

浮動小数 Gröbner 基底の実際的な計算法

佐々木 建昭 (Tateaki Sasaki)

筑波大学 数学系

INSTITUTE OF MATHEMATICS, UNIVERSITY OF TSUKUBA *

Abstract

浮動小数 Gröbner 基底の計算は非常に不安定で、その安定化は世界的課題である。前 2 論文で筆者らは、主要項のキャンセルが頻繁に起きて計算を不安定にすることを明らかにし、正確な項キャンセルを無害にする方法を提示した。筆者らは不正確な項キャンセル量の評価法も提示したが、その方法は非常に複雑で、理論的・実的に不完全であった。本稿では、不正確な項キャンセル量を簡単かつ正確に評価する方法を提示する。

1 はじめに

浮動小数を用いて計算した多項式イデアルの Gröbner 基底を浮動小数 Gröbner 基底という。浮動小数 Gröbner 基底には 2 種類ある。第一種は、入力多項式の係数は正確だが、何らかの理由で浮動小数を用いる場合。第二種は、入力多項式の係数が不正確なため、係数を浮動小数で表現せざるを得ない場合である。第一種では、誤差が増大すれば計算精度 (precision) を上げて再計算すればよいが、第二種では、計算結果の精度 (accuracy) を保って計算する方法を工夫することが不可欠である。本稿では第二種を扱う。

第一種の浮動小数 Gröbner 基底は論文 [15, 16, 17] で研究され、第二種の浮動小数 Gröbner 基底は論文 [18, 5, 3, 22, 21, 23, 7, 4, 20, 1] などで研究された。これらの研究にも拘わらず、第二種に対する計算法は近年まで深刻な問題だった：普通に計算すると誤差が非常に増大したり、余計な項が現れるのである。

浮動小数 Gröbner 基底の計算はなぜ不安定なのだろうか？原因として、Shirayanagi は**完全誤差項**を指摘し [15]、Sasaki-Kako は**主要項のキャンセル**を指摘した [12]。浮動小数の計算では、数学的には等価な数の減算は一般に 0 にはならず、誤差だらけの数になるが、それが完全誤差項である。論文 [12] で Sasaki-Kako は、Buchberger 算法において、主要項全体が互いに正確にキャンセルする一つのメカニズムを提示した：微小主項あるいは巨大主項の多項式が現れると、後の計算で主要項どうしが正確にキャンセルすることが多々あり、その場合、大きな桁落ち誤差を引き起こすのである。完全誤差項は、数係数を区関数 [15] あるいは“有効浮動小数” [6] で表現することにより、簡単かつ効率的に除去できる。問題は主要項キャンセルによる巨大な桁落ち誤差である。

Sasaki-Kako は主要項キャンセルを**正確な項キャンセル**と**不正確な項キャンセル**に分類した。論文 [13] において Sasaki-Kako は、計算精度を上げることにより正確な項キャンセルは無害化できることを指摘した。これが**高精度法**であり、第 2 章で詳しく説明する。残念ながら、高精度法は不正確な項キャンセルには全く無力である。Sasaki-Kako は同論文で、各係数に生じた不正確なキャンセル量を推定する方法も提案した。その方法は、多項式の減算で一定量以上の正確な項キャンセルが生じると推定された場合、キャンセルする主要項をそれぞれの多項式から除いた後に減算を実行する。それでもなお起きるであろう項キャンセルを不正確な項キャンセルとみなすものである。この方法は複雑な上に、正確な項キャンセルによる誤差を完全

*sasaki@math.tsukuba.ac.jp

には除けないので不十分である。さらに、正確な項キャンセルのメカニズムの多くが未知なので、理論的に不完全である。不正確な項キャンセルは計算した Gröbner 基底の精度に直結するので、その値を知ることには非常に重要である。現在の課題は、不正確な項キャンセル量を正確に知る方法の開発と、正確・不正確な項キャンセル量をいかに低減させるか、ということである。

