

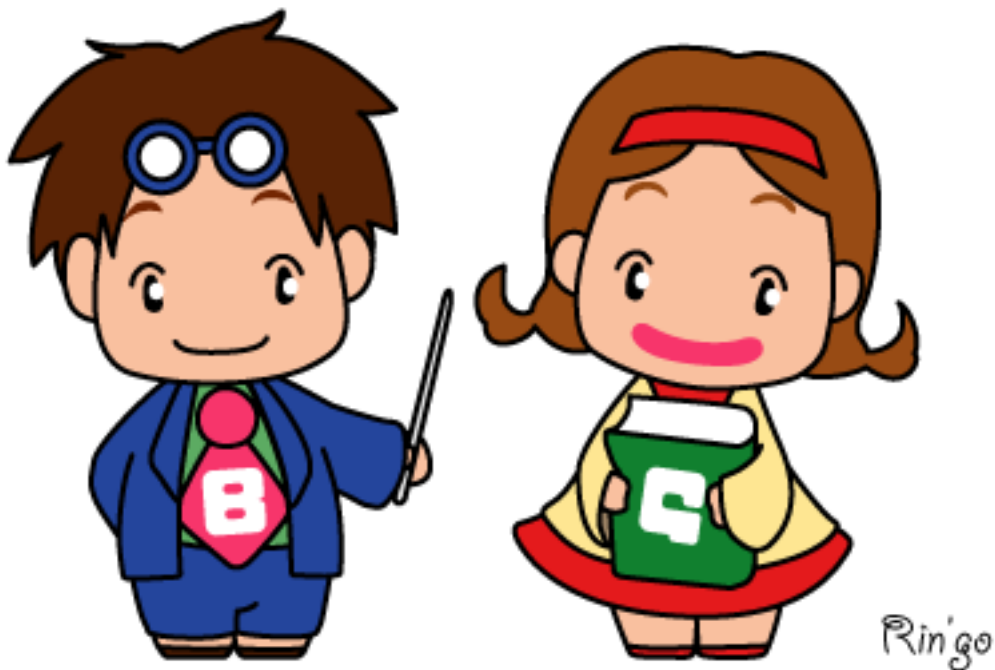
科学技術振興機構
戦略的創造研究推進事業 CREST 数学領域
“数学と諸分野の協働によるブレイクスルーの探索”

現代の産業社会とグレブナー基底の調和

<http://www.math.jst.go.jp/ja/scientists/teamhibi/index.html>

JST CREST GRÖBNER SCHOOL

日時：2009年 9月14日（月曜日）10:30～
9月18日（金曜日）15:15
場所：神戸大学理学部B棟428号室



はじめに

科学技術振興機構（JST）の戦略的創造研究推進事業（CREST）数学領域“数学と諸分野の協働によるブレイクスルーの探索”の研究課題「現代の産業社会とグレブナー基底の調和」は、2008年10月から研究活動を展開している。現代数学の潮流の一つを成すグレブナー基底の探究に携わる代数学者、計算機科学者、統計学者らから構成される共同研究組織を作り、グレブナー基底と先端科学技術との調和を議論する。具体的には、臨床試験、実験計画などの統計数学に有効なソフトを創出するために要求されるグレブナー基底の理論と計算機代数システムを、代数統計パッケージとして普及させる可能性とともに、周辺分野への応用も視野に入れながら、飛躍的に発展させる。現実の社会への積極的な貢献を念頭に置いた理論と計算機代数システムの相補的な展開は、従来の純粋数学の枠組では到達することが不可能であった純粋理論としての斬新な視点を育み、我が国のソフトウェア開発の能力の一層の向上を促進するものと期待される。

本 CREST 研究の研究期間は、2008年10月から2014年3月である。研究組織は <理論系グループ> <応用系グループ> <計算系グループ> から構成され、研究代表者は理論系グループに属している。応用系グループのグループリーダーは竹村彰通教授（東京大学大学院情報理工学系研究科）、計算系グループのグループリーダーは高山信毅教授（神戸大学大学院理学研究科）である。詳細は、

