数理解析研究所講究録
第 1793 巻 2012 年 38-45

浮動小数係数多項式 Half-GCD 法の実装と応用

讃岐 勝

MASARU SANUKI

筑波大学医学医療系 & 附属病院総合臨床教育センター

FACULTY OF MEDICINE, UNIVERSITY OF TSUKUBA,

CENTER FOR MEDICAL EDUCATION AND TRAINING TSUKUBA UNIVERSITY HOSPITAL *

Abstract

本稿では，近似 GCD を求めるために改良した浮動小数係数多項式用の Half-GCD 法が整数係数多項式の場合に得られる情報を得られる改良を行う.

1 はじめに

近似代数の歴史の中で，近似 GCD 計算法に関する研究は最も盛んに行われているもののうちの一つである．その中で研究者は次をみたすような近似 GCD を求めるべき試行錯誤している．$\boxed{1}$ 高い次数の多項式の近似 GCD を高速に求める．$\boxed{2}$ 摂動を小さくする．$\boxed{3}$ ニーズにあった算法を開発する．当然$\boxed{3}$を目標に研究するべきであるが，多くの研究ははじめの 2 つのうち 1 つのみをターゲットに研究される傾向がある．特に 1 変数 GCD に関してそのような傾向がある．下に列挙した.

- 効率性を求める.
  - 互除法・PRS(多項式剰余列) 法 [SN89, SS07]：微小主項の除算による桁落ち (自己簡約) を避けるために軸交換をする必要がある．ただし，厳密に軸交換をすると算法として QRGCD 法に近づくため，効率が悪くなる.
  - 高速算法の利用 [Zhi03, BB07]：displacement を利用し，多項式の次数 m に対して $O(m^2)$ の計算量で GCD が計算できる.
- 摂動を抑える
  - QR 法に基づく方法 [CWZ04]：
  - SVD(特異値分解) に基づく方法 [Zen04]：GCD の次数を与える必要がある.
  - 最適化法・STNL 法 [KYZ05, Terui09]：GCD の次数を与える必要がある.
  - その他 [Pan01]

本稿では，1 つ目をターゲットにして算法の開発を行い，そのような考えのもと議論を行う．そのため，摂動に関する議論は細かく行わない.

筆者は Half-GCD 法を浮動小数多項式に対しても適用できるようにし，整数多項式に並ぶまでに浮動小数 GCD(近似 GCD) 計算を効率化したが (1000 次の多項式の GCD の計算に成功)，精度に関しては何も検討をしておらず考慮すべき点が多々ある [Sanuki11]．本稿では，その点に関して触れる.

*sanuki@md.tsukuba.ac.jp

1.1 記号・定義

本稿では次の記号を使用する．浮動小数 $\mathbb{F}$ を係数にする 1 変数多項式 $f(x), g(x) \in \mathbb{F}[x]$ を次で表現する．

$$f(x) = f_m x^m + f_{m-1}x^{m-1} + \cdots + f_0, \quad g(x) = g_n x^n + g_{n-1}x^{n-1} + \cdots + g_0.(d = m - n \geq 0) \tag{1}$$

$\deg(f), \mathrm{lc}(f)$ は多項式 f の次数，主係数をそれぞれ表す．$\gcd(f, g)$ は多項式 f と g の GCD(最大公約子) を表す．

2 Half-GCD 法

Half-GCD 法は互除法を基に開発された GCD 算法である (整数多項式版 [Moenck73])．浮動小数係数多項式の互除法は不安定であることが知られているため，そのまま適応することはできない．この章では，half-GCD 法が適応できるような消去法を提案し，Half-GCD 法の説明をする．

2.1 両端消去

多項式 $f(x)$ と $g(x)$ に対する両端消去を次のように定義する．

定義 1 (両端消去 [Sanuki11])
$f(x)$ と $g(x)$ による両端消去 *(double-side reduction)* とは次で定義する演算である．

$$\begin{pmatrix} f'(x) \\ g'(x) \end{pmatrix} = \begin{pmatrix} 1 & -\gamma x^d \\ -w/x & 1/x \end{pmatrix} \begin{pmatrix} f(x) \\ g(x) \end{pmatrix}. \tag{2}$$