本稿では、計算結果のみならず計算中に現れる多項式の各係数に生じる不正確な項キャンセル量を簡単かつ正確に知る方法を提案する。提案手法は、入力多項式の各係数の誤差部に“マーク”を入れ、高精度法と併用するものである。なお、正確・不正確な項キャンセル量を低減させる方法については、昨年の数理論研究集会で報告したので参照されたい [14]。

2 正確な項キャンセルと高精度法

$\mathbb{C}[x, y, \dots, z]$ の多項式を F, G 等と表すが、係数は浮動小数である。 F のノルムを $\|F\|$ と表す：本稿では無限大ノルム (数係数の絶対値の中で最大のもの) を用いる。項順序 \succ の下で、 F の主項 (単項式)、主係数、残余をそれぞれ $lt(F), lc(F), rt(F)$ と表す： $F = lt(F) + rt(F), lt(F) \succ rt(F)$ 。係数のない項をべき積といい、lpp と略記する： $lt(F) = lc(F) lpp(F)$ 。S 多項式と主項簡約をそれぞれ $Spol(F, G), Lred(F, G)$ と表す； $Lred(F, G)$ は $F \xrightarrow{G}$ と表し、 $F \xrightarrow{G}$ は主項が G -既約になるまで簡約することを意味する。

本稿では多項式の簡約は主項簡約に限定する、すなわち、S 多項式生成と主項簡約のみで Gröbner 基底を計算する。この制約は理論解析にとって本質的に重要である (尤も、理論解析は本稿には出てこないが)。簡約 Gröbner 基底が欲しいときは、非簡約 Gröbner 基底を計算したあとで簡約すればよい。

正確な項キャンセルがどのようなメカニズムで起きようとも、系の初期精度は正確な項キャンセルによる誤差から次の方法で保護できる [13]。

高精度法 与えられた基底の各多項式の初期精度を ϵ_{init} ($\epsilon_{init} \ll 1$) とし、浮動小数 Gröbner 基底計算で係数に生じるであろう正確なキャンセル量の上界を C_{exact} とする。まず、計算精度 ϵ_{cal} を $\epsilon_{init}/C_{exact}$ に設定し、入力多項式の全ての数係数を精度 ϵ_{cal} に変換 (高精度化) する。ついで、Buchberger 算法で Gröbner 基底を計算し、最後に、数係数を元の精度の浮動小数に戻す。

もちろん、 C_{exact} は事前には未知なので、実際には $\epsilon_{cal} = 10^{-30} \Rightarrow 10^{-60} \Rightarrow 10^{-90} \Rightarrow \dots$ のように精度を上げつつ試行することが必要である。

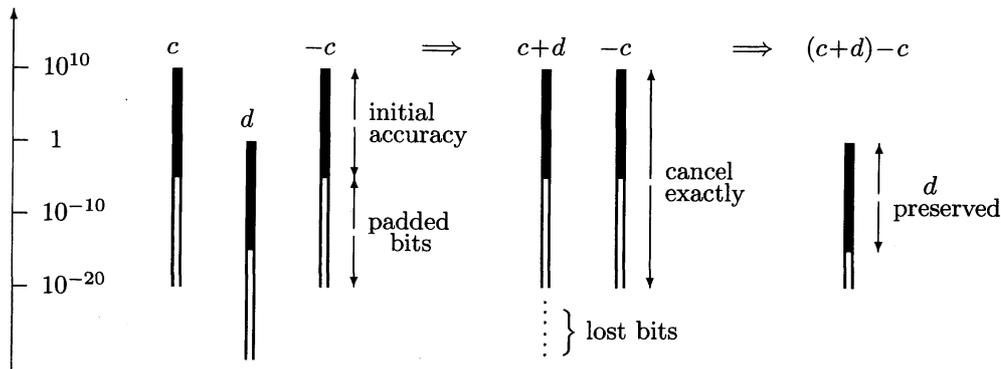


図 1: 高精度法での $c+d-c, |c| \gg |d|$ の計算

高精度法がなぜ初期精度を正確キャンセルから保護するか、図で説明する。たとえば、浮動小数で $c+d-c$ の計算を考える。ここで、 $c \simeq 10^{10}, d \simeq 1$ であり、 c と d は初期 (相対) 精度が 10^{-15} とし、高精度法では

