Bézoutの終結式行列を用いたGPGCD法による1変数多項式の近似GCDの計算 II

Algorithm for Calculating Approximate GCD of Univariate Polynomials with the Bézout Resultant Matrix II

筑波大学数理物質科学研究科　池 泊明 *1

Chi Boming

Graduate School of Pure and Applied Sciences University of Tsukuba

筑波大学数理物質系　照井 章 *2

Akira Terui

Faculty of Pure and Applied Sciences University of Tsukuba

Abstract

我々はこれまでに，1 変数多項式に対する近似 GCD 計算の反復算法である GPGCD アルゴリズムを提案している．本稿では，オリジナルの GPGCD アルゴリズムに用いた Sylvester の終結式行列に代えて，Bézout の終結式行列を用いた GPGCD アルゴリズムを提案し，その実験結果について述べる．実験は，与えられた多項式が高次の場合に，本アルゴリズムが求めた近似 GCD の精度がオリジナルの GPGCD アルゴリズムよりよいことと，計算効率について，本アルゴリズムが SNTLS アルゴリズムより良いことを示した．

Abstract

For a given pair of univariate polynomials with real coefficients and a given degree, we propose a modification of the GPGCD algorithm, presented in our previous research, for calculating approximate greatest common divisor (GCD). In the proposed algorithm, the Bézout matrix is used in transferring the approximate GCD problem to a constrained minimization problem, whereas, in the original GPGCD algorithm, the Sylvester subresultant matrix is used. Experiments show that, in the case that the degrees of the given polynomials are large, the proposed algorithm computes more accurate approximate GCDs than those computed by the original algorithm. They also show that the proposed algorithm is more efficient than the SNTLS algorithm, which also uses the Bézout matrix, with smaller amount of perturbations on the given polynomials.

*1 〒 305-8571 茨城県つくば市天王台 1-1-1　E-mail: hakumei-t@math.tsukuba.ac.jp

*2 〒 305-8571 茨城県つくば市天王台 1-1-1　E-mail: terui@math.tsukuba.ac.jp

1　はじめに

最大公約子 (GCD) 計算は数式処理において最も基本的かつ重要な計算の一つであり，近似 GCD 計算は数式・数値融合計算の中でも古くから研究されているテーマの一つである．近似 GCD 計算には，多項式剰余列 ([1], [11], [12]),（部分）終結式行列の低ランク近似 ([4], [5], [7], [8], [9], [13], [17], [19])，最適化 ([2], [10], [14], [18]) など，さまざまなアプローチがある．

前稿では，最適化法に基づく近似 GCD 計算アルゴリズムの一つである Sylvester の終結式行列を用いた GPGCD アルゴリズム [16] をもとに，Bézout の終結式行列を用いた GPGCD アルゴリズムを提案した [20]．本稿では，Bézout の終結式行列を用いた GPGCD アルゴリズムを改良し，実験をし，実験結果を Sylvester 法と SNTLS アルゴリズム [14] と比較した．

2　近似 GCD 計算の制約付き最適化問題への帰着

本稿では，近似 GCD 計算を以下の問題を解く形で考える．

問題 1
実係数の m 次多項式 $f(x)$, n 次多項式 $g(x)$ と正整数 d に対し，

$$\begin{gathered}\tilde{f}(x)=f(x)+\Delta f(x)=h(x)\bar{f}(x),\quad \tilde{g}(x)=g(x)+\Delta g(x)=h(x)\bar{g}(x),\\ \deg(h(x))=d,\quad \gcd(\bar{f}(x),\bar{g}(x))=1,\\ \deg(\Delta f(x))\le\deg(f(x)),\quad \deg(\Delta g(x))\le\deg(g(x)),\end{gathered}\tag{1}$$

を満たし，$\|\Delta f(x)\|^2+\|\Delta g(x)\|^2$ を最小化する $\tilde{f}(x)$, $\tilde{g}(x)$, $h(x)$ を求めよ．　□

