前処理によるBarnettの定理に基づく近似GCD計算の安定化の検討

Towards to Stable Approximate GCD Computation Based on Barnett's Theorem by Preprocessing

筑波大学医学医療系　　讃岐勝 *1
Masaru Sanuki
Faculty of Medicine, University of Tsukuba

Abstract

In this paper, we consider stabilizing the approximate GCD computation by Barnett's theorem by preprocessing the Bezout matrix. As preprocessing methods, we discuss numerical matrix construction methods and the removal of unstable elements/rows of matrix that are over-determined systems, using many numerical examples. The condition number is one measure of numerical instability that is algorithm-independent, but there are several algorithm-specific measures. In this paper, we also discuss the conversion to a dominant-diagonal matrix, which is a convergence condition for the Gauss-Seidel method and the Jacobi method.

1　はじめに

浮動小数係数の1変数多項式 $f(x), g(x) \in \mathbb{F}[x]$ を次で与える．

$$\begin{aligned} f(x) &= f_n x^n + \ldots + f_0 = c(x)\tilde{f}(x) + \Delta_f, \\ g(x) &= g_n x^n + \ldots + g_0 = c(x)\tilde{g}(x) + \Delta_g. \end{aligned}$$

その共通因子 $c(x)$ (係数は摂動を持つので，近似共通因子と呼ばれる) の中でも，最大次数のものは近似GCDと呼ばれ，分野の黎明期から現在に至るまで種々の研究がなされている．

近似GCDを計算するアルゴリズムにおいて，数値計算に基づく方法は非常に有効であり，これまで様々な方法が提案されてきた: QR法に基づく方法 [16, 9]，Sylvester行列の零空間を計算する方法 [8]，最適化手法 [25, 26]，および，実験として興味深いもの [18] などである．この中でもBarnettらによる構造行列からなる線型方程式に帰着した方法はシンプルであり [2, 3]，特に，Bezout行列やHankel行列からなる線型方程式に帰着した方法 [11] は様々な派生研究を生んでいる [4, 17]．本稿では，この中でBezout行列に基づく方法に関する前処理について考察を行うが，考察する内容は非常にマニアックである．そのため，Bezout行列の定義および近似GCDを求める方法 (Barnettの定理) を先に紹介し，その上で本稿で扱うテーマを述べる．

*1 〒 305-8575 茨城県つくば市天王台 1-1-1　E-mail: sanuki@md.tsukuba.ac.jp

定義 1 (Bezout 行列)
入力多項式 $f(x)$ と $g(x)$ について, 多項式 $\mathrm{Bpol}(f,g) = \dfrac{f(x)g(y)-f(y)g(x)}{x-y} = \displaystyle\sum_{1\le i,j\le n} b_{i,j}x^{i-1}y^{j-1} \in \mathbb{F}[x,y]$ の係数からなる *Bezout* 行列 $\mathrm{Bez}(f,g)$ とは，次の行列である.

$$\mathrm{Bez}(f,g) = \begin{pmatrix} b_{1,1} & \cdots & b_{1,n} \\ \vdots & \ddots & \vdots \\ b_{n,1} & \cdots & b_{n,n} \end{pmatrix} = (\boldsymbol{b}_1, \boldsymbol{b}_1, \cdots, \boldsymbol{b}_n) \in \mathbb{F}^{n\times n}. \tag{1}$$

すなわち，$\mathrm{Bez}(f,g)$ の (i,j)-要素 $b_{i,j}$ は，$\mathrm{Bpol}(f,g)$ の $x^{i-1}y^{j-1}$-係数である. ∎

命題 2 (Barnett の定理の改良版 [11])
$k = \deg(\gcd(f,g))$ とする．このとき, $n-k$ 個のベクトル $\boldsymbol{b}_{k+1}, \ldots, \boldsymbol{b}_n$ は一次独立であり, かつ, $\boldsymbol{b}_i (1 \le 0 \le k)$ は $\boldsymbol{b}_{k+1}, \ldots, \boldsymbol{b}_n$ によって張られ，さらに次の関係をみたす.

$$\boldsymbol{b}_i = c_{i,1}\boldsymbol{b}_{k+1} + \sum_{j=1}^{n-k-1} c_{i,1+j}\boldsymbol{b}_{k+1+j},\ for\ 1 \le i \le k. \tag{2}$$

ここで，各 $c_{i,1}$ は $\gcd(f,g) = c_k x^k + \cdots + c_0$ の x^{i-1} の係数 c_{i-1} を主係数 c_k で割った値 c_{i-1}/c_k に対応する，すわなち，$c_{i,1} = c_{i-1}/c_k$ である. ∎