<http://www.math.jst.go.jp/ja/scientists/teamhibi/index.html>

を参照されたい。

次世代の数学ソフトウェアシステムは、ソフトウェア、解説文書、作業思考環境、書籍などを一体化させた“環境”である。加えて、その環境に一芸に秀でたソフトウェア群をうまく統合して行くのが、健全な開発方向である。Knoppix/Math はそのような方向性を持つ“desktop 環境”であるが、解説文書や書籍との統合、一芸に秀でたソフトウェア群との統合については全く未完成である。本 CREST 研究では、今後、様々な特色を持つグレブナー基底ソフトウェア群の性能増強とともに、これらを統合した次世代システムのプロトタイプを試験的に提供する。具体的な企画として、本 CREST 研究の共同研究者が分担執筆する著書を出版し、グレブナー基底の理論、応用、計算における成果を簡潔に紹介するとともに、開発したソフトウェアを付録に付け、一般のユーザーが臨床試験、実験計画などを含む広範な「現場」においてグレブナー基底を使う伊呂波に習熟できる“環境”の整備に着手する。

CREST 神戸 SCHOOL は、そのような“環境”整備の一環として立案された。すなわち、大学院生を主たる対象とし、グレブナー基底とその周辺の話題についての理解を深めるとともに、KNOPPIX/Math のユーザーになることを目標とし、講義のみならず、演習を充実させ、実際にパソコンを動かしながら最先端の話題に触れることができるように配慮した。本冊子は、講演原稿、板書ノート、配布資料、演習問題とその解答を集めた資料集である。付録として、研究代表者の応用数学会での総合講演の講演原稿を載せ、SCHOOL に参加した大学院生の感想文も添付している。神戸大学の会場を準備してくださった高山信毅教授、野呂正行教授とともに、演習問題と解答の作成に尽力してくださった中山洋将さん、西山絢太さんに感謝します。

2009年12月1日

研究代表者 日比 孝之
(大阪大学大学院情報科学研究科)

JST CREST GRÖBNER SCHOOL

科学技術振興機構 戦略的創造研究推進事業 CREST 数学領域 “数学と諸分野の協働によるブレークスルーの探索” の研究課題「現代の産業社会とグレブナー基底の調和」（平成20年10月～平成26年3月）の事業目的達成の一環として、下記の要領により CREST 神戸 SCHOOL を開催しますので、御案内申し上げます。

研究代表者 日比 孝之
(大阪大学大学院情報科学研究科)

記

日時：2009年 9月14日（月曜日）10：30～
9月18日（金曜日）15：15

場所：神戸大学理学部B棟428号室

<http://www.kobe-u.ac.jp/info/access/rokko/from-station.htm>

大学院生を主たる対象とし、グレブナー基底とその周辺の話題についての理解を深めるとともに、KNOPPIX/Math のユーザーになることを目標とする。

9月14日（月曜日）

（準備1）10：30～12：30

「理論的な観点からの導入」

大杉 英史（立教大学理学部・JST CREST）

日比 孝之（大阪大学大学院情報科学研究科・JST CREST）

[概要：グレブナー基底の歴史的な背景、グレブナー基底を導入するための基本的な概念（割り算アルゴリズムなど）を紹介し、イデアルの生成系からグレブナー基底を計算する Buchberger アルゴリズム、連立方程式を解くための消去法など、理論の基礎となる諸定理について解説する。]

（準備2）14：30～16：30

「計算機を使う観点からの導入」

野呂 正行（神戸大学大学院理学研究科・JST CREST）

濱田 龍義（福岡大学理学部・JST CREST）

[概要：グレブナー基底に関するアルゴリズムを実行できる環境として、KNOPPIX/Math 仮想マシンに習熟する。仮想マシン実行用ソフト（VMware Player、VMware Fusion）、仮想マシンのインストールを各自 PC に行ったあと、エディタの使い方などの基本操作を実習する。その後、（準備1）で学んだ諸概念を、幾つかのソフトウェアを使って実際に確認する。]

9月15日（火曜日）

（講義1）10：00～12：00

「マルコフ基底と実験計画」

竹村 彰通（東京大学大学院情報理工学系研究科・JST CREST）

青木 敏（鹿児島大学大学院理工学研究科・JST CREST）

[概要：実験計画法は、グレブナー基底理論の統計学への応用として、早くから注目されてきた分野の一つである。本講義では、まず、一部実施計画の別名関係とイデアル所属問題の対応を解説する。次に、実験計画データが計数値の場合の仮説検定を、グレブナー基底（マルコフ基底）を利用したモンテカルロ法で遂行する方法について解説する。]

