数理解析研究所講究録
第 646 巻 1988 年 19-36

U-終結式の計算について

三菱総研　藤瀬 哲朗
(Tetsuro Fujise)
東大・大型計算機センター　村尾 裕一
(Hirokazu Murao)

1. はじめに

U-終結式を用いた連立代数方程式の有限解の記号的構成法について報告する。U-終結式を利用することは、解の元の方程式の性質を落とさずに構成できる有利な面をもつが、その構成は Van der Waerden が彼の現代代数学の注釈に「これに関連した実際的な計算は非常に複雑すぎて現実に実行することはとても望めない」と挙げた計算でもある。ここでは、連立代数方程式の多重度を含めた有限解の全解を、ある制約式を導き、それにより定められる媒介変数を含む記号式の形で求める。特に、有理式表現と多項式表現の二通りの解表示形式を構成した。

2. 問題

$\mathbb{Q}$を有理数体とし、$\mathbb{Q}[x_1, \cdots, x_n]$ をn変数の$\mathbb{Q}$-係数多項

式環とする。

$$f_1,\ f_2,\ \cdots,\ f_k \in \mathbb{Q}[x_1, \cdots, x_n]$$

を与えたとき，連立代数方程式

$$f_1 = 0,\ f_2 = 0,\ \cdots,\ f_n = 0 \qquad \cdots\cdots \quad (1)$$

を解く。(1)が有限個の有限解をもつ場合，次のDをここではU-終結式と呼ぶ(注)：

$$D(U_0, U_1, \cdots, U_n) = \prod_{j=1}^{N} (a_{0j} U_0 + a_{1j} U_1 + \cdots + a_{nj} U_n)$$

但し，

N ： (1)の多重度を含んだ有限解の個数，

$\left(\dfrac{a_{1j}}{a_{0j}}, \dfrac{a_{2j}}{a_{0j}}, \cdots, \dfrac{a_{nj}}{a_{0j}}\right)$ ：(1)の有限解，$a_{0j} \neq 0$

とする。

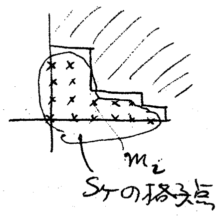
3。U-終結式及び数値解の構成法

[1] によると，U-終結式は，以下の算法で計算される。

算法 1　（記法等詳しくは [1] を参照）

① イデアル $(f_1, f_2, \cdots, f_k)$ の Gröbner basis G を全次数順序で求める。

② $\{m_1, m_2, \cdots, m_s\}$ を G で規約な単項式の集合とする。
($\mathbb{Q}[x_1, \cdots, x_n]/(f_1, \cdots, f_k)$ の基底)



(注)
本来の U-終結式は無限遠解 $(a_{0j} = 0)$ も含む。

$$l = U_0 + U_1 x + \cdots + U_n x_n \quad , \quad l_i = l \cdot m_i$$

とおき，l_i を G で簡約化した結果を $\tilde{l_i}$ とし，基底 $m_1, \cdots, m_s$ を用いて

$$\tilde{l_i} = \sum_{j=1}^{s} c_{ji} m_j$$

と表わす。この係数 c_{ji} より行列をつくり

$$\Lambda = (c_{ji})$$

と書く（ここで $c_{ji} \in \mathbb{Q}[U_0, U_1, \cdots, U_n]$ で全次数は高々1.)。

③ $D(U_0, U_1, \cdots, U_n) = \det \Lambda$ を計算する。この D が U-終結式（我々の意味で）である。

さらに，[1]によれば，解は以下のようにして求められる（但し数値解）。… 骨子のみを記す。

__算法2__　（D の1次因子（～平面）への分解）

① D と適当な平面 $(U_0, U_1, \cdots, U_n) = (1, b_1 t, \cdots, b_n t)$ との交点を求める。D を媒介変数 (t) 表示する：

$$D(U_0, U_1, \cdots, U_n) = G(t)。$$

② 一変数代数方程式 $G(t) = 0$ を数値的に解き，$t_1, \cdots, t_s$ を得る。

③ $a_i(t) = \frac{\partial}{\partial U_i} D$ とし，解

$$\left(\frac{a_1(t_j)}{a_0(t_j)}, \frac{a_2(t_j)}{a_0(t_j)}, \cdots, \frac{a_n(t_j)}{a_0(t_j)} \right)$$

を得る $(j = 1, \cdots, s)$。

もし、解が m 乗根である場合には、

$$a_0^{(m)}(t) = \frac{\partial^m D}{\partial U_0^m}, \quad a_i^{(m)}(t) = \frac{\partial^m D}{\partial U_0^{m-1}\partial U_i}$$