c と d は 10 進 30 桁の浮動小数に変換されるとする。この場合、下 15 桁は完全に意味のない数字が並ぶが、計算機は 30 桁の数値全体をあたかも相対精度 10^{-30} の数のように正確に扱う。さらに、計算機は最初に和 $c+d$ を計算し、つぎに 差 $(c+d) - c$ を計算するとする：図 1 参照。さて、 c と d が加えられたとき、 $c+d$ は形式的には誤差 10^{-20} の数になる（もちろん、下 15 桁は意味のない数字である）。つぎに、 $c+d$ から c を引くと、結果は大きさが d で形式的には誤差 10^{-20} の数になる。この計算で、数値 $c+d$ に入っている c は誤差部分も含めて $-c$ により完全に打ち消される。結果として、 d の初期精度が保たれる。

一般には c と d は加減乗除算を何度も経た数値であるが、正確な項キャンセルとは係数がパラメータでも成立するものなので、 c と d は Gröbner 基底計算のどのステップで現れる数でもよい。

3 不正確な項キャンセルについて

高精度法は不正確な項キャンセルには全く無力である。まず、不正確キャンセルの例を示す。

例 1 (不正確な項キャンセル) P_1, P_2, P_3 を下記の多項式とする。

$$\begin{cases} P_1 = 57/56 x^2 y + 68/67 x z^2 - 79/78 x y + 89/88 x \\ P_2 = xy z^3 - xy^2 z + x y z \\ P_3 = 56/57 x y^2 - 67/68 y z^2 + 78/79 y^2 - 88/89 y \end{cases} \quad (3.1)$$

まず、上記多項式の有理係数を倍精度の浮動小数に変換して各係数に相対精度 2×10^{-16} の誤差を導入し、それを初期基底とする。次に、項順序として全次数順序を選び、精度 10^{-30} の浮動小数で Gröbner 基底を計算すると、次の系が得られる。下線部は、有理数で計算した基底と比較して合っている数字である。

$$\begin{cases} P_1 \text{ and } P_3 \text{ are unchanged, } P_2 \rightarrow P'_2 \text{ (no cancellation),} \\ P_6 = y^2 z^2 - \underline{2.995436947732552644538319700370} xy^2 \\ \quad - \underline{1.0020782165123748257674951096740} y^3 \\ \quad + \underline{1.9983254691737245140192885621560} xy \\ \quad + \underline{1.003521717256414556431326513500} y^2, \\ P_7 = x z^2 - \underline{1.764316342370426661429391997320e-3} y z^2 \\ \quad - \underline{9.947232450186805419457332443380e-1} xy \\ \quad + \underline{1.7679829737261936385647927531480e-3} y^2 + \dots \end{cases}$$

高精度計算でも多くの係数に約 10^5 の誤差が生じていることが分かる。この誤差は不正確な項キャンセルによるものであることが計算をトレースしてみると分かる：まず、 $P_4 := \text{Spol}(P_1, P_3)$ が計算され、ついで P_2 が P_4 で簡約されるが、その際に次の近似線形従属関係に遭遇する。

$$\|56/57 y z P_1 - 57/56 x z P_3 - 2P_2\| = 0.000041.$$

この近似従属関係が P_6 と P_7 のいくつかの係数で約 10^5 のキャンセルを引き起こすのである。◇

初期基底を $\{F_1, \dots, F_r\}$ とする。例 1 は、初期基底あるいは中間基底の多項式の間近似線形従属関係が存在すれば不正確な項キャンセルが引き起こされる、と示唆する。一方、Buchberger 算法で現れる任意の多項式 P は、初期基底の要素により $P = a_1 F_1 + \dots + a_r F_r$, $a_i \in \mathbb{C}[x, y, \dots, z]$ ($i=1, \dots, r$) と表すことができる。 (a_1, \dots, a_r) は P のシジジーと呼ばれる。もしも $\|P\| \ll \max\{\|a_1 F_1\|, \dots, \|a_r F_r\|\}$ なら、 F_1, \dots, F_r の間に線形従属関係があることになる。