（演習1）14：00～

「マルコフ基底と実験計画」

中山 洋将（神戸大学大学院理学研究科・JST CREST）

西山 絢太（神戸大学大学院理学研究科・JST CREST）

[概要：トーリックイデアルを使った、整数計画の解法、実行可能集合の数の上げの方法を実習する。システムとしては CoCoA、4ti2、Risa/Asir 等を利用する。数の上げの方法を基礎として、マルコフ基底を利用したモンテカルロ法を実習する。]

9月16日(水曜日)

(講義2) 10:00~12:00

「D-加群とグレブナー基底」

高山 信毅(神戸大学大学院理学研究科・JST CREST)

[概要: パラメータ付き積分の満たす微分方程式系を計算する問題を中心として、D-加群のアルゴリズムを紹介する。特に holonomic rank の計算、Pfaffian への変換、ODE section の計算、b-関数の計算、積分アルゴリズムについて解説する。付録として、A-超幾何方程式の b-関数を計算すると多面体の計算がいろいろできることも紹介する。]

(演習2) 14:00~

「D-加群とグレブナー基底」

中山 洋将(神戸大学大学院理学研究科・JST CREST)

西山 絢太(神戸大学大学院理学研究科・JST CREST)

[概要: Risa/Asir、Macaulay2、kan/sm1 などを利用し、holonomic rank の計算、Pfaffian への変換、ODE section の計算、b-関数の計算、積分アルゴリズムなどを実習する。可能であれば、Maple を利用して常微分方程式の解の級数展開の方法も実習する。]

9月17日(木曜日)

(講義3) 10:00~12:00

「凸多面体とグレブナー基底」

大杉 英史(立教大学理学部・JST CREST)

日比 孝之(大阪大学大学院情報科学研究科・JST CREST)

[概要: 凸多面体に付随するトーリックイデアルのグレブナー基底から構成される、多面体の三角形分割(正則三角形分割)、与えられたトーリックイデアルが持ち得るグレブナー基底を特徴付けるステイト多面体およびグレブナーファンについて解説する。]

(演習3) 14:00~

「凸多面体とグレブナー基底」

中山 洋将(神戸大学大学院理学研究科・JST CREST)

西山 絢太(神戸大学大学院理学研究科・JST CREST)

[概要: 凸多面体の種々の基本的な構成を polymake を使って実習し、Macaulay2 や Risa/Asir を使って、正則三角形分割をグレブナー基底から構成する。また、TiGERS などを利用し、正則三角形分割の数え上げ計算を行うとともに、gfan や kan/sm1 を利用し Groebner fan / tropical variety の計算を行う。]

9月18日(金曜日)

(展望1) 10:00~10:30

「非心ウィシャート分布のモーメントのグラフ表現とその応用」

栗木 哲(統計数理研究所数理・推論研究系・JST CREST)

沼田 泰英(東京大学大学院情報理工学系研究科・JST CREST)

(展望2) 10:45~11:15

「層別データの統計分析と記号推論」

大津 起夫(大学入試センター研究開発部・JST CREST)

(展望3) 11:30~12:00

「多変数公開鍵暗号とグレブナー基底」

只木孝太郎(中央大学研究開発機構・JST CREST)

(展望4) 14:00~14:30

「微分環と差分環の話題から」

野海 正俊(神戸大学大学院自然科学系先端融合研究環・JST CREST)

(展望5) 14:45~15:15

「役に立つ(かもしれない)最適化法」

武田 朗子(慶応義塾大学理工学部・JST CREST)

参加希望者は、予め、大杉英史 ohsugi@rkmath.rikkyo.ac.jp 宛、メールでの連絡をお願いします。8月10日(月曜日)を〆切りとします。旅費の援助を希望される場合は、その旨、記載してください。参加人数は約20名を予定しています。参加者はノートパソコンを持参してください。全日程参加を原則とします。その他、問い合わせ等も、ohsugi@rkmath.rikkyo.ac.jp 宛にお願いします。

9月14日(月曜日)

(準備1) 10:30~12:30

「理論的な観点からの導入」

大杉 英史(立教大学理学部・JST CREST)

日比 孝之(大阪大学大学院情報科学研究科・JST CREST)

[概要: グレブナー基底の歴史的な背景、グレブナー基底を導入するための基本的な概念(割り算アルゴリズムなど)を紹介し、イデアルの生成系からグレブナー基底を計算する Buchberger アルゴリズム、連立方程式を解くための消去法など、理論の基礎となる諸定理について解説する。]