$(i=1, \cdots, n)$

とし、同様に解

$$\left(\frac{a_1^{(m)}(t_j)}{a_0^{(m)}(t_j)}, \frac{a_2^{(m)}(t_j)}{a_0^{(m)}(t_j)}, \cdots, \frac{a_n^{(m)}(t_j)}{a_0^{(m)}(t_j)} \right)$$

を得る。

4. 記号解の構成方針

前節の算法2で $G(t)$ を数値的に解かずに、全解をパラメータ t を含んだ形で構成する。実際の計算の効率化は次節で説明するとして、記号解の構成方針を述べる。

方針

① U-終結式 $D(U_0, U_1, \cdots, U_n)$ を媒介変数表示し、

$$G(t) = D(U_0=1, U_1=b_1 t, \cdots, U_n=b_n t)$$

とおく（算法2の①と同じ）。

② $G(t)$ を $\mathbb{Q}$ 上で因数分解する：

$$G(t) = G_1(t)^{e_1} G_2(t)^{e_2} \cdots G_r(t)^{e_r}.$$

③ $G_i(t)=0$ を制約式として（$\mathbb{Q}$ 上代数拡大）解を構成する。

i) 単根の場合 ($e_i = 1$).

i-i) D の各変数での偏導関数を媒介変数で表す:

$$H_j(t) = \frac{\partial D}{\partial U_j} \quad (U_0 = 1, U_1 = b_1 t, \cdots, U_n = b_n t).$$

i-ii) $G_i(t) = 0$ の下で，有理式表示解

$$\left(\frac{H_1(t)}{H_0(t)}, \frac{H_2(t)}{H_0(t)}, \cdots, \frac{H_n(t)}{H_0(t)} \right)$$

を得る。

ii) 重根の場合 ($e_i > 1$).

ii-i) 次の偏導関数を媒介変数で表す:

$$H_0^{(e_i)}(t) = \frac{\partial^{e_i} D}{\partial U_0^{e_i}} \quad (U_0 = 1, U_1 = b_1 t, \cdots, U_n = b_n t),$$

$$H_j^{(e_i)}(t) = \frac{\partial^{e_i} D}{\partial U_0^{e_i - 1} \partial U_j} \quad (U_0 = 1, U_1 = b_1 t, \cdots, U_n = b_n t).$$

ii-ii) $G_i(t) = 0$ の下で，有理式表示解

$$\left(\frac{H_1^{(e_i)}(t)}{H_0^{(e_i)}(t)}, \frac{H_2^{(e_i)}(t)}{H_0^{(e_i)}(t)}, \cdots, \frac{H_n^{(e_i)}(t)}{H_0^{(e_i)}(t)} \right)$$

を得る。

④ さらに ② の方法により $G_i(t) = 0$ の下で各有理式を有理化し，多項式表示解を得る。

以上のように，前節の数値の場合と，算法としては同等であるが，算法 2-② の数値計算は，必ずしも数式処理には向かない（実際 [1] では ② 以降は FORTRAN によっている)。今回，この算法の各段階における効率化技法を開発・実現し，全面的に記号式

で計算を行いうるように改良した。これらについて、次節で説明する。

5。行列式とその偏導関数の計算

上の算法において最も手間のかかる計算は、Λ の行列式Dの計算である。Λ 自体は、一般には零要素を多く含む疎な行列となるが、多変数でしかも容易に高次元（＝有限解の個数）になりうる。さらにその行列式であるDは、元の問題（2節のD式）からわかるように、密な斉次多項式となる。例えば、桂先生の問題の case 4（5変数5多項式）では、Λ は6変数多項式を要素とする 16×16 の行列となり、その行列式の多変数多項式を計算することは殆んど不可能である。

本算法で必要となる式は、行列式Dの多変数多項式そのものではなく、D及びその偏導関数の媒介変数表示である。このことに注意すれば、[1]で指摘されたように、$G(t)$ については行列 Λ の各要素 c_{ji} を媒介変数表示し、その後で行列式の計算

$$G(t) = \det\left(c_{ji}(u_0 = 1, u_1 = b_1 t, \cdots, u_n = b_n t) \right)$$

を行えばよいし、$H_j(t)$ 及び $H_j^{(e_i)}(t)$ についても、行列式の一階偏微分を

$$\frac{\partial}{\partial U_k} \det\left(C_{ji} \right) = \sum_m \det\begin{pmatrix} c_{11} & \cdots & \frac{\partial}{\partial U_k} c_{1m} & \cdots & c_{1n} \\ \vdots & & \vdots & & \vdots \\ c_{n1} & \cdots & \frac{\partial}{\partial U_k} c_{nm} & \cdots & c_{nn} \end{pmatrix}$$