ただし，$\gamma = f_m/g_n$，$w = g_0/f_0$ $(f_0 \neq 0)$ である．

このとき，次数について次をみたす．

$$\deg(f'), \deg(g') < \max\{\deg(f), \deg(g)\}. \tag{3}$$

この条件は必ず次数が落ちることを意味しており安定化剰余列法 [SS07] の問題点を解決する．この両端消去を利用した互除法を **dual-PRS 法**という．

命題 2 (Dual-PRS 法 [Sanuki11])
Dual-PRS 法は停止する．停止したときに得られた多項式は *GCD* でなく，*GCD* を因子に持つ多項式である (例 1 を参照)． ∎

例 1 (GCD が得られない場合 [Sanuki11])
$f_1 = x^5 + 1$ と $g_1 = x^3 + 1$ による両端消去によって次が得られる．

$$\begin{pmatrix} f_1' \\ g_1' \end{pmatrix} = \begin{pmatrix} 1 & -x^2 \\ -1/x & 1/x \end{pmatrix} \begin{pmatrix} f_1 \\ g_1 \end{pmatrix} = \begin{pmatrix} x^2 - 1 \\ -x^2(x^2 - 1) \end{pmatrix}.$$

$\gcd(f_1, g_1) = x + 1$ なので，結果として GCD を因子にもつ多項式が得られた．各多項式の余因子が対称式であり，両端消去は頭項と定数項を同時に消去するため，結果として余分な項が掛かった共通因子を持つ多項式のペアが得られた．

別の例として，互いに素な $f_2 = x^5 + 1$ と $g_2 = x^3 - 1$ に両端消去を適応すると，

$$\begin{pmatrix} f_2' \\ g_2' \end{pmatrix} = \begin{pmatrix} 1 & -x^2 \\ -1/x & 1/x \end{pmatrix} \begin{pmatrix} f_2 \\ g_2 \end{pmatrix} = \begin{pmatrix} x^2+1 \\ -x^2(x^2+1) \end{pmatrix}.$$

これについても，上に似た理由から因子を持つ．ここでは述べない (詳しくは [Sanuki11] を参照). ∎

Dual-PRS 法によって得れらた GCD 候補 $\tilde{c}(x)$ が GCD があるか確認するためには除算 (試し割り) によって余りが小さいか確認する．また $\tilde{c}(x)$ から入力多項式の GCD を得るために次を行う．

- $\gcd(f(x), \tilde{c}(x))$ または $\gcd(g(x), \tilde{c}(x))$ を計算する．このときに得られた多項式の次数は $\deg(\tilde{c})$ より小さい．その後，試し割りによって確認を行う．
- オプション：$f(x)$ および $g(x)$ の $\tilde{c}(x)$ の剰余同士による GCD を計算する (計算の効率化).

得られる GCD の候補の次数は，常に減少するので GCD は得られ計算は必ず停止する．何度も GCD を計算するので，計算量について見積もる必要がある．

補題 3 (Dual-PRS 法の計算量)
上の工夫による *Dual-PRS* 法を用いた場合でも，*GCD* を計算するために必要となる計算量は $O(m^2)$ である．

証明 与えられた多項式より GCD の候補が $O(m^2)$ の計算量で計算できる．チェックのために行う除算の計算量は高々$O(m)$ である．常に次数が落ちるので，最悪 GCD が一回の両端消去によって得られる場合，すなわち次数が 1 程度しか落ちない場合が考えられるが，それは GCD が得られた時にのみ起こり得る．そのため，GCD が得られるまでの計算量は高々$O(m^2)$ である． ∎

2.2 Half-GCD 法

両端消去を利用した浮動小数係数 half-GCD 法について説明する．算法は整数係数 half-GCD 法を改良して構成されている．

2.2.1 整数多項式 Half-GCD 法

Moenck によって，整数多項式の half-GCD 法が提案された．

アルゴリズム 1 (整数多項式 Half-GCD 法 [Moenck73])