P の計算で正確な主要項キャンセルが起きると、 P のシジジー計算でもほぼ同量の主要項キャンセルが起きることを論文 [12] は証明したが、分り難い。筆者は最近、Buchberger 算法に対する部分終結式もどきの理論を作った [11]。Buchberger 算法で計算される任意の多項式は、部分終結式をかなり拡張した行列式 (要素は初期多項式の係数) で表される、というものである。多項式剰余列 $(P_1, P_2, \dots, P_i, \dots)$ の要素 P_i を部分終結式で表すとき、 $P_i = a_i P_1 + b_i P_2$ を満たす a_i と b_i は部分終結式の最後の列を x のべきで置き換えた行列式で表される。同様に、シジジーも部分終結式もどきの最後の列をべき積で置き換えて表すことができる。このことから直接的に、シジジー計算でも P におけるとほぼ同じ主要項キャンセルが起きることを示せる。よって、われわれは不正確な項キャンセルを次のように特徴づける。

不正確な項キャンセルの特徴づけ P と (a_1, \dots, a_r) は上記で定義されたものとする。このとき、次式で定める C_{inxt} を P に起きる不正確な項キャンセルの量であるとする。

$$\frac{\max\{\|a_1 F_1\|, \dots, \|a_r F_r\|\}}{\|P\|} = C_{\text{inxt}} \gg 1. \quad (3.2)$$

4 新しい方法：高精度マーキング法

本章では、浮動小数 Gröbner 基底に対する実際的で安定した計算法を提案する。まず、算法開発の前提となる状況と、直面している課題とをまとめておく。

- 入力多項式の各係数は誤差を持っており、われわれは各係数の相対精度を知っているとする。入力系の係数の相対精度を $\varepsilon_{\text{init}}$ とする： $\varepsilon_{\text{init}}$ としては相対精度の平均値 $\text{avr}_i\{|\text{error}(c_i)|/|\text{value}(c_i)|\}$ が妥当であろうが、最悪の相対精度を使うのも一手である。もしも基底計算の途中で、ある多項式に巨大な不正確キャンセルが起き初期精度がほとんど失われたなら、その多項式は破棄しなければならない。特に、主係数の精度がほとんど失われたなら、主項を破棄する必要がある。したがって、計算途中の多項式も含め全ての多項式の全ての数係数の桁落ち量を知る必要がある。
- 高精度法は正確な項キャンセルの量を減らす訳ではなく、有効浮動小数で検知できるのは正確キャンセルと不正確キャンセルによる桁落ち誤差の和のみである。各多項式に生じる不正確キャンセルの量は (3.2) で評価できるが、シジジー計算は非常に重く、しかも各係数に生じた不正確キャンセルの量はシジジー計算からは知り得ない。それを知るには巧みな工夫が必要である。

上記第一点より、入力係数が不正確な場合、必然的に“近似 Gröbner 基底”を計算することになる。

筆者が提案する新しい方法は次の三つの工夫からなる：上記第二点を解決するため“マーク”を導入するので、この方法を**高精度マーキング法**と呼ぶ。

1. 完全誤差項を除去するため各数係数は区間数か有効浮動小数で表し、正確な項キャンセルは高精度法で無害化する。入力系の初期相対精度を $\varepsilon_{\text{init}}$ 、出力系の各係数に生じる [正確+不正確な] 全桁落ち量を C_{total} とし、計算精度を ε_{cal} とする。 C_{total} をモニターしながら計算し、もしも

$$\varepsilon_{\text{cal}} < \varepsilon_{\text{init}}/C_{\text{total}} \quad (4.1)$$

ならば、計算は十分な精度で行われたとする。 ε_{cal} が (4.1) を満たさないと判明すれば、計算精度を上げて Gröbner 基底計算を初めからやり直す。