本稿では，多項式のノルム $\|\cdot\|$ を 2 ノルム $\|\cdot\|_2$ とする．
与えられた多項式 $f(x), g(x)$ を

$$f(x)=f_m x^m+\cdots+f_0x^0,\quad g(x)=g_nx^n+\cdots+g_0x^0\tag{2}$$

とし，求める近似多項式 $\tilde{f}(x)$, $\tilde{g}(x)$ を

$$\tilde{f}(x)=\tilde{f}_m x^m+\cdots+\tilde{f}_0x^0,\quad \tilde{g}(x)=\tilde{g}_nx^n+\cdots+\tilde{g}_0x^0\tag{3}$$

とおく．一般性を失うことなく，$m\ge n$ と仮定する．以下では，$g(x)$ を m 次多項式 $g_mx^m+\cdots+g_0x^0$, $g_m=\cdots=g_{n+1}=0$ として扱う．
Bézout の終結式行列の定義と性質を以下に述べる．

定義 1 (Bézout の終結式行列 [6])
m 次多項式

$$f(x)=f_m x^m+\cdots+f_0x^0,\quad g(x)=g_mx^m+\cdots+g_0x^0$$

に対し，

$$b_{ij}=\sum_{d=1}^{m_{ij}} f_{j+d-1}g_{i-d}-f_{i-d}g_{j+d-1},\quad m_{ij}=\min\{i,m+1-j\},\tag{4}$$

を満たす行列 $\mathrm{Bez}(f,g)=(b_{ij})_{i,j=1,\dots,m}$ を f と g の ***Bézout*** の終結式行列と呼ぶ．　□

補題 2 (Bézout の終結式行列の性質 [6])
m 次多項式 f, g に対し，$\mathrm{Bez}(f,g)$ は m 次正方行列かつ対称行列であり，以下の関係を満たす．ここに，行列 A の階数を $\mathrm{rank}(A)$ で表す．

$$m - \mathrm{rank}(\mathrm{Bez}(f,g)) = \deg(\gcd(f,g)). \tag{5}$$

□

以下では，$\tilde{B} = \mathrm{Bez}(\tilde{f},\tilde{g})$ とおく．
問題 1 の条件式 (1) より

$$\deg(\tilde{f},\tilde{g}) = \deg(h(x)) = d \tag{6}$$

とおく．このとき，式 (6) と補題 2 より，

$$\mathrm{rank}(\tilde{B}) = m - d,$$

が成り立つ．
$\tilde{B} = (\tilde{b}_{ij})$ の特異値分解を $\tilde{B} = \tilde{U}\tilde{\Sigma}\tilde{V}^T$ とし，特異ベクトルである行列 $\tilde{V} = (\tilde{\boldsymbol{v}}_1, \cdots, \tilde{\boldsymbol{v}}_m)$ の後ろから k 個目の列ベクトルを

$$\tilde{\boldsymbol{v}} = \tilde{\boldsymbol{v}}_{m-d+1} = (\tilde{v}_1, \ldots, \tilde{v}_m) \tag{7}$$

とおく．補題 2 と特異値分解の性質より，

$$\tilde{B}\tilde{\boldsymbol{v}} = \boldsymbol{0} \tag{8}$$

が得られる．よって，制約条件を

$$G_i = \tilde{b}_{i1}\tilde{v}_1 + \cdots + \tilde{b}_{im}\tilde{v}_m = 0, \quad 1 \le i \le m \tag{9}$$

とおく．式 (4) より，$\tilde{b}_{ij}$ は係数 $\tilde{f}_i$ と $\tilde{g}_i$ より表せる：

$$\tilde{b}_{ij} = \sum_{d=1}^{m_{ij}} \tilde{f}_{j+d-1}\tilde{g}_{i-d} - \tilde{f}_{i-d}\tilde{g}_{j+d-1}, \quad m_{ij} = \min\{i, m+1-j\}. \tag{10}$$

