g(s)=a_i s^i+\cdots+a_0$ なる多項式を作る。

3. $g(s)$ を G で $<_1$ について Mora の割り算を行い、余りを R とする。$R=0$ の場合、$g(s)\in\mathcal{I}$ となり、$(a_0,\cdots,a_i,0,\cdots,0)\in L_i$ で、$g(s)$ は確かに $\mathcal{I}\cap\mathbb{C}[s]$ の生成元になる。$R\ne0$ の場合、N を 1 増やして 1. に戻る。

4.4 local b 関数の計算その2

$\mathcal{I}$ を、$\mathrm{Ann}_{\mathcal{D}[s]}f^s$ と f の生成する $\mathcal{D}[s]$ のイデアルとする。$\mathcal{I} \cap \mathbb{C}[s]$ の生成元が local b 関数だったので、(アルゴリズム 5) を使って local b 関数を計算できる。実際に計算例を挙げる。

例 3 ($f = x^2(y+1)^2z^2$ の原点での local b 関数)
まず結果から述べると、*global* b 関数は $(s+1)^3(s+1/2)^3$ であり、*local* b 関数は $(s+1)^2(s+1/2)^2$ である。$\mathrm{Ann}_{\widehat{\mathcal{D}}[s]}f^s$ の生成元 H は、$\{P_1 = -2s + z\partial_z, P_2 = x\partial_x - z\partial_z, P_3 = -\partial_y - y\partial_y + z\partial_z\}$ である。$\mathcal{J} = \widehat{\mathcal{D}}[s]\cdot(H \cup \{f\})$ の $<_1$ についてのグレブナ基底 G は、$\{f, P_1, P_2, P_3, P_4 = -z^4\partial_z^2 - 2yz^4\partial_z^2 - y^2z^4\partial_z^2 - 4z^3\partial_z - 8yz^3\partial_z - 4y^2z^3\partial_z - 2z^2 - 4yz^2 - 2y^2z^2, P_5 = xz^3\partial_z + 2xyz^3\partial_z + xy^2z^3\partial_z + 2xz^2 + 4xyz^2 + 2xy^2z^2\}$ となる。$d = 6, N = 7$ として、$\mathrm{NF}(s^i, G, <_1, 7)$ $(i = 0, \cdots, 6)$ を計算すれば、

$$\mathrm{NF}(1, G, <_1, 7) = 1, \quad \mathrm{NF}(s, G, <_1, 7) = 1/2z\partial_z, \quad \mathrm{NF}(s^2, G, <_1, 7) = 1/4z^2\partial_z^2 + 1/4z\partial_z,$$

$$\mathrm{NF}(s^3, G, <_1, 7) = 1/8z^3\partial_z^3 + 3/8z^2\partial_z^2 + 1/8z\partial_z, \quad \mathrm{NF}(s^4, G, <_1, 7) = -3/8z^3\partial_z^3 - 31/16z^2\partial_z^2 - 31/16z\partial_z - 1/4,$$

$$\mathrm{NF}(s^5, G, <_1, 7) = 23/32z^3\partial_z^3 + 135/32z^2\partial_z^2 + 157/32z\partial_z + 3/4,$$

$$\mathrm{NF}(s^6, G, <_1, 7) = -9/8z^3\partial_z^3 - 447/64z^2\partial_z^2 - 555/64z\partial_z - 23/16$$

となる。ここで上で得た正規形の近似から、$L_{6,7}, L_{5,7}, \cdots$ と順に計算していくと、$L_{4,7} \neq \{0\}, L_{3,7} = \{0\}$ が出てくる。ここまでで、$L_3 = L_2 = \cdots = L_0 = \{0\}$ は確定した。すなわち、*local* b 関数の次数は 3 より大きいことがわかる。$(1/4, 3/2, 13/4, 3, 1, 0, 0) \in L_{4,7}$ であることが計算からわかるから、$g(s) = s^4 + 3s^3 + 13/4s^2 + 3/2s + 1/4$ が *local* b 関数の候補である。この $g(s)$ を G で $<_1$ について *Mora* の割り算を行えば、余りが 0 になることがわかる。よって、$g(s) \in \mathcal{J}$ となり、$g(s)$ は $\mathcal{J}$ に含まれる s について最小次数の元であることが保証される。$g(s)$ は *local* b 関数である。

参 考 文 献

[1] 大阿久俊則:グレブナ基底と線型偏微分方程式系 (計算代数解析入門), 上智大学数学講究録 ,No. 38, (1994)

[2] 大阿久俊則:計算代数と D 加群, 朝倉書店, (2002)

[3] 広中平祐, ト部東介:解析空間入門, 朝倉書店, (1981)

[4] Castro,F.: Calculs effectifs pour les idéaux d'opérateurs différentiels, Travaux en Cours 24, (1987)

[5] Granger,M.,Oaku,T.: Mimimal filtered free resolutions and division algorithms for analytic D-modules ,(2003)

[6] Granger,M.,Oaku,T.,Takayama,N.:Tangent cone algorithm for homogenized differential operators ,(2003)

[7] Noro,M.: An Efficient Modular Algorithm for Computing the Global b- function , Mathematical Software, (2002), 147 - 157

[8] Oaku,T.: An algorithm of computing b-function, Duke Math. J., 87, (1997), no.1 115 - 132

[9] Oaku,T.: Algorithm for the b-function and D-modules asscociated with a polynomial, J.Pure.Appl.Algebra 117/118, (1997), 495 - 518