と展開し、同様の操作を行えばよい。これにより、$G(t)$及び$H_j(t)$即ち単根の場合には、すべての行列式計算を一変数の場合に帰着させることができる。

さらに上式の右辺のΣ中の行列式の計算には、小行列展開を行うにしてもGauss消去法を用いるにしても、多くの共通部分式が含まれる。我々は、それらの再計算を避け、組織的に偏導関数を構成する方法を考案した[4]。例えばGauss消去法(1-step Bareiss)では、$\frac{\partial}{\partial U_k}$行を含まない部分の消去はすべてに共通して行い、$\frac{\partial}{\partial U_k}$行による消去以降は別々に消去を進めていく。これにより、単純に消去のための要素計算の個数を数えるだけでも約4分に1になる。さらにこれらの算法で、偏導関数の計算を、$\left(C_{ji} \right)$と$\left(\frac{\partial}{\partial U_k} C_{ji} \right)$の二つの行列から式を構成する演算とみなせば、補間法の拡張も自然に行われ、かつ補間値を計算する度に$\frac{\partial}{\partial U_k} C_{ji}$を再計算することも避けられる(研究会において、竹島(富士通国際研)、一松(数解研)両氏により、一変数行列式の計算には補間法が最適であるとの指摘があった。実際[3]の解析及び我々のその後の実験により実証されている)[4]。

重根の場合には，高階偏導関数が必要となるが，その場合には，U_0に対する代入は行わずに，U_0と媒介変数tの二変数多項式を要素として行列式及びその$\frac{\partial}{\partial U_j}$の計算を行い，得られた結果の多項式を$U_0$で偏微すればよい。ここで，階数分だけ前記展開法を適用するのは，和の項数が組合わせ的に増加するので得策ではない。この方法は，単根の場合の$H_j(t)$の計算にも適用しうるが，計算量では劣る。

以上で，記号解の構成に必要な行列式の計算は，すべて一変数，重根の場合でも二変数の多項式を要素とする行列の行列式計算に帰着させうることがわかった。行列式計算については，Aのように零要素の多い行列に対しては入れ子式小行列式展開法が向くとされているが，行列の次元が大きくなると殆んど計算不能（30×30程度で小行列式の個数がアドレス空間を超してしまう）となるので，f_iの個数や次数が小さくない限り不向きである。一変数或いは二変数と変数が少ないため，中間式の膨張は大したことがなく，Gauss消去法（Bareiss）でも充分に効率的であるが，効率及びメモリ（中間式の膨張はない）の両面から補間法が最適である。[4,5]

6. むすび

〔1〕で示されたU-終結式による代数方程式の解法アルゴリ

ズムを、記号式のまま適用し、すべての有限解を代数的数を導入して記号式で求めるよう発展させた。この算法による実際の計算で問題となるのは、多変数高次元の行列の行列式計算と、重根の場合の行列式の高階導関数の計算だが、本稿ではこの二点に対処する計算技法を紹介した。それらの手法は数学的にはあまり意味のないものだが、数式処理において計算量を軽減するには有効な手法である。実際，数式の爆発的膨張と多大な計算時間のために、[1] の算法そのままでは計算不能であった問題も、一変数化することにより計算可能となった（付録の実例参照）。

本稿では触れていないが，平面を与える L_i の媒介変数表示の決め方にも注意が必要である。数学的には重複度 e_i を正しく与えない可能性があり、また t の各多項式の係数は一般には長大な多倍長整数となり代数拡大がわかりにくくまた計算上も効率的でないという問題がある。前者については [1] に詳しいが，後者については未だ解決策はなく今後の課題である（効率という点では、中国人剰余定理の適用も試みたが値の範囲が厳密にわからない限り不利である）。

最後に、算法１のステップ②における簡約規則の生成と基底による展開，各の構成における各 $G_i(t)$ に対する $H_j(t)$ 達の計算，さらに行列式の計算等々、高レベルの計算において並列化が期待

できることも指摘しておこう。

文献

[1] H. Kobayashi, T. Fujise and A. Furukawa, "Solving Systems of Algebraic Equations by a General Elimination Method", to appear in JSC.

[2] 村尾裕一, 古川昭夫, 「不定方程式の p-進解法とその応用」, 数理解析研究所講究録 486, pp. 11-26 (1983).

[3] E. Horowitz and S. Sahni, "On Computing the Exact Determinant of Matrices with Polynomial Entries", JACM, Vol. 22, No. 1, pp. 38-50 (1975).

[4] H. Murao, "Symbolic Computation of the Derivatives of Matrix Determinants", in preparation.

[5] H. Murao, T. Fujise and H. Kobayashi, "Computational Aspects of U-resultant and its Factorization with Applications to Solving Algebraic Equations", in preparation.