INTRODUCTION TO GRÖBNER BASES

TAKAYUKI HIBI AND HIDEFUMI OHSUGI

Let $S = K[x_1, \dots, x_n]$ denote the polynomial ring in n variables over a field K with $\deg x_i = 1$ for $i = 1, 2, \dots, n$, and let

$$\text{Mon}(S) = \{x_1^{a_1} x_2^{a_2} \cdots x_n^{a_n} : a_i \in \mathbb{Z}_+, i = 1, 2, \dots, n\},$$

be the set of monomials of S , where \mathbb{Z}_+ is the set of nonnegative integers. In particular $1 \in \text{Mon}(S)$. For monomials $\mathbf{x}^{\mathbf{a}} = x_1^{a_1} x_2^{a_2} \cdots x_n^{a_n}$ and $\mathbf{x}^{\mathbf{b}} = x_1^{b_1} x_2^{b_2} \cdots x_n^{b_n}$ of S , we say that $\mathbf{x}^{\mathbf{b}}$ divides $\mathbf{x}^{\mathbf{a}}$ if $b_i \leq a_i$ for $i = 1, 2, \dots, n$. We write $\mathbf{x}^{\mathbf{b}} \mid \mathbf{x}^{\mathbf{a}}$ if $\mathbf{x}^{\mathbf{b}}$ divides $\mathbf{x}^{\mathbf{a}}$. Let \mathcal{M} be a nonempty subset of $\text{Mon}(S)$. A monomial $\mathbf{x}^{\mathbf{a}} \in \mathcal{M}$ is said to be a *minimal element* of \mathcal{M} with respect to divisibility if whenever $\mathbf{x}^{\mathbf{b}} \mid \mathbf{x}^{\mathbf{a}}$ with $\mathbf{x}^{\mathbf{b}} \in \mathcal{M}$, then $\mathbf{x}^{\mathbf{b}} = \mathbf{x}^{\mathbf{a}}$. Let \mathcal{M}^{\min} denote the set of minimal elements of \mathcal{M} .

Theorem 1 (DICKSON'S LEMMA). *Let \mathcal{M} be a nonempty subset of $\text{Mon}(S)$. Then \mathcal{M}^{\min} is a finite set.*

Proof. We prove Dickson's lemma by using induction on n , the number of variables of $S = K[x_1, x_2, \dots, x_n]$. Let $n = 1$. If d is the smallest integer for which $x_1^d \in \mathcal{M}$, then $\mathcal{M}^{\min} = \{x_1^d\}$. Thus \mathcal{M}^{\min} is a finite set.

Let $n \geq 2$ and $B = K[\mathbf{x}] = K[x_1, x_2, \dots, x_{n-1}]$. We use the notation y instead of x_n . Thus $S = K[x_1, x_2, \dots, x_{n-1}, y]$. Let \mathcal{M} be a nonempty subset of $\text{Mon}(S)$. Write \mathcal{N} for the subset of $\text{Mon}(B)$ which consists of those monomials $\mathbf{x}^{\mathbf{a}}$, where $\mathbf{a} \in \mathbb{Z}_+^{n-1}$, such that $\mathbf{x}^{\mathbf{a}} y^b \in \mathcal{M}$ for some $b \geq 0$. Our induction hypothesis says that \mathcal{N}^{\min} is a finite set. Let $\mathcal{N}^{\min} = \{u_1, u_2, \dots, u_s\}$. By the definition of \mathcal{N} , for each $1 \leq i \leq s$, there is $b_i \geq 0$ with $u_i y^{b_i} \in \mathcal{M}$. Let $b = \max\{b_1, b_2, \dots, b_s\}$. Now, for each $0 \leq \xi < b$, define the subset \mathcal{N}_ξ of \mathcal{N} to be

$$\mathcal{N}_\xi = \{\mathbf{x}^{\mathbf{a}} \in \mathcal{N} : \mathbf{x}^{\mathbf{a}} y^\xi \in \mathcal{M}\}.$$

Again, our induction hypothesis says that, for each $0 \leq \xi < b$, the set \mathcal{N}_ξ^{\min} is finite. Let $\mathcal{N}_\xi^{\min} = \{u_1^{(\xi)}, u_2^{(\xi)}, \dots, u_{s_\xi}^{(\xi)}\}$. We now show that each monomial belonging to \mathcal{M} is divisible by one of the monomials which appear in the following list:

$$\begin{aligned} &u_1 y^{b_1}, u_2 y^{b_2}, \dots, u_s y^{b_s}, \\ &u_1^{(0)}, u_2^{(0)}, \dots, u_{s_0}^{(0)}, \\ &u_1^{(1)} y, u_2^{(1)} y, \dots, u_{s_1}^{(1)} y, \\ &\dots\dots\dots \\ &u_1^{(b-1)} y^{b-1}, u_2^{(b-1)} y^{b-1}, \dots, u_{s_{b-1}}^{(b-1)} y^{b-1}. \end{aligned}$$