実際に行う計算は

$$\mathrm{Bez}(f,g) \quad = \quad \left(\begin{array}{ccc|c} \boldsymbol{b}_1 & \cdots & \boldsymbol{b}_k & \mathrm{Bez}_{n-k}(f,g) \end{array} \right)$$

と行列を分解するとき ($\mathrm{Bez}_{n-k}(f,g) \in \mathbb{F}^{n\times(n-k)}$)，次の線形方程式系を解くことである (必要となるのは，解の第 1 要素のみである).

$$\mathrm{Bez}_{n-k}(f,g)\boldsymbol{x} = \boldsymbol{b}_i \text{ for } 1 \le i \le k \tag{3}$$

線型方程式 (3) は過剰決定系であり，k 個の行は削っても数学的には問題がないが，計算が不安定になるため数値的には問題がある．行の選び方によって安定性が変わることの実例 (例 1) を次に示す.

例 1
次の多項式 $f(x)$ と $g(x)$ の近似 *GCD* $c(x)$ を求める.

$$f_1(x) = c_1(x)(x^3+4x^2+3x-1), g_1(x) = c_1(x)(x^3-4x^2+1)\ with\ c_1(x) = 1/5.0x^2 - x + 1$$

$f_1(x)$ と $g_1(x)$ からなる *Bezout* 行列は次で表される.

$$\mathcal{B}^{(1)} = \left(\begin{array}{ccc|cc} -3.0 & 3.0 & & -2.60 & 2.0 & -0.400 \\ 3.0 & -17.0 & & 19.60 & -7.800 & 1.0 \\ -2.600 & 19.60 & & -12.92 & -1.200 & 0.9200 \\ 2.0 & -7.800 & & -1.200 & 6.2400 & -1.480 \\ -0.400 & 1.0 & & 0.9200 & -1.480 & 0.3200 \end{array} \right)$$

共通因子の主係数が小さいとき，部分行列部分 $\mathrm{Bez}_3(f,g)$ の条件数が近似 *GCD* の主係数に依存および次数に比例して大きくなることが知られており，近似 *GCD* の微小主係数問題と呼ばれる *[17]*. 実際に $\mathcal{B}^{(1)}$ の $3,4,5$ 行・$3\sim5$ 列から構成される部分行列 $\mathcal{B}^{(1)}_{345,3..5}$ の条件数は $\mathrm{cond}(\mathcal{B}^{(1)}_{345,3..5}) = 5.95\times10^4$ であり，条件数が大きくなっている．ここで，選択する行の組み合わせを変えることで条件数は次のように変化する.

選択した行	条件数
1,2,3 行目	$\mathrm{cond}(\mathcal{B}^{(1)}_{123,3..5}) = 1.43 \times 10^3$
1,2,4 行目	$\mathrm{cond}(\mathcal{B}^{(1)}_{124,3..5}) = 1.23 \times 10^3$
1,2,5 行目	$\mathrm{cond}(\mathcal{B}^{(1)}_{125,3..5}) = 1.81 \times 10^3$
1,3,5 行目	$\mathrm{cond}(\mathcal{B}^{(1)}_{135,3..5}) = 1.18 \times 10^3$
2,3,4 行目	$\mathrm{cond}(\mathcal{B}^{(1)}_{234,3..5}) = 4.50 \times 10^3$
2,3,5 行目	$\mathrm{cond}(\mathcal{B}^{(1)}_{235,3..5}) = 8.80 \times 10^3$

後ろ部分から選択すると部分行列は対称行列になり数学的には性質がよいが，精度が一番悪い． ∎

次の例 2 は主係数部と定数部を入れ替えた場合の例である (列のとり方を変えた).

例 2

次の多項式 $f(x)$ と $g(x)$ の近似 *GCD* $c_2(x)$ を求める.

$$f_2(x) = c_2(x)(x^3 + 4x - 1), g_2(x) = c_2(x)(x^3 - 4x^2 + 1),\ with\ c_2(x) = (0.1x - 1)(x + 0.5).$$

$f_2(x)$ と $g_2(x)$ からなる *Bezout* 行列は次で表される.

$$\mathcal{B}^{(2)}_{\mathrm{before}}(f_2, g_2) = \left(\begin{array}{cc||ccc} -1.0 & -0.900 & 1.6000 & -1.1500 & 0.1000 \\ -0.900 & -4.3100 & -4.7100 & 0.6150 & -0.0100 \\ \hline\hline 1.6000 & -4.7100 & -12.0450 & 7.2500 & -0.6000 \\ -1.1500 & 0.6150 & 7.2500 & 2.6700 & -0.340 \\ 0.1000 & -0.0100 & -0.6000 & -0.340 & 0.040 \end{array}\right)$$