2. 初期多項式を二通りの方法でマーキングし、二つの初期基底を用意する。ここで、マーキングは次のように行われる。初期係数 c_0 が誤差 r_0 を持っているとするとき、大きさ $r_0/|c_0|$ の微小な数を、 c_0 に

加える (一つのマーキング) か、 c_0 から引く (別のマーキング) か、するのである。このことにより c_0 のビット列のうち r_0 の頭の位置のビットが変化 (マークされる) ことになる。マーキングのあと、二つの初期基底の全係数を上述のように高精度化する。

3. $\Phi_0 = \{F_1, \dots, F_r\}$ と $\Phi'_0 = \{F'_1, \dots, F'_r\}$ を、高精度化されて、(異なる) マーキングをされた二つの初期基底とする: Φ_0 と Φ'_0 の対応する係数は、マークされたビットより上位のビット列は全て等しいが、下位のビット列は異なっている。 ϵ_{cal} は (4.1) を満たすとする。 Φ_0, Φ'_0 から出発して計算される二つの対応する中間基底の多項式の対応する係数をそれぞれ c, c' とする。このとき、 c と c' を以下のように二つの部分に分ける。

$$c = \bar{c} + d, \quad c' = \bar{c} + d'. \tag{4.2}$$

ここで、 d と d' は先頭ビットが異なっており、 \bar{c} のビットのうち、 d と d' の先頭ビットにあたる部分から下位には 0 が並ぶ。よって、 \bar{c} のビットのうち d と d' の先頭ビットまでの数字が正しいといえる。故に、 c に含まれる不正確なキャンセル量は $\max\{|d|, |d'|\} / (\epsilon_{init} \bar{c})$ である。

図 2 で高精度マーキング法を説明する。 c_0 と c'_0 は異なってマークされた二つの初期基底の対応する係数である; 簡単のため 1.0 に規格化したので、マークは ϵ_{init} の位置である。 ooo と $***$ は、 c_0 と c'_0 を精度 ϵ_{cal} に増やす際に付加されたビットであり、無意味な数字の羅列である。マークが異なるので、 c_0 と c'_0 ではこれらの数字も異なっている。一方、 c と c' は二つの中間基底あるいは最終基底の対応する係数である。理解の容易さのため、マークの位置を c_0, c'_0 と合わせた。 c と c' の下位ビットの多くは正確な項キャンセルのため (形式的な) 誤差だらけだが、計算精度 ϵ_{cal} が十分あるので、正確な項キャンセルは ϵ_{init} より下位に現れる。一方、マークされた位置より上位のビットは正しく計算されているはずである。したがって、図の ϵ_{init} より上位に何ビット残っているかで、不正確な項キャンセル量が評価できる。

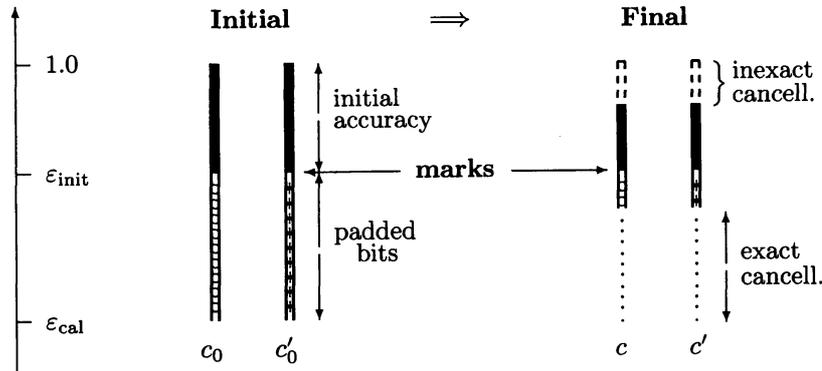


図 2: 高精度マーキング法の説明

例 2 (高精度マーキング法の適用例) 例 1 で与えた系に高精度マーキング法を適用する。

マークは、第一の系には $P \rightarrow P + 1.0e-13 \times rt(P)$ で、第二の系には $P \rightarrow P' = P - 1.0e-13 \times rt(P)$ で、それぞれ付与した。(このマーキングは簡単過ぎるだろう。実際の計算では係数毎に乱数で割り振りを変えるのがよからう)。二つの初期基底に対する Gröbner 基底を精度 10^{-30} の多倍長有効浮動小数で計算した。そのうちの P_6 と P'_6 を次頁に示す; 下線部は P_6 と P'_6 で合っている数字である。#BE[f, e] は、値部が f 、誤差部が e の多倍長有効浮動小数を表す。誤差部は $10^{-28} \times f$ に初期設定した。