Input: f_h, g_h
Steps: *if* $\deg(f_h) = 1$ *or* $\deg(g_h) = 1$ *then return* 商行列
else
  split f_h *as* $f_h = f_{h,1}x^k + f_{h,0}$, *and*
  split g_h *as* $g_h = g_{h,1}x^k + g_{h,0}$, *where* $k = \lceil \deg(f_h)/2 \rceil$.
  $R_1 = \mathrm{HGCD}(f_{h,1}, g_{h,1})$;
  $(q_1, q_2)^T = R_1(f_h, g_h)$
  $R_2 = \mathrm{HGCD}(q_1, q_2)$;
  return $R_2 \cdot R_1$;
end;;

アルゴリズム 1 にある商行列は次の 2×2 行列である.

$$\begin{pmatrix} p_i \\ p_{i+1} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & -\mathrm{quo}(p_{i-1}, p_i) \end{pmatrix} \begin{pmatrix} p_{i-1} \\ p_i \end{pmatrix}, \tag{4}$$

ただし, $\mathrm{quo}(p_{i-1}, p_i)$ は p_{i-1} を p_i で割った時の商である. 商の計算は, 多項式の係数全てを必要とせず多項式の第 1・第 2 主係数のみから可能である. そのため分割統治法 (divide-and-conquer algorithm) を適応できる. そのため, 計算量は次の命題のように見積もられる.

命題 4 (計算量 [Moenck73])
計算量は $\log m \cdot M(m)$ である. ただし, $M(m)$ は次数 m の多項式乗算の計算量である. ∎

上の補題より, half-GCD 法の計算量は乗算に依存する. FFT を利用した乗算を用いたときの計算量は $O(m \cdot \log^2 m)$ と $O(m^2)$ 未満になる.

2.2.2 浮動小数係数 Half-GCD 法

整数多項式の half-GCD 法を基に浮動小数多項式に対して算法を拡張した [Sanuki11]. 問題となるのは, 1) 商行列の構成法, 2) 分割統治法の適応方法である. 消去法として両端消去を利用するので, 商行列は式 (2) で表される行列になる. また, 商行列は多項式の主項および定数項部分のみから構成するため, 多項式の分割を次数の高い部分だけではなく次数の低い部分に対して行う必要がある. そのため, 次のようにアルゴリズムを構成した. アルゴリズム内にある f_h, f_ℓ, g_h, g_ℓ は, それぞれ $f = f_h x^k + f_\ell$ および $g = g_h x^k + g_\ell$ を表す. ただし, $k = \lceil \deg(f) \rceil$ かつ $\deg(f_\ell), \deg(g_\ell) < k$ をみたすように多項式を分割する.

アルゴリズム 2 (浮動小数係数 Half-GCD 法)

Input: *four polynomials* f_h, f_ℓ, g_h, g_ℓ
with $\deg(f_h) = \deg(g_h) \le \deg(f_\ell) = \deg(g_\ell)$.
Steps: *if* $\deg(f_h) = 0$ *then return* 商行列
else
  split f_h *and* f_ℓ *as* $f_h = f_{h,1}x^k + f_{h,0}$, $f_\ell = f_{\ell,1}x^k + f_{\ell,0}$,
  and split g_h *and* g_ℓ $g_h = g_{h,1}x^k + g_{h,0}$, $g_\ell = g_{\ell,1}x^k + g_{\ell,0}$, *where* $k = \lceil \deg(f_h)/2 \rceil$.
  $R_1 = \mathrm{HGCD}(f_{h,1}, g_{h,1}, f_{\ell,0}, g_{\ell,0})$;
  $(q_1, q_2)^T = R_1(f_h, g_h)$ *and* $(r_1, r_2)^T = R_1(f_{\ell,0}, g_{\ell,0})$
  $R_2 = \mathrm{HGCD}(q_1, q_2, r_1, r_2)$;
  return $R_2 \cdot R_1$;
end;;

命題 5 (計算量 [Sanuki11])
計算量は $\log m \cdot M(m)$ である. ただし, $M(m)$ は次数 m の多項式乗算の計算量である. ∎

注意 1 (実装に関する工夫)
巨大次数の多項式の *GCD* を計算する場合, 計算を進めていく上で丸め誤差による精度低下は避けられない. そのため入力より大きい精度で計算をする必要がある.