柱問題 (case 2 ~ 3変数 3多項式)

```
              2           2        2
F1 := X  - X + 2*Y  + 2*Z

F2 := Y*(2*X + 2*Z - 1)

F3 := X + 2*Y + 2*Z - 1

ON TIME, TRACE, SUBSTDET]-BAREISS,DET]-UNIVEC;

TIME: 22 MS
MATRIX MM2;

MM2 := LAM]-MATRIX(F1, F2, F3, U)$
Time for Groebner basis calculation: 18ms
=====  INPUT POLYNOMIALS  =====

 2           2        2
X  - X + 2*Y  + 2*Z

Y*(2*X + 2*Z - 1)

X + 2*Y + 2*Z - 1

-----  GROEBNER BASIS : ((X Y Z) . 3)  -----

              3        2
(7*Y + 210*Z  - 79*Z  + 3*Z)/210

                         2
(10*Y*Z - Y + 12*Z  - 4*Z)/10

     2            2
(5*Y  - Y - 3*Z  + Z)/5

X + 2*Y + 2*Z - 1

*** D ([sum of total deg] - [# vars] + 1) = 3

Time for this calculation: 1ms

Time to compute the basis :2ms

===== 4 BASIS MONOMIALS =====

1

Y

Z

 2
Z
```

← 行列 Λ の作成

基底となる単項式

```
Time for making reduction rules: 10ms

----- 8 REDUCTION RULES FOLLOW -----

 3                      2
Z  ==> ( - 7*Y + 79*Z  - 3*Z)/210

    2                2
Y*Z  ==> (7*Y - 24*Z  + 8*Z)/140

                      2
Y*Z ==> (Y - 12*Z  + 4*Z)/10

 2               2
Y  ==> (Y + 3*Z  - Z)/5

    2                        2
X*Z  ==> ( - 7*Y + 124*Z  - 18*Z)/210

                       2
X*Z ==> ( - Y + 2*Z  + Z)/5

                          2
X*Y ==> (2*(Y + 3*Z  - Z))/5

X ==>  - 2*Y - 2*Z + 1

Rules are represented with the above basis. :2ms + 9ms (GC)
Matrix generated.  Time :5ms

TIME: 98 MS

ARRAY UFAC 50, GGGFAC 50$

URESULTANT := DET]-BAREISS MM2$

TIME: 44 MS
FACTORIZE(URESULTANT, UFAC);

3

URESULTANT;

             4          3              3              3             2   2
(5000*(84*U0  + 184*U0 *U1 + 36*U0 *U2 + 40*U0 *U3 + 136*U0 *U1  +

          2              2              2   2          2               2
     60*U0 *U1*U2 + 80*U0 *U1*U3 + 3*U0 *U2  + 18*U0 *U2*U3 + U0

        2              3              2              2                 2
     *U3  + 40*U0*U1  + 28*U0*U1 *U2 + 48*U0*U1 *U3 + 4*U0*U1*U2

                                      2          2                2
     + 24*U0*U1*U2*U3 + 4*U0*U1*U3  + U0*U2 *U3 + 2*U0*U2*U3  -

          3        4        3          3         2   2        2
     U0*U3  + 4*U1  + 4*U1 *U2 + 8*U1 *U3 + U1 *U2  + 6*U1 *U2*U3

            2   2          2                 2          3
     + 3*U1 *U3  + U1*U2 *U3 + 2*U1*U2*U3  - U1*U3 ))/420000
```

…正攻法

K2-3

```
UFAC 0;

1/84

UFAC 1;

U0 + U1
```

→ 答 (1, 0, 0)

```
UFAC 2;

      2                                      2
28*U0  + 24*U0*U1 + 12*U0*U2 + 4*U0*U3 + 4*U1  + 4*U1*U2 + 4*U1*U3

     2                 2
 + U2  + 2*U2*U3 - U3
```

→ 答 ?

```
UFAC 3;

3*U0 + U1 + U3
```

→ 答 (1/3, 0, 1/3)

```
GGG := UNIVJ-SUBSTJ-DET(MM2, 1, T, 2*T, 3*T)$

*** 1 is to be substituted for U0
```