$\mathcal{B}^{(2)}_{\mathrm{before:}}$ の 3 ∼ 5 行・3 ∼ 5 列から構成される部分行列 $\mathcal{B}^{(2)}_{\mathrm{before:3..5,3..5}}$ の条件数は 6.32×10^6 である．条件数が高くなるのは近似共通因子の主係数が小さいことに起因する *(全体の係数に比べて 1/10 程度小さい)*. ここで，次の変換によって主係数と定数項を入れ替える.

$$f_2(x) \to f_2(1/x) \times x^{\deg(f_2)}, g_2(x) \to g_2(1/x) \times x^{\deg(g_2)}$$

なる変換をする．ここで，$\deg(f)$ は $f(x)$ の次数を表す．この変換後の多項式で構成される *Bezout* 行列は，もとの *Bezout* 行列をひっくり返した行列に等しい.

$$\mathcal{B}^{(2)}_{\mathrm{after}} = \left(\begin{array}{cc||ccc} -0.040 & 0.340 & 0.6000 & 0.0100 & -0.1000 \\ 0.340 & -2.6700 & -7.2500 & -0.6150 & 1.1500 \\ \hline\hline 0.6000 & -7.2500 & 12.0450 & 4.7100 & -1.6000 \\ 0.0100 & -0.6150 & 4.7100 & 4.3100 & 0.900 \\ -0.1000 & 1.1500 & -1.6000 & 0.900 & 1.0 \end{array}\right)$$

${\mathcal{B}^{(2)}_{*}}_{3..5,3..5}$ の条件数は 6.32×10^6 から 235. と大幅に減少した．これは，近似 *GCD* の主係数が小さくなくなったことに起因する *(この例では，1/10 から 1/2 へ変更しており，この場合も微小主係数の影響は残っている)*. ∎

本稿で扱う話題

例 1 と例 2 より，列のとり方によって条件数が大きく変化することがわかる．そこで，本稿では計算アルゴリズムを変えることなく入力を変えるなどの前処理を検討することによって計算全体の安定化につい

て検討を行う．具体的には，Bezout 行列そのものを変えることの検討，および，部分行列の作り方を検討 (行の選択)，を数値例をもとに考察する．

2　Bezout 行列の再構成

本章では，入力多項式を変換することによって Bezout 行列の条件数が改善できないか検討を行う．2.1 節では入力多項式と Bezout 行列の関係を，2.2 節では変換方法と実際の特徴について述べる．

2.1　Bezout 行列の性質

次の性質がある[1)]．

1. $\mathrm{Bez}(af, g) = a\mathrm{Bez}(f, g)$ with $a \in F$
2. $\mathrm{Bez}(f, g) = \mathrm{Bez}(f + ag, g) = \mathrm{Bez}(f + ag, f + bg)$ with $a, b \in F$

多項式の和による変換を施しても，Bezout 行列は変化しないことから，前処理として有効でないことがわかる．

2.2　入力多項式の変換

近似 GCD の微小主係数が条件数を増大させる大きな要因であるので，主係数を大きくすることで改善できないか検討する．$x \to ax$ with $|a| > 1$ なる変換を実施したのが次の例である (結果，有効ではないことがわかる)．

例 3 (有効ではない変換：$x \to ax$)
例 *2* の多項式で検討する．

$$f_2(x) = c_2(x)(x^3 + 4x - 1), g_2(x) = c_2(x)(x^3 - 4x^2 + 1) \ with\ c_2(x) = (0.1x - 1)(x + 0.5).$$

$x \to ax$ という変換をした後，$a = 2, 5, 10$ とそれぞれ代入して部分行列の条件数を計算したところ，条件数は 6.32×10^6 から，次の値にそれぞれ変化した．

- $a = 2$*:* $\to 6.42 \times 10^5$
- $a = 5$*:* $\to 1.16 \times 10^5$ *(このときの行列を表示)*
- $a = 10$*:*$\to 7.02 \times 10^4$

Bezout 行列の要素が $\mathrm{Bpol}(f, g)$ の係数で，多項式の積和からなるために中間部の要素が大きく，かつ，両端部が小さくなる傾向がある．それゆえ，行列の要素も同様の傾向を示し，行列の両端が小さくなる傾向にある．

$$\left(\begin{array}{cc|ccc} -5.0 & -22.500 & 200.0 & -718.7500 & 312.5000 \\ -22.500 & -538.7500 & -2943.7500 & 1921.8750 & -156.2500 \\ \hline 200.0 & -2943.7500 & -37640.6250 & 113281.2500 & -46875.0 \\ -718.7500 & 1921.8750 & 113281.2500 & 208593.7500 & -132812.500 \\ 312.5000 & -156.2500 & -46875.0 & -132812.500 & 78125.0 \end{array}\right)$$