In fact, since, for each monomial $w = \mathbf{x}^{\mathbf{a}}y^\gamma \in \mathcal{M}$ with $\mathbf{x}^{\mathbf{a}} \in \text{Mon}(B)$, one has $\mathbf{x}^{\mathbf{a}} \in \mathcal{N}$, it follows that if $\gamma \geq b$, then w is divisible by one of the monomials $u_1y^{b_1}, u_2y^{b_2}, \dots, u_sy^{b_s}$, and that if $0 \leq \gamma < b$, then w is divisible by one of the monomials $u_1^{(\gamma)}y^\gamma, u_2^{(\gamma)}y^\gamma, \dots, u_{s_\gamma}^{(\gamma)}y^\gamma$. Clearly, the monomials listed above are in \mathcal{M} . Hence \mathcal{M}^{\min} is a subset of the set of monomials listed above. Thus \mathcal{M}^{\min} is finite, as desired. \square

A *monomial order* on S is a total order $<$ on $\text{Mon}(S)$ such that

- $1 < u$ for all $1 \neq u \in \text{Mon}(S)$;
- if $u, v \in \text{Mon}(S)$ and $u < v$, then $uw < vw$ for all $w \in \text{Mon}(S)$.

Example 2. (a) Let $\mathbf{a} = (a_1, a_2, \dots, a_n)$ and $\mathbf{b} = (b_1, b_2, \dots, b_n)$ be vectors belonging to \mathbb{Z}_+^n . We define the total order $<_{\text{lex}}$ on $\text{Mon}(S)$ by setting $\mathbf{x}^{\mathbf{a}} <_{\text{lex}} \mathbf{x}^{\mathbf{b}}$ if either (i) $\sum_{i=1}^n a_i < \sum_{i=1}^n b_i$, or (ii) $\sum_{i=1}^n a_i = \sum_{i=1}^n b_i$ and the left-most nonzero component of the vector $\mathbf{a} - \mathbf{b}$ is negative. It follows that $<_{\text{lex}}$ is a monomial order on S , which is called the *lexicographic order* on S induced by the ordering $x_1 > x_2 > \dots > x_n$.

(b) Let $\mathbf{a} = (a_1, a_2, \dots, a_n)$ and $\mathbf{b} = (b_1, b_2, \dots, b_n)$ be vectors belonging to \mathbb{Z}_+^n . We define the total order $<_{\text{rev}}$ on $\text{Mon}(S)$ by setting $\mathbf{x}^{\mathbf{a}} <_{\text{rev}} \mathbf{x}^{\mathbf{b}}$ if either (i) $\sum_{i=1}^n a_i < \sum_{i=1}^n b_i$, or (ii) $\sum_{i=1}^n a_i = \sum_{i=1}^n b_i$ and the right-most nonzero component of the vector $\mathbf{a} - \mathbf{b}$ is positive. It follows that $<_{\text{rev}}$ is a monomial order on S , which is called the *reverse lexicographic order* on S induced by the ordering $x_1 > x_2 > \dots > x_n$.

For example, $x_2x_3 <_{\text{lex}} x_1x_4$ and $x_1x_4 <_{\text{rev}} x_2x_3$ in $K[x_1, x_2, x_3, x_4]$. Among the monomials of degree 2 of $K[x_1, x_2, x_3]$, one has

$$x_3^2 <_{\text{lex}} x_2x_3 <_{\text{lex}} x_2^2 <_{\text{lex}} x_1x_3 <_{\text{lex}} x_1x_2 <_{\text{lex}} x_1^2$$

and

$$x_3^2 <_{\text{rev}} x_2x_3 <_{\text{rev}} x_1x_3 <_{\text{rev}} x_2^2 <_{\text{rev}} x_1x_2 <_{\text{rev}} x_1^2.$$

Exercise 3. List the 10 monomials of degree 3 of $K[x_1, x_2, x_3]$ with respect to each of $<_{\text{lex}}$ and $<_{\text{rev}}$.