与えられた多項式 $f(x)$, $g(x)$ の Bézout 行列を $B = \mathrm{Bez}(f(x), g(x))$ とする．$B = (b_{ij})$ の特異値分解を $B = U\Sigma V^T$ とし，$V = (\boldsymbol{v}_1, \cdots, \boldsymbol{v}_m)$ の後ろから d 個目の列ベクトルを

$$\boldsymbol{v} = \boldsymbol{v}_{m-d+1} = (v_1, \ldots, v_m)^T \tag{11}$$

とする．ここでは，多項式 $\tilde{f}, \tilde{g}$ の f, g からの摂動，および $\tilde{\boldsymbol{v}}$ の $\boldsymbol{v}$ からの摂動を最小化する．よって，目的関数を

$$F = \sum_{i=0}^{m}(\tilde{f}_i - f_i)^2 + \sum_{i=0}^{n}(\tilde{g}_i - g_i)^2 + \sum_{i=1}^{m}(\tilde{v}_i - v_i)^2, \tag{12}$$

とおく．
以上により，

$$\begin{aligned} \boldsymbol{x} &= (x_1, x_2, \ldots, x_{2m+n+2})^T \\ &= (\tilde{f}_0, \ldots, \tilde{f}_m, \tilde{g}_0, \ldots, \tilde{g}_n, \tilde{v}_1, \ldots, \tilde{v}_m)^T. \end{aligned} \tag{13}$$

に対し，制約条件を $\boldsymbol{G}(\boldsymbol{x}) = (G_1(\boldsymbol{x}), \ldots, G_m(\boldsymbol{x}))^T = \boldsymbol{0}$，目的関数を $F(\boldsymbol{x})$ とおき，制約付き最適化問題を以下の通り定める．

問題 2 (制約付き最適化問題)
制約条件 $\boldsymbol{G}(\boldsymbol{x}) = (G_1(\boldsymbol{x}), \ldots, G_m(\boldsymbol{x}))^T = \boldsymbol{0}$ のもとで，目的関数 $F(\boldsymbol{x})$ を最小化する $\boldsymbol{x}$ を求めよ． □

3　修正 Newton 法

本稿では，問題 2 を修正 Newton 法 [15] によって解く．修正 Newton 法は勾配射影法の一つの拡張であり，オリジナルの GPGCD アルゴリズムにも用いられている．

2 回連続微分可能な関数 $F(\boldsymbol{x})$ と $G_i(\boldsymbol{x}), i = 1, \dots, m$ に対し、制約付き最適化問題を制約条件 $\boldsymbol{G}(\boldsymbol{x}) = (G_1(\boldsymbol{x}), \dots, G_m(\boldsymbol{x}))^T = \boldsymbol{0}, G_i(\boldsymbol{x}) : \mathbb{R}^n \to \mathbb{R}$ のもとで，目的関数 $F(\boldsymbol{x}) : \mathbb{R}^n \to \mathbb{R}$ を最小化する問題とする．$\boldsymbol{g}(\boldsymbol{x}_k) = 0$ を満たす $\boldsymbol{x}_k$ に対し，探索方向 $\boldsymbol{d}_k$ と乗数 $\boldsymbol{\lambda}_k$ は次式より計算される：

$$\begin{pmatrix} I & -(J_{\boldsymbol{g}}(\boldsymbol{x}_k))^T \\ J_{\boldsymbol{g}}(\boldsymbol{x}_k) & O \end{pmatrix} \begin{pmatrix} \boldsymbol{d}_k \\ \boldsymbol{\lambda}_{k+1} \end{pmatrix} = - \begin{pmatrix} \nabla f(\boldsymbol{x}_k) \\ \boldsymbol{g}(\boldsymbol{x}_k) \end{pmatrix}, \tag{14}$$

ここに， $J_{\boldsymbol{g}}(\boldsymbol{x})$ はヤコビ行列である：

$$J_{\boldsymbol{g}}(\boldsymbol{x}) = \frac{\partial g_i}{\partial f_j}. \tag{15}$$

アルゴリズム 1 (修正 Newton 法 [15].)