[1)] 前処理として使えない変換である

a が大きくするにつれて，最後の行および列は他に比べて小さくなってしまうため，結果的にさらに数値的に不安定な行列になることから，この変換は有効でない． ∎

k が小さい場合に，選択できる行が少ない．多項式から構成される行列のサイズが増えると選択できる行が増えることが期待できる．

行列のサイズを増大させても問題ないと考えた場合，次の変換がすぐに思いつく．

$$f(x) \to f'(x) = (ax+1)f(x), \quad g(x) \to g'(x) = (bx+1)g(x)$$

次は a, b の条件によって上記の変換の効果について検討した結果である．

変換について

1. $a \neq 0$ and $b = 0$ のとき
   $\mathrm{Bez}(f', g') \in \mathbb{F}^{(n+1)\times(n+1)}$ and $\mathrm{Bez}_{n-k}(f', g') \in \mathbb{F}^{(n+1)\times(n-k+1)}$ であり，近似 GCD そのものは変わらず次数も変化しないため部分行列の選び方の数は変わらない．

2. $a \neq b$ and $a, b \neq 0$ のとき
   $\mathrm{Bez}(F', G') \in \mathbb{F}^{(n+2)\times(n+2)}$ and $\mathrm{Bez}_{n-k}(F', G') \in \mathbb{F}^{(n+2)\times(n-k+1)}$ であり，近似 GCD そのものは変わらず次数は変化しないため部分行列の選び方の数は変わらない．

3. $a = b$ and $a, b \neq 0$ のとき
   $\mathrm{Bez}(F', G') \in \mathbb{F}^{(n+2)\times(n+2)}$ and $\mathrm{Bez}_{n-k-1}(F', G') \in \mathbb{F}^{(n+2)\times(n-k)}$ であり，であり，近似 GCD が変わり部分行列については選択可能な行が 2 行増える．近似 GCD そのものが変化し次数が 1 増えるため，解くべき線型方程式の個数は 1 つ増える．

次の例は $a = b$ の場合の数値例である．

例 4

例 *1* で与えられた多項式について，入力多項式を変換することによる前処理によって条件数の変化を見る．

$$f_1(x) = c_1(x)(x^3 + 4x^2 + 3x - 1),\ g_1(x) = c_1(x)(x^3 - 4x^2 + 1)\ \textit{with}\ c_1(x) = 1/5.0x^2 - x + 1$$

次の変換を行う．

$$f_1(x) \to f_1'(x) = (10x+1)f_1(x)\ \textit{and}\ g_1(x) \to g_1'(x) = (10x+1)g_1(x)$$

このとき，近似 *GCD* は $c(x)(10x+1)$ となり次数が増えている．構成される *Bezout* 行列 $\mathcal{B}^{(3)}$ の部分行列の条件数は次の通りである．部分行列の列のサイズは変わらず，行のサイズが *1* 増えたため，部分行列の組み合わせが増えている．

4,5,6 行	$\mathrm{cond}(\mathcal{B}^{(3)}_{456,4..6}) = 5.43 \times 10^4$
1,2,3 行	$\mathrm{cond}(\mathcal{B}^{(3)}_{123,4..6}) = 1.46 \times 10^5$
1,2,4 行	$\mathrm{cond}(\mathcal{B}^{(3)}_{124,4..6}) = 8.91 \times 10^3$
1,2,5 行	$\mathrm{cond}(\mathcal{B}^{(3)}_{125,4..6}) = 9.61 \times 10^2$
1,2,6 行	$\mathrm{cond}(\mathcal{B}^{(3)}_{126,4..6}) = 2.07 \times 10^2$
2,3,4 行	$\mathrm{cond}(\mathcal{B}^{(3)}_{234,4..6}) = 1.27 \times 10^3$
2,3,5 行	$\mathrm{cond}(\mathcal{B}^{(3)}_{235,4..6}) = 1.03 \times 10^3$
1,3,5 行	$\mathrm{cond}(\mathcal{B}^{(3)}_{135,4..6}) = 9.01 \times 10^3$
1,5,6 行	$\mathrm{cond}(\mathcal{B}^{(3)}_{156,4..6}) = 3.17 \times 10^3$

表 1: 条件数の変換について

組み合わせによっては条件数の改善が見られたが ($\mathcal{B}^{(3)}_{125,4..6}$ と $\mathcal{B}^{(3)}_{126,4..6}$)，すべての場合において改善されるわけではなく，選び方でばらつきがある傾向があるようである．