→ constraint eq.

```
*** T is to be substituted for U1

*** 2*T is to be substituted for U2

*** 3*T is to be substituted for U3


TIME: 14 MS
FACTORIZE(GGG, GGGFAC);
```

hyperplane: (1, t, 2t, 3t)

```
3


TIME: 20 MS
S0 := SUBSTJ-DFJ-DET(MM2, U0, 1, T, 2*T, 3*T)$

TIME: 27 MS
S1 := SUBSTJ-DFJ-DET(MM2, U1, 1, T, 2*T, 3*T)$

TIME: 31 MS
S2 := SUBSTJ-DFJ-DET(MM2, U2, 1, T, 2*T, 3*T)$

TIME: 27 MS
S3 := SUBSTJ-DFJ-DET(MM2, U3, 1, T, 2*T, 3*T)$

TIME: 27 MS
RESULTANT(NUM S0, NUM GGG, T);

33339600000000000000000000000
```

o.k.

```
A1 := S1 / S0$

A2 := S2 / S0$

A3 := S3 / S0$
```

解 : (A1, A2, A3)

K2-4

constraint eq. により決まる交点、から、実際の解を求む。

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% ANSWERS %%%%%%%%%%%%%%%%%%%%%%%%%;
%---------- (1) -------------;

GGGFAC 1;

T + 1

LET T = -1;

WRITE "ANSWER FOR T = ", T, ": (", A1, ", ", A2, ", ", A3, ")"$

ANSWER FOR T = (-1): (1, 0, 0)
```

← 解

```
SUB(X=1, Y=0, Z=0, F1);
0

SUB(X=1, Y=0, Z=0, F2);

0

SUB(X=1, Y=0, Z=0, F3);

0
```

検算

```
%---------- (2) -------------;
GGGFAC 2;

4*T + 3

LET T = -3 / 4;

WRITE "ANSWER FOR T = ", T, ": (", A1, ", ", A2, ", ", A3, ")"$

ANSWER FOR T = ( - 3)/4: (1/3, 0, 1/3)
```

← 解

```
SUB(X=1/3, Y=0, Z=1/3, F1);

0

SUB(X=1/3, Y=0, Z=1/3, F2);

0

SUB(X=1/3, Y=0, Z=1/3, F3);

0
```

検算

```
%--------- (3) ------------;
 GGGFAC 3;

       2
31*T  + 60*T + 28

DEFMINPOL(GGGFAC 3);

       2
31*T  + 60*T + 28

A1 := S1 / S0$

A1;

                  3          2                                   3
(315000000*(267*T  + 716*T  + 632*T + 184))/(315000000*(457*T  +

              2
      1250*T  + 1128*T + 336))

A2 := S2 / S0$

A3 := S3 / S0$

AA1 := RATSIMP A1;

AA1 := ( - 31*T - 18)/28

AA2 := RATSIMP A2;

AA2 := ( - 31*T - 18)/56

AA3 := RATSIMP A3;

AA3 := (31*T + 32)/28

D := 8*SQRT(2)

T1 := (-60 + D) / (2 * 31);

T1 := (2*(2*SQRT(2) - 15))/31

T2 := (-60 - D) / (2 * 31);

T2 := (2*( - 2*SQRT(2) - 15))/31
```

代数 拡大

因みに 太は

```
LET T = T1;

WRITE "ANSWER FOR T = ", T, ": (", AA1, ", ", AA2, ", ", AA3, ")"$

ANSWER FOR T = (2*(2*SQRT(2) - 15))/31: (( - SQRT(2) + 3)/7, ( -
   SQRT(2) + 3)/14, (2*SQRT(2) + 1)/14)

SUB(X=AA1, Y=AA2, Z=AA3, F1);

0

SUB(X=AA1, Y=AA2, Z=AA3, F2);

0

SUB(X=AA1, Y=AA2, Z=AA3, F3);

0

LET T = T2;

WRITE "ANSWER FOR T = ", T, ": (", AA1, ", ", AA2, ", ", AA3, ")"$

ANSWER FOR T = (2*( - 2*SQRT(2) - 15))/31: ((SQRT(2) + 3)/7, (SQRT(
   2) + 3)/14, ( - 2*SQRT(2) + 1)/14)

SUB(X=AA1, Y=AA2, Z=AA3, F1);

0

SUB(X=AA1, Y=AA2, Z=AA3, F2);

0

SUB(X=AA1, Y=AA2, Z=AA3, F3);

0
```

柱問題 (case 3 — 4変数 4多項式)

K3-1

```
            2     2             2       2
F1  := 2*W   + X   - X + 2*Y   + 2*Z

F2  := 2*W*Z + 2*X*Y + 2*Y*Z - Y

                              2
F3  := 2*W*Y + 2*X*Z + Y   - Y

F4  := 2*W + X + 2*Y + 2*Z - 1

ON TIME, TRACE, SUBSTDET]-BAREISS, DET]-UNIVEC;

MM3 := LAM]-MATRIX(F1, F2, F3, F4, U)$
Time for Groebner basis calculation: 357ms
=====  INPUT POLYNOMIALS  =====