Lemma 4. Let $<$ be a monomial order on S . Let $u, v \in \text{Mon}(S)$ with $u \neq v$ and suppose that u divides v . Then $u < v$.

Proof. Write $v = uw$ with $w \in \text{Mon}(S)$. Since $w \neq 1$, one has $1 < w$. Thus $1 \cdot u < w \cdot u$. Hence $u < v$, as desired. \square

We will work with a fixed monomial order $<$ on S . Let $f = \sum_{u \in \text{Mon}(S)} a_u u$ be a nonzero polynomial of S with each $a_u \in K$. The *support* of f is the finite set

$$\text{supp}(f) = \{u \in \text{Mon}(S) : a_u \neq 0\}.$$

The *initial monomial* of f with respect to $<$ is the biggest monomial with respect to $<$ among the monomials belonging to $\text{supp}(f)$.

Recall that an ideal of S is a nonempty subset I of S such that

- if $f, g \in I$, then $f \pm g \in I$;
- if $f \in I$ and $h \in S$, then $fh \in I$.

Given a subset $\{f_\lambda\}_{\lambda \in \Lambda}$ of S , we write $(\{f_\lambda\}_{\lambda \in \Lambda})$ for the set of polynomials of the form $\sum_{\lambda \in \Lambda} h_\lambda f_\lambda$, where $\{\lambda \in \Lambda : h_\lambda \neq 0\}$ is finite. Then $(\{f_\lambda\}_{\lambda \in \Lambda})$ is an ideal of S , which is called the ideal of S generated by $\{f_\lambda\}_{\lambda \in \Lambda}$. When Λ is finite, say, $\Lambda = \{1, 2, \dots, s\}$, we write (f_1, f_2, \dots, f_s) instead of $(\{f_1, f_2, \dots, f_s\})$. Conversely, given an ideal I of S , there exists a subset $(\{f_\lambda\}_{\lambda \in \Lambda})$ of S with $I = (\{f_\lambda\}_{\lambda \in \Lambda})$. We call $\{f_\lambda\}_{\lambda \in \Lambda}$ a system of generators of I . We say that an ideal I of S is *finitely generated* if I possesses a system of generators consisting of a finite number of polynomials. Later, we will see that every ideal of S is finitely generated (Corollary 9).

A *monomial ideal* is an ideal which is generated by a set of monomials. Let $I \subset S$ be a monomial ideal. It follows that I is generated by a subset $\mathcal{N} \subset \text{Mon}(S)$ if and only if $(I \cap \text{Mon}(S))^{\min} \subset \mathcal{N}$. Hence $(I \cap \text{Mon}(S))^{\min}$ is a *unique* minimal system of monomial generators of I . Dickson's lemma guarantees that $(I \cap \text{Mon}(S))^{\min}$ is finite. Thus in particular every monomial ideal is finitely generated.

Let I be a nonzero ideal of S . The *initial ideal* of I with respect to $<$ is the monomial ideal of S which is generated by $\{\text{in}_<(f) : 0 \neq f \in I\}$. We write $\text{in}_<(I)$ for the initial ideal of I . Thus

$$\text{in}_<(I) = (\{\text{in}_<(f) : 0 \neq f \in I\}).$$

Since $(\text{in}_<(I) \cap \text{Mon}(S))^{\min}$ is a minimal system of monomial generators of $\text{in}_<(I)$, and since $\text{in}_<(I) \cap \text{Mon}(S) = (\{\text{in}_<(f) : 0 \neq f \in I\})$, there exists a finite number of nonzero polynomials g_1, g_2, \dots, g_s belonging to I such that $\text{in}_<(I)$ is generated by the set $\{\text{in}_<(g_1), \text{in}_<(g_2), \dots, \text{in}_<(g_s)\}$ of their initial monomials.

Definition 5. Let I be a nonzero ideal of S . A finite set $\{g_1, g_2, \dots, g_s\}$ of nonzero polynomials with each $g_i \in I$ is said to be a *Gröbner basis* of I with respect to $<$ if the initial ideal $\text{in}_<(I)$ of I is generated by the set $\{\text{in}_<(g_1), \text{in}_<(g_2), \dots, \text{in}_<(g_s)\}$ of their initial monomials.