行の選択方法として，近い行を選ぶのではなく，互いに少しずつずつ離れた行を選択したときに条件数が大きくならない傾向があるようである．おそらく，共通因子に関係する線形関係がなくなると思われるが，今後さらなる理論的考察が必要である．

∎

3　反復法の観点でみる前処理

前章では，主に部分行列の条件数を小さくすることについて考察を行った．これは数値算法のアルゴリズムによらず，有効な方法である．本章では，反復法を例にアルゴリズムに依存する前処理について考察する．すなわち，反復法が収束するような前処理・変換が可能について検討をする．

3.1　Gauss-Seidel 法と Jaconi 法

本節では，線型方程式を解くための古典的な反復法である Gauss-Seidel 法，SOR 法[2] と Jacobi 法[3]を扱う[4]．

Gauss-Seidel 法と Jacobi 法は非常に似ているアルゴリズムであり次のように実行される．

線形方程式 $\mathcal{A}\boldsymbol{x} = \boldsymbol{b}$ が与えられたとき，行列 $\mathcal{A}$ を $\mathcal{A} = (\mathcal{L} + \mathcal{D} + \mathcal{U})$ と分解し次の反復式で $\mathcal{A}\boldsymbol{x} = \boldsymbol{b}$ を計算する．

- Gauss-Seidel 法
  反復式を次で構成.

$$(\mathcal{L} + \mathcal{D})\boldsymbol{x}^{(t+1)} = \boldsymbol{b} - \mathcal{U}\boldsymbol{x}^{(t)}$$

  であり，解の各成分は次で計算される．

$$x_i^{(t+1)} = \frac{1}{a_{i,i}}\Big(b_i - \sum_{j=1}^{i-1} a_{i,j}x_j^{(t+1)} - \sum_{j=i+1}^{n} a_{i,j}x_j^{(t)}\Big)$$

[2] 逐次過緩和法, successive overrelaxation method. Gauss-Seidel 法に加速パラメータを入れたもの
[3] 並列化可能．多変数への拡張の時に k 倍の高速化
[4] Krylov 部分空間法に比べて早いわけではない

- Jacobi 法
  反復式を次で構成.

$$\mathcal{D}\boldsymbol{x}^{(t+1)} = \boldsymbol{b} - (\mathcal{L}+\mathcal{U})\boldsymbol{x}^{(t)}$$

  であり，解の各成分は次で計算される (並列化可能であることが特徴).

$$x_i^{(t+1)} = \frac{1}{a_{i,i}}\Big(b_i - \sum_{j\neq j} a_{i,j}x_j^{(t)}\Big)$$

Gauss-Seidel 法と Jacobi 法の収束条件は次であることが知られている.

命題 3 (Gauss-Seidel 法と Jacobi 法の収束条件)
Gauss-Seidel 法と *Jacobi* 法は収束するための十分条件は行列 $\mathcal{A} = (a_{i,j})$ が *(*狭義*)* 優対角行列であることである.

$$|a_{i,i}| > \sum_{j\neq i} |a_{i,j}| \text{ for } 1 \leq i \leq n$$

次節では Bezout 行列の部分行列がこれらを満たすように変換できるのか，についても一緒に観察する.

3.2 優対角行列への変換の検討

例 5
次の多項式で検討する *(*微小主係数の共通因子を持たず，数値的には安定している*)*.

$$f_3(x) = (\underline{2x^2} - x + 1)(x^3 + 4x - 1), g_3(x) = (\underline{2x^2} - x + 1)(x^3 - 4x^2 + 1)$$

このときの *Bezout* 行列 $\mathrm{Bez}(f_3, g_3)$ は次のとおりである.

$$\mathrm{Bez}(f_3,g_3) = \left(\begin{array}{cc||ccc} -4 & 8 & -14 & 10 & -4 \\ 8 & -30 & 44 & -50 & 12 \\ \hline -14 & 44 & -62 & 64 & -8 \\ 10 & -50 & 64 & -84 & 8 \\ -4 & 12 & -8 & 8 & 16 \end{array}\right)$$

- $a(x) \neq b(x)$ *and* $b(x) = 1$ を改めて考える.

$$\begin{aligned} f_4(x) &= (a_1x - a_0)f_3(x) = (a_1x - a_0)(2x^2 - x + 1)(x^3 + 4x - 1) \\ g_4(x) &= g_3(x) = (2x^2 - x + 1)(x^3 - 4x^2 + 1) \end{aligned}$$

  このとき，*Bezout* 行列は次のようになる.