    2      2           2       2
2*W   + X  - X + 2*Y   + 2*Z

2*W*Z + 2*X*Y + 2*Y*Z - Y

                      2
2*W*Y + 2*X*Z + Y   - Y

2*W + X + 2*Y + 2*Z - 1

-----  GROEBNER BASIS : ((W X Y Z) . 4)  -----

        2                              4             3             2
(8788*Y   - 13150*Y*Z + 423*Y + 24948*Z   - 21624*Z   + 10712*Z   - 846
 *Z)/24948

      2          2               3          2
(27*Y *Z + 11*Y   - 23*Y*Z - 12*Z   + 10*Z )/27

(2*W + X + 2*Y + 2*Z - 1)/2

              2                     2
(3*X*Y - Y   - 2*Y*Z - 2*Y - 4*Z   + 2*Z)/3

                2                     2
(3*X*Z - 2*Y   - 4*Y*Z - Y - 2*Z   + Z)/3

     2                2                  2
(3*X   - 4*X + 12*Y   + 16*Y*Z + 16*Z   - 8*Z + 1)/3

     3     2                  3        2
(9*Y   - Y   - 2*Y*Z - 12*Z   + 4*Z )/9

          2            2                             3         2
( - 73*Y   + 216*Y*Z   + 88*Y*Z - 9*Y + 276*Z   - 176*Z   + 18*Z)/216

*** D ([sum of total deg] - [# vars] + 1) = 4

Time for this calculation: 2ms
```

```
Time to compute the basis :7ms

===== 8 BASIS MONOMIALS =====

1

X

Y

Z

 2
Y

Y*Z

 2
Z

 3
Z

Time for making reduction rules: 78ms

----- 20 REDUCTION RULES FOLLOW -----

 4                  2                                  3            2
Z  ==> ( - 8788*Y   + 13150*Y*Z - 423*Y + 21624*Z   - 10712*Z   + 846*
        Z)/24948

   3              2                             3         2
Y*Z  ==> (1453*Y   - 1480*Y*Z + 39*Y + 3144*Z   + 68*Z   - 78*Z)/8316

   2            2                          3          2
Y*Z  ==> (73*Y   - 88*Y*Z + 9*Y - 276*Z   + 176*Z   - 18*Z)/216

 2                  2                    3         2
Y *Z ==> ( - 11*Y   + 23*Y*Z + 12*Z   - 10*Z )/27

              .
              .
              .
              .

                     2                         2
W*Y ==> ( - 7*Y   - 8*Y*Z + Y - 4*Z   + 2*Z)/6

                       2                         2
W*X ==> ( - X + 6*Y   + 4*Y*Z - 6*Y + 4*Z   - 2*Z + 1)/6

W ==> ( - X - 2*Y - 2*Z + 1)/2

Rules are represented with the above basis. :15ms + 9ms (GC)
Matrix generated.  Time :26ms

TIME: 674 MS

URESULTANT := DET MM3$
```

正攻法

```
FACTORIZE(URESULTANT, URFAC);

4

TIME: 954 MS

URFAC 1;

U0 + U2
```

解 →

```
TIME: 3 MS
SUB(W=0,X=1,Y=0,Z=0, F2);

0

TIME: 4 MS
SUB(W=0,X=1,Y=0,Z=0, F3);

0

TIME: 4 MS
SUB(W=0,X=1,Y=0,Z=0, F4);

0

TIME: 5 MS
SUB(W=0,X=1,Y=0,Z=0, F1);

0

URFAC 2;

         5              4              4               4              4
24948*U0  - 12096*U0 *U1 + 38772*U0 *U2 + 53784*U0 *U3 + 1296*U0 *

           3   2              3                 3                   3
U4 + 3276*U0 *U1  - 1656*U0 *U1*U2 - 11412*U0 *U1*U3 - 18504*U0 *U1

               3   2             3                  3                  3
*U4 + 31896*U0 *U2  + 66240*U0 *U2*U3 - 18936*U0 *U2*U4 + 37368*U0

   2                         3  2            2   3
*U3                          ..  + 1694*U0 *U1  + 3..
```

→ ?