例 1 では初期誤差が 2×10^{-16} に設定され、第一と第三の係数に約 10^5 の不正確な項キャンセルが発生していた。次頁の P_6 と P'_6 それぞれの第一と第三の係数を見ると、下線を付された数字 (正しく計算されたとと言える数字) の個数が例 1 に比べて 3 個づつ少ないが、これは初期精度が $10^3/2$ だけ低いためであり、

不正確キャンセル量は例1と例2でほぼ同じであることが分る。一方、例1と例2の両方で、第二と第四の係数はほとんど桁落ちしておらず、両者に矛盾はない。

$$\begin{aligned}
 P_6 &= y^2 z^2 - \#BE[2.995436923212796550471880942470, 4.0e-24] xy^2 \\
 &\quad - \#BE[1.0020782165123748434155553029900, 2.0e-25] y^3 \\
 &\quad + \#BE[1.9983254446538675410656723073600, 4.0e-25] xy \\
 &\quad + \#BE[1.0035217172564145740994791026400, 2.0e-25] y^2, \\
 P'_6 &= y^2 z^2 - \#BE[2.995436972279552973969466159280, 1.6e-24] xy^2 \\
 &\quad - \#BE[1.0020782165123748081194349163500, 7.7e-25] y^3 \\
 &\quad + \#BE[1.9983254937208258344069357248680, 1.6e-24] xy \\
 &\quad + \#BE[1.0035217172564143766318913416240, 7.7e-25] y^2.
 \end{aligned}$$

◇

5 正確・不正確な項キャンセルの低減化

論文 [22] で Traverso-Zanoni は多くの実際例を計算し、 2^{100} あるいはそれ以上の桁落ちが頻繁に生じることを報告している。このうち大部分は正確なキャンセルであり、それは高精度法で無害化できるものの、高精度になればなるほど計算時間がかかるので、正確な項キャンセル量を低減させたい。さらに、不正確な項キャンセルは得られた Gröbner 基底の精度を損なうので、不正確な項キャンセル量も低減させたい。

項キャンセル量の低減化に関しては、筆者らは昨年の数理研究集会で二つの簡単なアイデアを提案し、その有効性を実験で確認した [14]。アイデアの一つは、多項式に対して異常度 (abnormality) を、|主係数| が || 残余 || に比べて小さい又は大きいほど異常度が大きくなるように定義し、S 多項式生成と主項簡約においては異常度の小さい多項式から順に選択するものである。これは、数値行列に対するピボッティングに相当する操作であるが、項キャンセルの低減は不十分なのが現状である。とは言うものの、ヨーロッパの指導的研究者をして “Approximate Gröbner bases – an impossible concept?” [20] と言わしめた浮動小数 Gröbner 基底計算も、最大の壁は突破されて、算法の効率化の段階に至ったことは大進歩と言えよう。

参 考 文 献

- [1] M. Bodrato and A. Zanoni, Intervals, syzygies, numerical Gröbner bases: a mixed study. *Proceedings of CASC2006 (Computer Algebra in Scientific Computing)*: Springer-Verlag LNCS 4194, 64-76, 2006.
- [2] J.-C. Faugère, A new efficient algorithm for computing Gröbner bases (F_4). *J. Pure Appl. Algebra*, **139**, 61-88, 1999.
- [3] E. Fortuna, P. Gianni and B. Trager, Degree reduction under specialization. *J. Pure Appl. Algebra*, **164**, 153-164, 2001.
- [4] L. Gonzalez-Vega, C. Traverso and A. Zanoni. Hilbert stratification and parametric Gröbner bases. *Proceedings of CASC2005 (Computer Algebra in Scientific Computing)*: Springer-Verlag LNCS 3718, 220-235, 2005.
- [5] M. Kalkbrener, On the stability of Gröbner bases under specialization. *J. Symb. Comput.*, **24**, 51-58, 1997.
- [6] F. Kako and T. Sasaki, Proposal of “effective” floating-point number. Preprint of Univ. Tsukuba, May 1997 (unpublished).