Step 1 $\boldsymbol{x}_k$ に対し，式 (1) より探索方向 $\boldsymbol{d}_k$ を計算する．

Step 2 $\|\boldsymbol{d}_k\|$ が充分に小さい場合に（$\|\boldsymbol{d}_k\| \le \epsilon$），$\boldsymbol{x}_k$ を出力する．そうでない場合，適切なステップ幅 α_k を選んで，$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha_k \boldsymbol{d}_k$ とし，*Step 1* に戻る．

4　近似 GCD の計算アルゴリズム

本章では，Bézout 行列を用いた近似 GCD 計算アルゴリズムについて述べる．修正 Newton 法における初期値は以下に示す．

4.1　ヤコビ行列の表現

制約条件 (9) と目的関数 (12) より，ヤコビ行列 (15) の要素を以下に示す：

$$\begin{aligned} \frac{\partial G_i}{\partial \tilde{\boldsymbol{x}}_j} &= \sum_{l=1}^{m} \frac{\partial \tilde{b}_{il}}{\partial \tilde{\boldsymbol{x}}_j} \tilde{v}_l \\ &= \begin{cases} -\sum_{k=0}^{i-1} \tilde{g}_k \tilde{v}_{j-i+k} & 1 \le i < j \le m+1 \\ \sum_{k=0}^{m-i} \tilde{g}_{i+k} \tilde{v}_{j+k} & 1 \le j \le i \le m \\ \sum_{k=0}^{i-1} \tilde{f}_k \tilde{v}_{j-m-i+k-1} & 1 \le i < j-m-1 \\ & \quad \le n+1 \\ -\sum_{k=0}^{m-i} \tilde{f}_{i+k} \tilde{v}_{j-m+k-1} & 1 \le j-m-1 \\ & \le \min\{i, n+1\} \le m \\ \tilde{b}_{i,j-m-n-2} & m+n+2 < j \\ & \quad \le 2m+n+2 \end{cases}. \end{aligned}$$

ここに，$\tilde{b}_{ij}$ は式 (10) より示される．ヤコビ行列のサイズは $m \times (2m+n+2)$ である．

4.2　初期値の設定

式 (2) で与えられる多項式 $f(x)$ の係数 f_i，$g(x)$ の係数 g_i，式 (11) で与えられる特異ベクトルの成分 $(v_1, \ldots, v_m)^T$ を用いて，初期値 $\boldsymbol{x}_0$ を以下の通り定める：

$$\boldsymbol{x}_0 = (f_0, \ldots f_m, g_0, \ldots, g_n, v_1, \ldots, v_m). \tag{16}$$

4.3　近似 GCD の計算

$\boldsymbol{x}^* = (\tilde{f}_0, \ldots \tilde{f}_m, \tilde{g}_0, \ldots, \tilde{g}_n, \tilde{v}_1, \ldots, \tilde{v}_m)$ を修正 Newton 法より出力する最適値とする．補題 2 より，Bézout 行列 $\tilde{B} = \mathrm{Bez}(\tilde{f}, \tilde{g})$ から $\tilde{f}(x)$ と $\tilde{g}(x)$ の GCD を計算する．

4.4　近似 GCD の計算アルゴリズム

以上により、近似 GCD の計算アルゴリズムを以下に示す：

アルゴリズム 2 (Bézout 行列を用いた GPGCD アルゴリズム)
入力：

- $F(x), G(x) \in \mathbb{R}(x) : \deg(f(x)) \geq \deg(g(x)) > 0$ を満たす与えられた多項式，
- $d \in \mathbb{N}$: 与えられた近似 *GCD* の次数 $d \leq \deg(g)$，
- $\epsilon > 0$: アルゴリズム *1* の中の停止条件，
- $0 < \alpha \leq 1$: *Algorithm 1* の *Step 2* のステップ幅.

出力：

- $\tilde{h}(x) : \deg(\tilde{H}) = d$ を満たす近似 *GCD*,
- $\tilde{f}(x), \tilde{g}(x)$: *GCD* $\tilde{h}$ を持つ近似多項式 f と g.

Step 1 初期値を式 (16) より設定し，式 (4.1) のヤコビ行列を計算する．*Bézout* 行列は効率的に $O(n^2)$ の計算量で計算できることに注意する ([3]).