A Gröbner basis of I with respect to $<$ exists. If \mathcal{G} is a Gröbner basis of I with respect to $<$, then every finite set \mathcal{G}' with $\mathcal{G} \subset \mathcal{G}' \subset I$ is also a Gröbner basis of I with respect to $<$. If $\mathcal{G} = \{g_1, \dots, g_s\}$ is a Gröbner basis of I with respect to $<$ and if f_1, \dots, f_s are nonzero polynomials belonging to I with each $\text{in}_<(f_i) = \text{in}_<(g_i)$, then $\{f_1, \dots, f_s\}$ is also a Gröbner basis of I with respect to $<$.

Example 6. Let $S = K[x_1, x_2, \dots, x_7]$ and $I = (f, g)$, where $f = x_1x_4 - x_2x_3$ and $g = x_4x_7 - x_5x_6$. Let $<_{\text{lex}}$ the lexicographic order on S induced by $x_1 > x_2 > \dots > x_7$. One has $\text{in}_{<_{\text{lex}}}(f) = x_1x_4$ and $\text{in}_{<_{\text{lex}}}(g) = x_4x_7$. We claim that $\{f, g\}$ is not a Gröbner basis of I with respect to $<_{\text{lex}}$. In fact, the polynomial $h = x_7f - x_1g = x_1x_5x_6 - x_2x_3x_7$ belongs to I , but its initial monomial $\text{in}_{<_{\text{lex}}}(h) = x_1x_5x_6$ can be divided by neither $\text{in}_{<_{\text{lex}}}(f)$ nor $\text{in}_{<_{\text{lex}}}(g)$. Hence $\text{in}_{<_{\text{lex}}}(h) \notin (\text{in}_{<_{\text{lex}}}(f), \text{in}_{<_{\text{lex}}}(g))$. Thus $\text{in}_{<_{\text{lex}}}(I) \neq (\text{in}_{<_{\text{lex}}}(f), \text{in}_{<_{\text{lex}}}(g))$. In other words, $\{f, g\}$ is not a Gröbner basis of I with respect to $<_{\text{lex}}$. Later, we will show that $\{f, g, h\}$ is a Gröbner basis of I with respect to $<_{\text{lex}}$ (Example 16).

Lemma 7. Let $<$ be a monomial order on $S = K[x_1, \dots, x_n]$. Then, for any monomial u of S , there is no infinite descending sequence of the form

$$(1) \quad u = u_0 > u_1 > u_2 > \dots$$

Proof. Suppose, on the contrary, that one has an infinite descending sequence (1) and write \mathcal{M} for the set of monomials $\{u_0, u_1, u_2, \dots\}$. It follows from Dickson's lemma that \mathcal{M}^{\min} is a finite set, say $\mathcal{M}^{\min} = \{u_{i_1}, u_{i_2}, \dots, u_{i_s}\}$ with $i_1 < i_2 < \dots < i_s$. Then the monomial u_{i_s+1} is divided by u_{i_j} for some $1 \leq j \leq s$. Thus by Lemma 4 one has $u_{i_j} < u_{i_s+1}$, which contradicts $i_j < i_s + 1$. \square

Theorem 8. *Let I be a nonzero ideal of $S = K[x_1, \dots, x_n]$ and $\mathcal{G} = \{g_1, \dots, g_s\}$ a Gröbner basis of I with respect to a monomial order $<$ on S . Then $I = (g_1, \dots, g_s)$. In other words, every Gröbner basis of I is a system of generators of I .*

Proof. (Gordan) Let $0 \neq f \in I$. Since $\text{in}_<(f) \in \text{in}_<(I)$ and since \mathcal{G} is a Gröbner basis of I , i.e., $\text{in}_<(I) = (\text{in}_<(g_1), \dots, \text{in}_<(g_s))$, it follows that there is g_{i_0} such that $\text{in}_<(g_{i_0})$ divides $\text{in}_<(f)$. Let $\text{in}_<(f) = w_0 \text{in}_<(g_{i_0})$ with $w_0 \in \text{Mon}(S)$. Let $h_0 = f - c_{i_0}^{-1} c_0 w_0 g_{i_0}$, where c_0 is the coefficient of $\text{in}_<(f)$ in f and where c_{i_0} is the coefficient of $\text{in}_<(g_{i_0})$ in g_{i_0} . Then $h_0 \in I$. Since $\text{in}_<(w_0 g_{i_0}) = w_0 \text{in}_<(g_{i_0})$ it follows that $\text{in}_<(h_0) < \text{in}_<(f)$. If $h_0 = 0$, then $f \in (g_1, \dots, g_s)$.