- [7] A. Kondratyev, H.J. Stetter and S. Winkler, Numerical computation of Gröbner bases. *Proceedings of CASC2004 (Computer Algebra in Scientific Computing)*, 295-306, St. Petersburg, Russia, 2004.
- [8] B. Mourrain, A new criterion for normal form algorithms. *Lec. Notes Comp. Sci.*, **179**, 430-443, Springer, 1999.
- [9] B. Mourrain, Pythagore's dilemma, symbolic-numeric computation, and the border basis method. *Symbolic-Numeric Computations (Trends in Mathematics)*, 223-243, Birkhäuser Verlag, 2007.
- [10] K. Nagasaka, A Study on Gröbnerbasis with inexact input. *Lec. Notes Comp. Sci.: Proceedings of CASC2009 (Computer Algebra in Scientific Computing)*, **5743**, 248-258, Kobe, Japan, 2009.
- [11] T. Sasaki, A subresultant-like theory for Buchberger's procedure. Preprint of Univ. Tsukuba, 16 pages, Jan. 2010.
- [12] T. Sasaki and F. Kako, Computing floating-point Gröbner base stably. *Proceedings of SNC2007 (Symbolic Numeric Computation)*, 180-189, London, Canada, 2007.
- [13] T. Sasaki and F. Kako, Floating-point Gröbner basis computation with ill-conditionedness estimation. *Proceedings of ASCM2007 (Asian Symposium on Computer Mathematics)*, Singapore, Dec. 2007; see also LNAI 5081, 278-292, Deepak Kapur (Ed.), Springer, 2008.
- [14] 佐々木建昭, 甲斐博, 浮動小数 Gröbner 基底の悪条件性について. RIMS 研究集会: Computer Algebra – Design of Algorithms, Implementations and Applications. 2008 年 11 月, 京都.
- [15] K. Shirayanagi, An algorithm to compute floating-point Gröbner bases. *Mathematical Computation with Maple V. Ideas and Applications*, Birkhäuser, 95-106, 1993.
- [16] K. Shirayanagi, Floating point Gröbner bases. *Mathematics and Computers in Simulation*, **42**, 509-528, 1996.
- [17] K. Shirayanagi and M. Sweedler, Remarks on automatic algorithm stabilization. *J. Symb. Comput.*, **26**, 761-765, 1998.
- [18] H.J. Stetter, Stabilization of polynomial systems solving with Gröbner bases. *Proceedings of ISSAC'97 (Intern'l Symposium on Symbolic and Algebraic Computation)*, 117-124, ACM Press, 1997.
- [19] H.J. Stetter. *Numerical Polynomial Algebra*. SIAM Publ., Philadelphia, 2004.
- [20] H.J. Stetter, Approximate Gröbner bases – an impossible concept?. *Proceedings of SNC2005 (Symbolic-Numeric Computation)*, 235-236, Xi'an, China, 2005.
- [21] C. Traverso, Syzygies, and the stabilization of numerical Buchberger algorithm. *Proceedings of LMCS2002 (Logic, Mathematics and Computer Science)*, 244-255, RISC-Linz, Austria, 2002.
- [22] C. Traverso and A. Zanoni, Numerical stability and stabilization of Gröbner basis computation. *Proceedings of ISSAC2002 (Intern'l Symposium on Symbolic and Algebraic Computation)*, 262-269, ACM Press, 2002.
- [23] V. Weispfenning, Gröbner bases for inexact input data. *Proceedings of CASC2003 (Computer Algebra in Scientific Computing)*, 403-411, Passau, Germany, 2003.