Step 2 式 (14) の探索方向 $\boldsymbol{d}_k$ および乗数 $\boldsymbol{\lambda}_{k+1}$ を計算する．

Step 3 $\|\boldsymbol{d}_k\| < \epsilon$ の場合に，$\boldsymbol{x}^*$ を $\boldsymbol{x}_k$ とし，多項式 $\tilde{f}(x)$ と $\tilde{g}(x)$ を計算し，Step 4 に進む．そうでない場合に，$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha \boldsymbol{d}_k$ とし，Step 2 に戻る．

Step 4 補題 2 より近似 GCD $\tilde{h}(x)$ を計算する．$\tilde{f}(x)$，$\tilde{g}(x)$ と $\tilde{h}(x)$ を出力する．

5 実験

我々は数理処理システム Maple 2016 を用いて本アルゴリズムの検証を行った[1)]．実験に用いた多項式の組 $f(x)$ と $g(x)$ は以下のように生成した：

$$
\begin{aligned}
f(x) &= f_0(x)h(x) + \frac{e_f}{\|f_N(x)\|} f_N(x), \\
g(x) &= g_0(x)h(x) + \frac{e_g}{\|g_N(x)\|} g_N(x).
\end{aligned}
\tag{17}
$$

ここに，多項式 $f_0(x)$，$g_0(x)$ は互いに素である．多項式 $f_0(x)$，$g_0(x)$，$h(x)$，$f_N(x)$，$g_N(x)$ が次数がそれぞれ $m-d$，$n-d$，d，$m-1$，$n-1$ の多項式であり，すべての係数は無作為に与えられた絶対値が 10 を超えない浮動小数である．式 (17) より，m 次多項式 $f(x)$ と n 次多項式 $g(x)$ の組を生成した．

実験では，$f(x)$ と $g(x)$ の次数の組 (m,n) に対し，100 組のテスト多項式を生成して実験を行った．次数 (m,n) のグループは合計 10 グループ作成した．各グループの (m,n,d) の値を表 1 に示す．今回の実験では，摂動多項式 e_f と e_g のノルムは $e_f = e_g = 0.01$ とする．

実験環境は表 2 に示す通りである．

5.1 近似 GCD の判定基準

これまで，近似 GCD の精度の検証においては，与えられた多項式 $f(x)$，$g(x)$ および近似 GCD アルゴリズムによる計算結果 $\tilde{f}(x)$，$\tilde{g}(x)$，$\tilde{h}(x)$ に対し，

$$
\sqrt{\|f(x)-\tilde{f}(x)\|_2^2 + \|g(x)-\tilde{g}(x)\|_2^2} \tag{18}
$$

を近似 GCD の精度の一つの基準にしていた．しかし，本稿の実験で，式 (18) の値が小さいにもかかわらず，$\tilde{f}(x)$ や $\tilde{g}(x)$ を $\tilde{h}(x)$ で割った剰余[2)]の 2 ノルムが大きくなる事例が見られた．

そこで，本稿では，近似 GCD の判定基準を以下の通り定めることにする．

多項式 $\tilde{f}(x)$ を $\tilde{h}(x)$ で割った剰余を $\hat{f}(x)$ とし，$\tilde{g}(x)$ を $\tilde{h}(x)$ で割った剰余を $\hat{g}(x)$ とする．この時，計算された近似 GCD が適合するなら，式 (18) がある実数を超えず，かつ剰余の 2 ノルム

$$
\sqrt{\|\hat{f}\|_2^2 + \|\hat{g}\|_2^2} \tag{19}
$$

が充分に小さい．

今回の実験における計算された近似 GCD の判定基準は以下の通りにした：

1. $\sqrt{\|f(x)-\tilde{f}(x)\|_2^2 + \|g(x)-\tilde{g}(x)\|_2^2} \le 1.414 \times 10^{-2} (\approx \sqrt{0.01^2 + 0.01^2})$,
2. $\sqrt{\|\hat{f}\|_2^2 + \|\hat{g}\|_2^2} \le 10^{-5}$.