$$\left(\begin{array}{cc||ccc} 4a_0 + a_1 & -8a_0 - 5a_1 & 14a_0 + 6a_1 & -10a_0 - 9a_1 & 4a_0 + a_1 & -2a_1 \\ -8a_0 - 5a_1 & 30a_0 + 13a_1 & -44a_0 - 20a_1 & 50a_0 + 19a_1 & -12a_0 - 5a_1 & 2a_1 \\ \hline 14a_0 + 6a_1 & -44a_0 - 20a_1 & 62a_0 + 32a_1 & -64a_0 - 32a_1 & 8a_0 + 10a_1 & 4a_1 \\ -10a_0 - 9a_1 & 50a_0 + 19a_1 & -64a_0 - 32a_1 & 84a_0 + 19a_1 & -8a_0 - 3a_1 & -10a_1 \\ 4a_0 + a_1 & -12a_0 - 5a_1 & 8a_0 + 10a_1 & -8a_0 - 3a_1 & -16a_0 - a_1 & 18a_1 \\ -2a_1 & 2a_1 & 4a_1 & -10a_1 & 18a_1 & -4a_1 \end{array}\right)$$

  行列を眺めると，次のような傾向が見られる.

– 傾向：次数の大きな多項式の情報に置き換わりがち．$(a_1x - a_0)$ に引っ張られる．
– $f(x)$ のみの変換で優対角行列にするのは難しそう．

それゆえ，検討するべき変換は，$(F, G) \to (a(x)F(x), b(x)G(x))$ のみでよい．

例 6 $(a(x) \neq b(x))$
例 5 の多項式について，次の前処理を行う．

$$\begin{aligned} f_5(x) &= (3x-2)f_3(x) = (3x-2)(2x^2-x+1)(x^3+4x-1) \\ g_5(x) &= (4x-10)g_3(x) = (4x-10)(2x^2-x+1)(x^3-4x^2+1) \end{aligned}$$

このとき *Bezout* 行列は次の通り．

$$\left(\begin{array}{cc||cccc} -102 & 302 & -476 & 510 & -182 & 76 \\ 302 & -1062 & 1768 & -2018 & 902 & -188 \\ \hline -476 & 1768 & -3008 & 3408 & -1588 & 120 \\ 510 & -2018 & 3408 & -3866 & 1730 & 60 \\ -182 & 902 & -1588 & 1730 & -794 & -428 \\ 76 & -188 & 120 & 60 & -428 & 280 \end{array}\right)$$

部分行列のサイズが大きくなり，かつ，優対角行列にすることは難しいそうである．

例 7 $(a(x) = b(x))$
例 5 の多項式について，次の前処理を行う．

$$\begin{aligned} f_6(x) &= (100x-2)f_3(x) = (100x-2)(2x^2-x+1)(x^3+4x-1) \\ g_6(x) &= (100x-2)g_3(x) = (100x-2)(2x^2-x+1)(x^3-4x^2+1) \end{aligned}$$

このとき，*Bezout* 行列は次のようになる．

$$\left(\begin{array}{ccc||ccc} -12 & 612 & -632 & 1608 & -416 & 800 \\ 612 & -31268 & 35100 & -85532 & 27440 & -42000 \\ -632 & 35100 & -180128 & 263132 & -338592 & 99600 \\ \hline 1608 & -85532 & 263132 & -333440 & 341584 & 20800 \\ -416 & 27440 & -338592 & 341584 & -598272 & -46400 \\ 800 & -42000 & 99600 & 20800 & -46400 & 320000 \end{array}\right)$$

- 線型方程式に利用する行列サイズは変わらないが，解く方程式の本数が増える．

次の変換は入力多項式に乗する多項式を変えたものである．

$$\begin{aligned} f_7(x) &= (200x-2)f_3(x) = (200x-2)(2x^2-x+1)(x^3+4x-1) \\ g_7(x) &= (200x-2)g_3(x) = (200x-2)(2x^2-x+1)(x^3-4x^2+1) \end{aligned}$$

このときの *Bezout* 行列は次の通りである．

$$\left(\begin{array}{ccc||ccc} -12 & 1212 & -1232 & 3208 & -816 & 1600 \\ 1212 & -122468 & 130100 & -330932 & 94840 & -164000 \\ -1232 & 130100 & -700128 & 1026132 & -1337192 & 399200 \\ \hline 3208 & -330932 & 1026132 & -1306640 & 1343184 & 81600 \\ -816 & 94840 & -1337192 & 1343184 & -2396672 & -172800 \\ 1600 & -164000 & 399200 & 81600 & -172800 & 1280000 \end{array}\right)$$

また，乗する多項式を変えた別の変換についても見る．

$$\begin{aligned} f_8(x) &= (x-2)f_3(x) = (x-2)(2x^2-x+1)(x^3+4x-1) \\ g_8(x) &= (x-2)f_3(x) = (x-2)(2x^2-x+1)(x^3-4x^2+1)) \end{aligned}$$