```
URFAC 3;

3*U0 + U1 + U2

URFAC 4;

3*U0 + U1 + U2
```

多重度!(2)

W X Y Z
(1/3, 1/3, 0, 0)


```
GGG := UNIV]-SUBST]-DET(MM3, 1, T, 2*T, 3*T, 4*T)$

*** 1 is to be substituted for U0

*** T is to be substituted for U1

*** 2*T is to be substituted for U2

*** 3*T is to be substituted for U3

*** 4*T is to be substituted for U4

TIME: 68 MS

                  8              7              6              5
GGG := (430506*T  + 2208519*T  + 4852355*T  + 5975243*T  + 4526479*

        4              3             2
       T  + 2171050*T  + 647748*T  + 110592*T + 8316)/8316

 2*T + 1

         5            4            3            2
 215253*T  + 566127*T  + 580354*T  + 296856*T  + 77328*T + 8316

 T + 1

 T + 1

S0 := SUBST]-DF]-DET(MM3, U0, 1, T, 2*T, 3*T, 4*T)$
TIME: 1134 MS
S1 := SUBST]-DF]-DET(MM3, U1, 1, T, 2*T, 3*T, 4*T)$

TIME: 1358 MS
OFF SUBSTDET]-BAREISS;

TIME: 2 MS
S2 := SUBST]-DF]-DET(MM3, U2, 1, T, 2*T, 3*T, 4*T)$

TIME: 1092 MS
S3 := SUBST]-DF]-DET(MM3, U3, 1, T, 2*T, 3*T, 4*T)$

TIME: 948 MS

S4 := SUBST]-DF]-DET(MM3, U4, 1, T, 2*T, 3*T, 4*T)$

TIME: 948 MS

A1 := S1 / S0$
A2 := S2 / S0$
A3 := S3 / S0$
A4 := S4 / S0$
```

T = -1 に対して.

K3-5.

```
PROCEDURE SUBSTT FFF; SUB(U0=1,U1=T,U2=2*T,U3=3*T,U4=4*T, FFF);
** 284 BYTES   5 MS     COMPILING SUBSTT

SUBSTT


TIME: 10 MS
AA1 := SUBSTT(DF(URESULTANT, U1, 2) / DF(URESULTANT, U0, U1));

                        6              5              4              3
AA1 := (2*( - 280278*T  - 1478265*T  - 2575843*T  - 2024438*T  -

                 2                                         6
          766464*T  - 127566*T - 6048))/(3*(2227731*T  + 6497516*

            5             4             3             2
           T  + 7455843*T  + 4406312*T  + 1498710*T  + 306936*T

            + 31752))


AA2 := SUBSTT(DF(URESULTANT, U2, 2) / DF(URESULTANT, U0, U2));


AA3 := SUBSTT(DF(URESULTANT, U3, 2)) / SUBSTT(DF(URESULTANT, U0, U3));

AA4 := SUBSTT(DF(URESULTANT, U4, 2)) / SUBSTT(DF(URESULTANT, U0, U4));
```

16

K3-6

```
DEFMINPOL GGGFAC 2;

        5            4            3            2
215253*T  + 566127*T  + 580354*T  + 296856*T  + 77328*T + 8316

AA1 := RATSIMP A1;

                                 4                     3
AA1 := (165533888966625*T  + 509355488951919*T  + 443766192375806*

        2
       T  + 40332981163668*T - 41214807676404)/65331259281684

                    .
                    .
                    .


SUB(W=AA1,X=AA2,Y=AA3,Z=AA4, F1);
                                        8
(29433233207263828991646620300559*T  +
                                        7
 126071571380074431434504466913362*T  +
                                        6
 232326498620684979948027343971 87*T  +
                                        5
 242641617677826046115664507131 88*T  +
                                        4
 158547049546382901942929028839 56*T  +
                                        3
 66909250043108878325705027206 60*T  +
                                        2
 17930177916667021434826418313 24*T  +

 2807830820169978502716114915 60*T + 19813194084102076744589897856)/

1508950205823957201899249 04

TIME: 45 MS
RATSIMP J*ANS;

0
```

桂問題 (case 4 — 5変数多項式)

K4-1

```
    GGG := UNIV]-SUBST]-DET(KATURA4]-MAT,
                            1, T, 2*T, 3*T, 4*T, 5*T);
*** 1 is to be substituted for U0
*** T is to be substituted for U1
*** 2*T is to be substituted for U2
*** 3*T is to be substituted for U3
*** 4*T is to be substituted for U4
*** 5*T is to be substituted for U5
GGG := (7098145657695O*T**16 + 900398903146545*T**15 +
5308552185119247*T**14 + 19305933303504198*T**13 +
48466146239968178*T**12 + 89045281349032332*T**11 +
123841123671293964*T**10 + 132984480066118920*T**9 +
111428523015620488*T**8 + 73097147827077504*T**7 +
37418939754192256*T**6 + 14791748447843072*T**5 +
4427009191281664*T**4 + 969912841830400*T**3 +
146729874343936*T**2 + 13697171963904*T + 594522120192)/
594522120192$