判定基準 1：計算された近似多項式の摂動が与えられた摂動を超えない．

判定基準 2：剰余のノルム $\sqrt{\|\hat{f}\|_2^2 + \|\hat{g}\|_2^2}$ が 10^{-5} より小さい．

1) 本稿で用いたソースコードは以下の場所で提供している： `https://github.com/ct1counter/Bezout-gpgcd`

2) 剰余は Maple の組み込み関数 `SNAP:-Remainder` より計算する．

5.2 実験結果

本アルゴリズムを用いてグループ 1 からグループ 10 までの多項式の近似 GCD を求めた．比較をするため，オリジナルの GPGCD アルゴリズムを用いてグループ 1 からグループ 10 までの多項式の近似 GCD を求め，さらに，本アルゴリズムと同じく Bézout 行列が用いられている SNTLS アルゴリズム [14] を用いてグループ 1 からグループ 5 までの多項式の近似 GCD を求めた．

各グループにおいて近似 GCD の判定基準に適合した実験例の個数を表 3 に示す．列 “Bézout”，“Sylvester”，“SNTLS” はそれぞれ Bézout 行列を用いた GPGCD アルゴリズム，オリジナルの GPGCD アルゴリズム，SNTLS アルゴリズムを用いた実験に適合する実験例の個数であり，“All” は三つ全部のアルゴリズムを用いた実験に適合する例の個数である．三つ全部のアルゴリズムを用いた実験に適合する実験例に対し，計算時間，摂動のノルム，剰余のノルム，反復回数の平均値をそれぞれ表 4，5，6，7 に示す．表 4，5，6，7 の列 “Bézout”，“Sylvester”，“SNTLS” はそれぞれ Bézout 行列を用いた GPGCD アルゴリズム，オリジナルの GPGCD アルゴリズム，SNTLS アルゴリズムのデータを示す．

剰余のノルムの平均値については，算術平均値は極端な例に影響される場合があるので，剰余のノルムの桁の平均値を表す幾何平均値も示している（表 6 を参照）．

5.3 オリジナルの GPGCD アルゴリズムとの比較

表 3 は，与えられる多項式の次数が高い場合に，本アルゴリズムを用いた実験の中で近似 GCD の判定基準に適合する実験例の個数がオリジナルの GPGCD アルゴリズムのそれより多いことを示している．この二つのアルゴリズムを用いた実験両方に適合する実験例において，本アルゴリズムはオリジナルの GPGCD アルゴリズムより計算時間がやや大きい（表 4 を参照）が，与えられる多項式の次数が高い場合に，本アルゴリズムはオリジナルの GPGCD アルゴリズムより剰余のノルムが小さい（表 6 を参照）．

5.4 SNTLS アルゴリズムとの比較

表 3 は，グループ 1 からグループ 5 の実験例において，本アルゴリズムを用いた実験の中で近似 GCD の判定基準に適合する実験例の個数が SNTLS アルゴリズムのそれより少ないことを示している．

三つ全部のアルゴリズムを用いた実験に適合する実験例において，SNTLS アルゴリズムは本アルゴリズムおよびオリジナルの GPGCD アルゴリズムより剰余のノルムが小さい（表 6 を参照）が，SNTLS アルゴリズムによって計算された近似多項式の摂動は本アルゴリズムおよびオリジナルの GPGCD アルゴリズムによる摂動よりも大きい（表 5 を参照）．計算時間の面では，本アルゴリズムおよびオリジナルの GPGCD アルゴリズムは SNTLS アルゴリズムより効率的に計算できる（表 4 を参照）．

6 結論

我々は以前提案した Bézout の終結式行列を用いた GPGCD アルゴリズムを改良し，実験を行った．与えられる多項式が高次の場合，本アルゴリズムはオリジナルの GPGCD アルゴリズムより精度が良いことを示した．本アルゴリズムを同じく Bézout の終結式行列を用いたアルゴリズムである SNTLS アルゴリズムと比較した場合，本アルゴリズムはより効率的に近似 GCD を計算し，近似多項式の摂動もより小さい．

本アルゴリズムの効率化は今後の課題である．

表 1: 実験多項式の次数 (17)

グループ	$m = \deg(f)$	$n = \deg(g)$	$d = \deg(h)$
1	10	10	5
2	20	20	10
3	30	30	15
4	40	40	20
5	50	50	25
6	60	60	30
7	70	70	35
8	80	80	40
9	90	90	45
10	100	100	50