このときの *Bezout* 行列は次の通りである．

$$\left(\begin{array}{ccc||ccc} -12 & 18 & -38 & 24 & -20 & 8 \\ 18 & -83 & 153 & -194 & 116 & -24 \\ -38 & 153 & -245 & 287 & -111 & 6 \\ \hline 24 & -194 & 287 & -404 & 133 & 10 \\ -20 & 116 & -111 & 133 & 84 & -68 \\ 8 & -24 & 6 & 10 & -68 & 32 \end{array}\right)$$

(f_8, g_8) とは異なる，乗する多項式を変えた別の変換についても見る．

$$\begin{aligned} f_9(x) &= (10x-2)(2x^2-x+1)(x^3+4x-1) \\ g_9(x) &= (10x-2)(2x^2-x+1)(x^3-4x^2+1) \end{aligned}$$

このときの *Bezout* 行列は次の通りである．

$$\left(\begin{array}{ccc||ccc} -12 & 72 & -92 & 168 & -56 & 80 \\ 72 & -488 & 900 & -1472 & 980 & -600 \\ -92 & 900 & -2828 & 3932 & -4152 & 960 \\ \hline 168 & -1472 & 3932 & -4760 & 4444 & 280 \\ -56 & 980 & -4152 & 4444 & -5712 & -1040 \\ 80 & -600 & 960 & 280 & -1040 & 3200 \end{array}\right)$$

この変換で，*(*対称*)* 優対角行列に変換できる場合の存在性が言えたが，数値を単純に設定することは難しいことも想像できる．

3.3 なぜ，優対角になりづらいのか？

なぜ，優対角になりづらいのか？ Bezout 行列が多項式の積 $\mathrm{Bpol}(f, g)$ の係数から構成される行列のため，中次くらいのところは，係数が大きくなる傾向がある．

前の章でも述べたが，優体格化は単純ではない．数値的に求めることが可能かを次の例で検討を行った．

例 8

f_3 と g_3 の次の前処理を考える *(*文字は 1 つにした*)*.

$$\begin{aligned} f_9(x) &= (ax-1)f_3(x) = (ax-1)(2x^2-x+1)(x^3+4x-1) \\ g_9(x) &= (ax-1)g_3(x) = (ax-1)(2x^2-x+1)(x^3-4x^2+1) \end{aligned}$$

a の値によって，*Bezout* 行列の部分行列が優対角になることはすでに示したか計算可能なのか，判定可能なのかを検討する．

まず，$f_9(x)$ と $g_9(x)$ の *Bezout* 行列を構成する．これの部分行列が優対角に変換可能なのかが判定できれば良い．

$$\left(\begin{array}{ccc||ccc} -3 & 3a+3 & -3a-8 & 8a+2 & -2a-4 & 4a \\ 3a+3 & -3a^2-6a-17 & 3a^2+25a+25 & -8a^2-27a-33 & 2a^2+37a+10 & -4a^2-10a \\ -3a-8 & 3a^2+25a+25 & -17a^2-50a-32 & 25a^2+65a+33 & -33a^2-43a+2 & 10a^2-2a \\ \hline 8a+2 & -8a^2-27a-33 & 25a^2+65a+33 & -32a^2-66a-60 & 33a^2+58a-4 & 2a^2+4a \\ -2a-4 & 2a^2+37a+10 & -33a^2-43a+2 & 33a^2+58a-4 & -60a^2+8a+32 & -4a^2-32a \\ 4a & -4a^2-10a & 10a^2-2a & 2a^2+4a & -4a^2-32a & 32a^2 \end{array}\right)$$

この部分行列が優対角になるのかは，変数 a の制約付き問題 *(a の次数は 2)* を解けばよく，これは難しくない．そのため，可能か否かの判定ができることがわかる． ∎

参 考 文 献

[1] A. V. Aho, J. E. Hopcroft and J. D. Ullman. *The design and analysis of computer algorithms.* Addison-Wesley, 1974.

[2] S. Barnett. *Greatest common divisor of two polynomials.* Linear Algebra Appl., **3**, 1970, 7–9.

[3] S. Barnett. *Greatest common divisor of several polynomials.* Proc. Camb. Phil. Soc., **70**, 1971, 263–268.

[4] D. Bini and P. Boito, *Structured matrix-based methods for polynomial ϵ-gcd: analysis and comparisons*, Proc. of ISSAC'07, ACM Press, 2007, 9–16.

[5] B. Beckermann and G. Labahn, *When are two numerical polynomials relatively prime?*, J. Symb. Comput., **26** (1998), 677–689.

[6] B. Beckermann and G. Labahn, *A fast and numerically stable Euclidean-like algorithm for detecting relatively prime numerical polynomials*, J. Symb. Comput., **26** (1998), 691–714.