2.3 Half-GCD 法による拡張互除法

GCD 計算だけではなく，次の関係式 (拡張互除法) をみたす $a(x)$ と $b(x)$ も様々なところで利用される．

$$a(x)f(x) + b(x)g(x) = \gcd(f, g). \tag{5}$$

ただし，$\deg(a) < \deg(g) - \deg(c)$ および $\deg(b) < \deg(f) - \deg(c)$ をみたす．整数多項式の half-GCD 法では GCD を計算すると同時に上をみたす多項式を得ることができるが，浮動小数多項式の half-GCD 法では GCD を計算しても，上を関係をみたす多項式は同時に得られない (一般に得られるのは有理式である)．以下では，浮動小数多項式の half-GCD 法で得られる関係式から式 (5) をみたす多項式を高速に計算できないか考察する．

2.3.1 式の次数

まず，浮動小数多項式の half-GCD 法で得られる式の次数を評価する．簡単のため，$\deg(f) = \deg(g)$ と仮定する (算法の都合上，このように仮定しても問題ない)．$p_1 = f$ および $p_2 = g$ とおくとき，この 2 つの多項式による両端消去は

$$\begin{pmatrix} p_3 \\ p_4 \end{pmatrix} = \frac{1}{x}\begin{pmatrix} x & -\gamma_1 x \\ -w_1 & 1 \end{pmatrix}\begin{pmatrix} p_1 \\ p_2 \end{pmatrix}, \tag{6}$$

で表され ($\gamma_1 = \mathrm{lc}(p_1)/\mathrm{lc}(p_2), w_1 = \mathrm{const}(p_2)/\mathrm{const}(p_1)$)，$p_3$ と p_4 による両端消去は，

$$\begin{aligned}\begin{pmatrix} p_5 \\ p_6 \end{pmatrix} &= \frac{1}{x}\begin{pmatrix} x & -\gamma_2 x \\ -w_2 & 1 \end{pmatrix}\begin{pmatrix} p_3 \\ p_4 \end{pmatrix} \\ &= \frac{1}{x^2}\begin{pmatrix} x & -\gamma_2 x \\ -w_2 & 1 \end{pmatrix}\begin{pmatrix} x & -\gamma_1 x \\ -w_1 & 1 \end{pmatrix}\begin{pmatrix} p_1 \\ p_2 \end{pmatrix} \\ &= \frac{1}{x^2}\begin{pmatrix} x^2 + \gamma_2 w_1 x & -\gamma_2 x^2 - \gamma_2 x \\ -w_2 x - w_1 & w_2\gamma_1 x + 1 \end{pmatrix}\begin{pmatrix} p_1 \\ p_2 \end{pmatrix},\end{aligned} \tag{7}$$

と表される ($\gamma_2 = \mathrm{lc}(p_3)/\mathrm{lc}(p_4), w_2 = \mathrm{const}(p_4)/\mathrm{const}(p_3)$)．構成される余因子を眺めると，1 回の両端消去によって分母の次数は 1 増え，分子の次数も 1 増えている．一般に次が成立する．

命題 6
上の仮定のとき，p_{2i-1} および p_{2i} と p_1 および p_2 に関して，

$$\begin{pmatrix} p_{2i-1} \\ p_{2i} \end{pmatrix} = \frac{1}{x^{i-1}}\begin{pmatrix} \tilde{a}_{2i-1} & \tilde{b}_{2i-1} \\ \tilde{a}_{2i} & \tilde{b}_{2i} \end{pmatrix}\begin{pmatrix} p_1 \\ p_2 \end{pmatrix} \tag{8}$$

をみたす有理式 $\frac{\tilde{a}_{2i-j}}{x^{i-1}}, \frac{\tilde{b}_{2i-j}}{x^{i-1}} \in \mathbb{F}[x, \frac{1}{x}]$ について次がいえる ($i \geq 2, j = 1, 0$)．

1. $\tilde{a}_{2i-j}, \tilde{b}_{2i-j} \in \mathbb{F}[x]$.
2. $\deg(\tilde{a}_{2i-j}), \deg(\tilde{b}_{2i-j}) \leq (i-1) + (j-1) = i + j - 2$.