表 2: 実験環境

CPU	メモリー	OS
Intel(R) Core(TM) i5-6600 @ 3.30GHz	RAM 8.00GB	Windows 10

謝　辞

SNTLS アルゴリズムのソースコードをご提供くださった支麗紅先生，および有益なコメントをくださった関川浩先生，長坂耕作先生，讃岐勝先生に感謝します．本研究は，科研費 16K05035 の支援を受けた．

参 考 文 献

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

[2] P. Chin, R. M. Corless, and G. F. Corliss. Optimization strategies for the approximate GCD problem. In *Proceedings of the 1998 International Symposium on Symbolic and Algebraic Computation*, pages 228–235. ACM, 1998.

[3] E. W. Chionh, M. Zhang, and R. N. Goldman. Fast computation of the Bezout and Dixon resultant matrices. *J. Symbolic Comput.*, 33(1):13–29, 2002.

[4] R. M. Corless, P. M. Gianni, B. M. Trager, and S. M. Watt. The singular value decomposition for polynomial systems. In *Proceedings of the 1995 International Symposium on Symbolic and Algebraic Computation*, pages 195–207. ACM, 1995.

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

[6] G. M. Diaz-Toca and L. Gonzalez-Vega. Barnett's Theorems About the Greatest Common Divisor of Several Univariate Polynomials Through Bezout-like Matrices. *J. Symbolic Comput.*, 34(1):59–81, 2002.

表 3: 判定基準に適合する実験例の個数

グループ	実験例の個数			
	Bézout	Sylvester	SNTLS	All
1	97	99	98	97
2	82	91	85	81
3	60	82	77	60
4	46	79	66	46
5	47	39	66	36
6	30	22	—	20
7	27	18	—	18
8	18	14	—	12
9	14	9	—	6
10	14	9	—	9

表 4: 計算時間

グループ	平均時間（sec.）		
	Bézout	Sylvester	SNTLS
1	7.504×10^{-2}	5.961×10^{-2}	1.535×10^{-1}
2	7.958×10^{-1}	4.367×10^{-1}	1.181
3	1.653	8.089×10^{-1}	3.242
4	2.985	1.347	11.730
5	4.426	1.958	22.546
6	6.651	2.954	—
7	7.981	3.646	—
8	10.569	5.185	—
9	12.440	4.393	—
10	17.592	5.984	—

表 5: 摂動のノルム (18)

グループ	平均値		
	Bézout	Sylvester	SNTLS
1	6.346931×10^{-3}	6.346940×10^{-3}	6.346997×10^{-3}
2	7.035965×10^{-3}	7.036040×10^{-3}	7.111237×10^{-3}
3	6.795088×10^{-3}	6.795097×10^{-3}	8.742038×10^{-3}
4	6.706903×10^{-3}	6.706909×10^{-3}	6.706925×10^{-3}
5	6.824742×10^{-3}	6.824741×10^{-3}	6.824750×10^{-3}
6	6.715120×10^{-3}	6.715120×10^{-3}	—
7	7.118744×10^{-3}	7.118732×10^{-3}	—
8	6.840067×10^{-3}	6.840078×10^{-3}	—
9	7.256039×10^{-3}	7.256032×10^{-3}	—
10	7.124559×10^{-3}	7.124564×10^{-3}	—

表 6: 剰余のノルム (19)