[7] D. Bini and V.Pan, *Polynomial and Matrix Computations*, Birkhäuser, 1994.

[8] R. Corless, P. Gianni, B. Trager and S. Watt, *The singular value decomposition for polynomial systems*, Proc. of ISSAC'95, ACM Press, 1995, 195–207.

[9] R. Corless, S. Watt and L. Zhi, *QR factoring to compute the GCD of univariate approximate polynomials*, IEEE Trans. Signal Proces., **52(12)** (2004), 3394–3402.

[10] E.-W. Chionh, M. Zhang and R. N. Goldman. *Fast computation of the Bezout and Dixon resultant matrices.* J. Symb. Compu., **33**(2002), 13–20.

[11] G. M. Diaz-Toca and L. Gonzalez-Vega. *Barnett's theorems about the greatest common divisor of several univariate polynomials though Bezout-like matrices. J. Symb. Compu.*, **34**, (2002), 59–81.

[12] G. H. Golub and C. F. Van Loan, *Matrix computations*, Johns Hopkins Univ. Press, Baltimore, Maryland, 1989.

[13] U. Helmke and P. A Fuhrmann. *Bezoutians.* Linear Algebra Appl., **122/123/124**, 1989, 1039–1097.

[14] D.E. Knuth. *The Art of Computer Programming, Volume 2 (3rd Ed.): Seminumerical Algorithms*, Addison-Wesley Longman Publishing Co., Inc. 1997.

[15] R. T. Moenck. *Fast computation of GCDs. Proc. 5th ACM Symp. Thory of Comput.*, 1973, 142–151.

[16] H. Ohsako, H. Sugiura and T. Torii. A stable extended algorithm for generating polynomial remainder sequence (in Japanese). *Trans. of JSIAM (Japan Society for Indus. Appl. Math.)* **7** (1997), 227–255.

[17] M. Sanuki. *Computing multivariate approximate GCD based on Barnett's theorem*, Proc. of Symbolic-Numeric Computation 2009 (SNC 2009), 2009, 149-157, 2009.

[18] M. Sanuki. *Challenge to fast and stable computation of approximate univariate GCD, based on displacement structures*, Proc. of SNC2011, ACM Press, 2011, 178–186.

[19] 讃岐 勝. 悪条件性に注目した近似GCDの見積もり、京都大学数理解析研究所講究録、2015 (to appear)

[20] A. Schönhage. Quasi-GCD. *J. Complexity*, **1**, 1985, 118–147.

[21] T. Sasaki and F. Kako, *An algebraic method for separating close-root clusters and the minimum root separation*, International Workshop on Symbolic-Numeric Computation 2005 (SN C 2005), D. Wang & L. Zhi (Eds.), 2005, 126–143.

[22] T. Sasaki and M-T. Noda, *Approximate square-free decomposition and root-finding of ill-conditioned algebraic equations*, J. Inform. Proces., **12** (1989), 159–168.

[23] M. Sanuki and T. Sasaki, *Computing approximate GCDs in ill-conditioned cases*, Proc. of SNC 2007, 2007, 170–179.

[24] D. Sun and L. Zhi, *tructured low rank approximation of a bezout matrix*, *Mathematics in Computer Science*, 1(2):427–437, Dec 2007.

[25] A. Terui, *An iterative method for calculating approximate GCD of univariate polynomials.* In *Proceedings of the 2009 International Symposium on Symbolic and Algebraic Computation - ISSAC '09*, pages 351–358. ACM Press, New York, New York, USA, 2009.

[26] A. Terui. *GPGCD, an iterative method for calculating approximate GCD, for multiple univariate polynomials*, In *Computer Algebra in Scientific Computing*, pages 238–249.

[27] J. R. Winkler and X. Lao, *The calculation of the degree of an approximate greatest common divisor of two polynomials*, J. of Comp. and Appl. Math., **235(6)**, 2011, 1587–1603.

[28] T. Y. Li and Z. Zeng, *A rank-revealing method with updating, downdating, and applications*, SIAM J. Matrix Anal. Appl., **26** (2005), no. 4, 918–946.

[29] Z. Zeng, *The approximate GCD of inexact polynomials part I: a univariate algorithm*, to appear, 2004.

[30] L. Zhi, *Displacement structure in computing the approximate GCD of univariate polynomials*, Proc. of ASCM2003, World Scientific, 2003, 288–298.

[31] C. J. Zarowski, X. Ma and F. W. Fairman, *QR-factorization method for computing the greatest common divisor of polynomials with inexact coefficients*, IEEE Trans. Signal Proces., **48(11)** (2000), 3042–3051.