TIME: 4853 MS

    S0 := SUBST]-DF]-DET(KATURA4]-MAT, U0,
                         1, T, 2*T, 3*T, 4*T, 5*T);
S0 := (900398903146545*T**15 + 10617104370238494*T**14 +
57917799910512594*T**13 + 193864584959872712*T**12 +
445226406745161660*T**11 + 743046742027763784*T**10 +
930891360462832440*T**9 + 891428184124963904*T**8 +
657874330443697536*T**7 + 374189397541922560*T**6 +
162709232926273792*T**5 + 53124110295379968*T**4 +
12608866943795200*T**3 + 2054218240815104*T**2 +
205457579458560*T + 9512353923072)/594522120192$

TIME: 142472 MS

    S1 := SUBST]-DF]-DET(KATURA4]-MAT, U1,
                         1, T, 2*T, 3*T, 4*T, 5*T);
S1 := (1524360417028605*T**15 + 17912289474586308*T**14 +
97354470648651450*T**13 + 324601224242994252*T**12 +
742426146884514092*T**11 + 1233750259015997216*T**10 +
1538769921583043704*T**9 + 1466754319775729168*T**8 +
1077338275765495552*T**7 + 609804164279025664*T**6 +
263853412089347584*T**5 + 85716790100806144*T**4 +
20242122012755968*T**3 + 3281131629993984*T**2 +
326509755826176*T + 15040637534208)/1783566360576$

TIME: 154500 MS

    S2 := SUBST]-DF]-DET(KATURA4]-MAT, U2,
                         1, T, 2*T, 3*T, 4*T, 5*T);

S2 := (92365718902650*T**15 + 1053154614282015*T**14 +
5542736621474049*T**13 + 17856112767293304*T**12 +
39366771203272040*T**11 + 62899055576437064*T**10 +
75224591576047956*T**9 + 68559730267543112*T**8 +
48003301414366352*T**7 + 25817861948087840*T**6 +
10578468652053376*T**5 + 3242512634360576*T**4 +
719689402012672*T**3 + 109188311385088*T**2 + 10123910983680
*T + 432379723776)/891783180288$

TIME: 146433 MS
```

K4-2

```
    S3 := SUBST]-DF]-DET(KATURA4]-MAT, U3,
                         1, T, 2*T, 3*T, 4*T, 5*T);

S3 := (86243998909320*T**15 + 994786882777122*T**14 +
5304286358617797*T**13 + 17341149696126915*T**12 +
38870341855812892*T**11 + 63275144194427936*T**10 +
77277567838765664*T**9 + 72107543148988860*T**8 +
51836037617415256*T**7 + 28713627168251552*T**6 +
12158866736192320*T**5 + 3866338215996416*T**4 +
893967761060608*T**3 + 141940718619648*T**2 + 13843264800768
*T + 625406386176)/891783180288$

TIME: 143012 MS

    S4 := SUBST]-DF]-DET(KATURA4]-MAT, U4,
                         1, T, 2*T, 3*T, 4*T, 5*T);
S4 := (40042730827350*T**15 + 485955676628535*T**14 +
2729759104385856*T**13 + 9412469152405689*T**12 +
22273892745050774*T**11 + 38308995482457868*T**10 +
49459697536627108*T**9 + 48802596321802644*T**8 +
37100388337184336*T**7 + 21728112603929968*T**6 +
9722983595120480*T**5 + 3264746669086336*T**4 +
796300326907904*T**3 + 133207760464896*T**2 + 13667860257792
*T + 648569585664)/445891590144$

TIME: 141571 MS

    S5 := SUBST]-DF]-DET(KATURA4]-MAT, U5,
                         1, T, 2*T, 3*T, 4*T, 5*T);
S5 := (71027787272175*T**15 + 929806117432173*T**14 +
5586382163716050*T**13 + 20451863781848748*T**12 +
51056739577112484*T**11 + 92110602062085596*T**10 +
124108970926591136*T**9 + 127220184179306752*T**8 +
100062126370497968*T**7 + 60406585524972352*T**6 +
27770538187763584*T**5 + 9551082015607552*T**4 +
2379723775536640*T**3 + 405672444356608*T**2 +
42325698674688*T + 2038361554944)/1783566360576$


TIME: 140988 MS
```

解: $(S_1/S_0,\ S_2/S_0,\ S_3/S_0,\ S_4/S_0,\ S_5/S_0)$

但し T は GGG により定義される。

17

6

桂問題 (case 5 — 6変数6多項式)

Gröbner basis の計算	1573秒
基底単項式を求める	0.36
単項式の簡約規則の生成	70.6

$\Rightarrow$ Δ : 7変数 32×32

行列式計算は補間法により

因みに 2-step Bareissでは (Gauss消去法)
1183秒

$G(t)$ 100秒

$\frac{\partial|\Delta|}{\partial u_j}(t)$ 1899秒
40分 ×2
2032秒
2055
1978
2003