グループ	算術平均値			幾何平均値		
	Bézout	Sylvester	SNTLS	Bézout	Sylvester	SNTLS
1	2.585×10^{-7}	5.591×10^{-11}	6.910×10^{-11}	1.563×10^{-11}	7.148×10^{-13}	6.939×10^{-13}
2	4.514×10^{-6}	3.002×10^{-10}	1.719×10^{-10}	4.026×10^{-10}	3.905×10^{-12}	3.522×10^{-12}
3	3.106×10^{-6}	2.594×10^{-10}	2.939×10^{-10}	6.765×10^{-9}	1.617×10^{-11}	1.506×10^{-11}
4	4.138×10^{-6}	1.816×10^{-10}	1.506×10^{-10}	2.962×10^{-8}	2.989×10^{-11}	2.587×10^{-11}
5	1.059×10^{-6}	1.693×10^{-5}	1.572×10^{-10}	2.054×10^{-7}	7.708×10^{-6}	7.428×10^{-11}
6	1.647×10^{-6}	2.923×10^{-5}	—	4.546×10^{-7}	1.460×10^{-5}	—
7	3.753×10^{-6}	2.906×10^{-5}	—	5.408×10^{-7}	1.762×10^{-5}	—
8	9.924×10^{-7}	3.267×10^{-5}	—	6.056×10^{-7}	2.383×10^{-5}	—
9	2.582×10^{-6}	3.970×10^{-5}	—	1.065×10^{-6}	3.184×10^{-5}	—
10	5.170×10^{-6}	6.185×10^{-5}	—	1.328×10^{-6}	5.090×10^{-5}	—

表 7: 反復回数

グループ	平均反復回数		
	Bézout	Sylvester	SNTLS
1	2.237	3.093	3.031
2	2.370	3.062	3.716
3	2.35	3.05	5.867
4	2.434	3.022	4.761
5	2.583	3.028	2.778
6	2.6	3.05	—
7	2.833	3	—
8	2.833	3	—
9	2.5	3	—
10	2.556	3	—

[7] I. Z. Emiris, A. Galligo, and H. Lombardi. Certified approximate univariate GCDs. *J. Pure Appl. Algebra*, 117/118:229–251, 1997.

[8] E. Kaltofen, Z. Yang, and L. Zhi. Approximate greatest common divisors of several polynomials with linearly constrained coefficients and singular polynomials. In *Proceedings of the 2006 International Symposium on Symbolic and Algebraic Computation*, pages 169–176. ACM, 2006.

[9] E. Kaltofen, Z. Yang, and L. Zhi. Structured low rank approximation of a Sylvester matrix. In D. Wang and L. Zhi, editors, *Symbolic-Numeric Computation*, Trends in Mathematics, pages 69–83. Birkhäuser, 2007.

[10] N. K. Karmarkar and Y. N. Lakshman. On approximate GCDs of univariate polynomials. *J. Symbolic Comput.*, 26(6):653–666, 1998.

[11] T. Sasaki and M.-T. Noda. Approximate square-free decomposition and root-finding of ill-conditioned algebraic equations. *J. Inform. Process.*, 12(2):159–168, 1989.

[12] A. Schönhage. Quasi-gcd computations. *J. Complexity*, 1(1):118–137, 1985.

[13] E. Schost and P.-J. Spaenlehauer. A quadratically convergent algorithm for structured low-rank approximation. *Found. Comput. Math.*, 16(2):457–492, 2016.

[14] D. Sun and L. Zhi. Structured Low Rank Approximation of a Bezout Matrix. *Math. Comput. Sci.*, 1(2):427–437, 2007.

[15] K. Tanabe. A geometric method in nonlinear programming. *J. Optimiz. Theory App.*, 30(2):181–210, 1980.

[16] A. Terui. An iterative method for calculating approximate GCD of univariate polynomials. In *Proceedings of the 2009 International Symposium on Symbolic and Algebraic Computation*, pages 351–358. ACM, 2009.

[17] 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 Process.*, 48(11):3042–3051, 2000.

[18] Z. Zeng. The numerical greatest common divisor of univariate polynomials. In L. Gurvits, P. Pébay, J. M. Rojas, and D. Thompson, editors, *Randomization, Relaxation, and Complexity in Polynomial Equation Solving*, volume 556 of *Contemporary Mathematics*, pages 187–217. AMS, 2011.

[19] L. Zhi. Displacement structure in computing approximate GCD of univariate polynomials. In *Computer mathematics: Proc. Six Asian Symposium on Computer Mathematics (ASCM 2003)*, volume 10 of *Lecture Notes Ser. Comput.*, pages 288–298. World Scientific, 2003.

[20] 池泊明，照井章. Bezout の終結式行列を用いた GPGCD 法による 1 変数多項式の近似 GCD の計算 (Computer Algebra: Theory and its Applications). 数理解析研究所講究録, 2104:8–13, 2019.