証明 帰納法で証明する．$i=2$ のとき，式 (7) より成り立つ．$i=k$ のとき，成り立つと仮定するとき，

$$\begin{aligned}\begin{pmatrix} p_{2k+1} \\ p_{2k+2} \end{pmatrix} &= \frac{1}{x}\begin{pmatrix} x & -\gamma_k x \\ -w_k & 1 \end{pmatrix}\begin{pmatrix} p_{2k-1} \\ p_{2k} \end{pmatrix} = \frac{1}{x}\begin{pmatrix} x & -\gamma_k x \\ -w_k & 1 \end{pmatrix}\frac{1}{x^{k-1}}\begin{pmatrix} \tilde{a}_{2k-1} & \tilde{b}_{2k-1} \\ \tilde{a}_{2k} & \tilde{b}_{2k} \end{pmatrix}\begin{pmatrix} p_1 \\ p_2 \end{pmatrix} \\ &= \frac{1}{x^k}\begin{pmatrix} x\tilde{a}_{2k-1} - \gamma_k x\tilde{a}_{2k} & x\tilde{b}_{2k-1} - \gamma_k x\tilde{b}_{2k} \\ -w_k\tilde{a}_{2k-1} + \tilde{a}_{2k} & -w_k\tilde{b}_{2k-1} + \tilde{b}_{2k} \end{pmatrix} = \frac{1}{x^k}\begin{pmatrix} \tilde{a}_{2k+1} & \tilde{b}_{2k+1} \\ \tilde{a}_{2k+2} & \tilde{b}_{2k+2} \end{pmatrix}.\end{aligned}$$

このとき，$\tilde{a}_{2k+2-j}, \tilde{b}_{2k+2-j} \in \mathbb{F}[x]]$ であり，$\tilde{a}_{2k-1} > \tilde{a}_{2k}$ および $\tilde{b}_{2k-1} > \tilde{b}_{2k}$ なので，$\deg(\tilde{a}_{2k+2-j}) = 1 + \deg(\tilde{a}_{2k-j})$ および $\deg(\tilde{b}_{2k+2-j}) = 1 + \deg(\tilde{b}_{2k-j})$ をみたす．ゆえに命題をみたす． ∎

2.3.2 構成法

Half-GCD 法によって，次をみたす多項式 $\tilde{a}, \tilde{b}$ がわかっている．

$$\frac{\tilde{a}(x)}{x^k}f(x) + \frac{\tilde{b}(x)}{x^k}g(x) = c, \tag{9}$$

ただし，$\deg(c) = k$ かつ $\deg(\tilde{a}), \deg(\tilde{b}) \leq m - k$ である．これから，式 (5) をみたす関係式は次のように構成する．

- $\frac{\tilde{a}(x)}{x^k}$ を次数の低い項から $g(x)$ の定数部によって消去を行う．
- $\frac{\tilde{b}(x)}{x^k}$ を次数の低い項から $f(x)$ の定数部によって消去を行う．

1 回の消去によって，消去された有理式の最低次数は 1 より大きくなる．この操作を繰り返すことによって次数が 0 未満の項を消すことが可能であり，このとき計算を停止する．このとき，得られるものは目的の項である．

命題 7 (拡張互除法の計算量)
先に述べた方法によって拡張互除法を計算するとき，計算量は $O(m)$ 以下である．

証明 全体の計算量は 1 回の除算分だけなので高々$O(m)$ である． ∎

注意 2 (誤差について)
1 回の除算しか行わないので，自己簡約による誤差は発生しない．

3 まとめ

近似 GCD 計算では，効率性を求めるために拡張互除法まで計算しないことが多い．ただ必要とする応用は多く，近似 GCD 計算の高速化と拡張互除法の計算が高速に計算できることが必要とされる．本稿ではそれを実現した．

今後の課題は，近似 GCD を用いた工学的応用に適応し効率について議論することである．

参 考 文 献

[AHU74] A. V. Aho, J. E. Hopcroft and J. D. Ullman. The design and analysis of computer algorithms. Addison-Wesley, 1974.

[BB07] 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.

[BL98] 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.

[CGTW95] 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.

[CWZ04] 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.

[EGL97] I. Emiris, A. Galligo and H. Lombardi. Certified approximate univariate GCDs. *J. Pure and Applied Alge.*, **117&118** (1997), 229–251.

[KS97] F. Kako and T. Sasaki. Proposal of "effective floating-point number" for approximate algebraic computation. *Preprint of Tsukuba Univ.*, 1997.

[KYZ05] E. Kaltofen, Z. Yang and L. Zhi. Structured low rank approximation of a Sylvester matrix. *International Workshop on SNC'05*, D. Wang & L. Zhi (Eds.), 2005, 188–201; full paper appear in *Symbolic-Numeric Computation (Trends in Mathematics)*, D. Wang & L. Zhi (Eds.), Birkhäuser Verlag, 2007, 69–83.

[LYZ10] Z. Li, Z. Yang, and L. Zhi. Blind image deconvolution via fast approximate GCD *Proc. of ISSAC'10*, ACM Press, 2010, 155–162.

[Moenck73] R. T. Moenck. Fast computation of GCDs. *Proc. 5th ACM Symp. Thory of Comput.*, 1973, 142–151.

[MY73] J. Moses and D. Y. Y. Yun. *The EZGCD algorithm.* Proc. ACM National Conference, ACM, 1973, 159–166.

[OST97] 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.

[Pan01] V. Pan. Univariate polynomials: nearly optimal algorithms for factorization and rootfinding. *Proc. of ISSAC'01*, ACM Press, 2001, 253–267.

[PW02] V. Pan and X. Yang. *Acceleration of Euclidean algorithm and extensions.* Proc. of ISSAC'02, ACM Press, 2002, 207–213.

[Sanuki05] M. Sanuki. Computing approximate GCD of multivariate polynomials (Extended abstract), Proc of SNC'05 . D. Wang & L. Zhi (Eds.), 2005, 308–314; full paper appear in Symbolic-Numeric Computation (Trends in Mathematics), D. Wang & L. Zhi (Eds.), Birkhäuser Verlag, 2007, 55–68.

[Sanuki09] M. Sanuki. Computing multivariate approximate GCD based on Barnett's theorem. *Proc. of SNC'09*, ACM Press, 2009, 149-157.

[Sanuki11] M. Sanuki. Challenge to fast and stable computation of approximate univariate GCD, based on displacement structures. *Proc. of SNC'11*, ACM Press, 2011, 178-186.

[SN89] 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.

[SF84] T. Sasaki and A. Furukawa. Secondary polynomial remainder sequence and an extension of sub-resultant theory. *J. Inform. Proces.*, **7** (1984), 175–184.

[SS89] T. Suzuki and T. Sasaki. On Sturm sequence with floating-point number coefficients. *RIMS Kokyuroku*, **685** (1989), 1–14.

[SS07] M. Sanuki and T. Sasaki. Computing approximate GCD in ill-conditioned cases. *Proc. of SNC'07*, ACM Press, 2007, 170–179.

[Terui09] A. Terui. An iterative method for calculating approximate GCD of univariate polynomials. *Proc. of ISSAC'09*, ACM Press, 2009, 351–358.

[Wang80] P. S. Wang. *The EEZ-GCD algorithm.* SIGSAM Bulletin **14** (1980), 50–60.

[Zen04] Z. Zeng. The approximate GCD of inexact polynomials part I: a univariate algorithm. *Preprint*, 2004.

[Zhi03] L. Zhi. Displacement structure in computing the approximate GCD of univariate polynomials. *Proc. of ASCM2003 (Asian Symposium on Computer Mathematics)*, World Scientific, 2003, 288–298.

[ZD04] Z. Zeng and B. H. Dayton. *The Approximate GCD of inexact polynomials part II: a multivariate algorithm.* Proc. of ISSAC'04, ACM, 2004, 320–327.

[ZN00] L. Zhi and M-T. Noda. *Approximate GCD of multivariate polynomials.* Proc. of ASCM2000, World Scientific, 2000, 9–18.

[ZMF00] 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.