Let $h_0 \neq 0$. Then the same technique as we used for f can be applied for h_0 . Thus $h_1 = f - c_{i_1}^{-1} c_1 w_1 g_{i_1} - c_{i_0}^{-1} c_0 w_0 g_{i_0}$, where c_1 is the coefficient of $\text{in}_<(h_0)$ in h_0 and where c_{i_1} is the coefficient of $\text{in}_<(g_{i_1})$ in g_{i_1} . Then $h_1 \in I$ and $\text{in}_<(h_1) < \text{in}_<(h_0)$. If $h_1 = 0$, then $f \in (g_1, \dots, g_s)$.

If $h_1 \neq 0$, then we proceed as before. Lemma 7 guarantees that this procedure must terminate. Thus we obtain an expression of the form $f = \sum_{q=0}^N c_{i_q}^{-1} c_q w_q g_{i_q}$. In particular, f belongs to (g_1, g_2, \dots, g_s) . Thus $I = (g_1, g_2, \dots, g_s)$, as desired. \square

Corollary 9 (HILBERT BASIS THEOREM). *Every ideal of the polynomial ring $S = K[x_1, \dots, x_n]$ is finitely generated.*

It is natural to ask if the converse of Theorem 8 is true or false. That is to say, if $I = (f_1, f_2, \dots, f_s)$ is an ideal of $S = K[x_1, \dots, x_n]$, then does there exist a monomial order $<$ on S such that $\{f_1, f_2, \dots, f_s\}$ is a Gröbner basis of I with respect to $<$?

Example 10 ([4]). Let $S = K[x_1, x_2, \dots, x_{10}]$ and I the ideal of S generated by

$$\begin{aligned} f_1 &= x_1 x_8 - x_2 x_6, & f_2 &= x_2 x_9 - x_3 x_7, & f_3 &= x_3 x_{10} - x_4 x_8, \\ f_4 &= x_4 x_6 - x_5 x_9, & f_5 &= x_5 x_7 - x_1 x_{10}. \end{aligned}$$

We claim that there exists *no* monomial order $<$ on S such that $\{f_1, \dots, f_5\}$ is a Gröbner basis of I with respect to $<$.

Suppose, on the contrary, that there exists a monomial order $<$ on S such that $\mathcal{G} = \{f_1, \dots, f_5\}$ is a Gröbner basis of I with respect to $<$. First, note that each of the five polynomials

$$\begin{aligned} &x_1 x_8 x_9 - x_3 x_6 x_7, \quad x_2 x_9 x_{10} - x_4 x_7 x_8, \quad x_2 x_6 x_{10} - x_5 x_7 x_8, \\ &x_3 x_6 x_{10} - x_5 x_8 x_9, \quad x_1 x_9 x_{10} - x_4 x_6 x_7 \end{aligned}$$

belongs to I . Let, say, $x_1 x_8 x_9 > x_3 x_6 x_7$. Since $x_1 x_8 x_9 \in \text{in}_<(I)$, there is $g \in \mathcal{G}$ such that $\text{in}_<(g)$ divides $x_1 x_8 x_9$. Such $g \in \mathcal{G}$ must be f_1 . Hence $x_1 x_8 > x_2 x_6$. Thus $x_2 x_6 \notin \text{in}_<(I)$. Hence there exists no $g \in \mathcal{G}$ such that $\text{in}_<(g)$ divides $x_2 x_6 x_{10}$. Hence $x_2 x_6 x_{10} < x_5 x_7 x_8$. Thus $x_5 x_7 > x_1 x_{10}$. Continuing these arguments, we obtain

$$\begin{aligned} x_1 x_8 x_9 &> x_3 x_6 x_7, \quad x_2 x_9 x_{10} > x_4 x_7 x_8, \quad x_2 x_6 x_{10} < x_5 x_7 x_8, \\ x_3 x_6 x_{10} &> x_5 x_8 x_9, \quad x_1 x_9 x_{10} < x_4 x_6 x_7 \end{aligned}$$

and

$$\begin{aligned} x_1x_8 &> x_2x_6, & x_2x_9 &> x_3x_7, & x_3x_{10} &> x_4x_8, \\ x_4x_6 &> x_5x_9, & x_5x_7 &> x_1x_{10}. \end{aligned}$$

Hence

$$(2) \quad (x_1x_8)(x_2x_9)(x_3x_{10})(x_4x_6)(x_5x_7) > (x_2x_6)(x_3x_7)(x_4x_8)(x_5x_9)(x_1x_{10}).$$

The opposite relation in (2) occurs in case of $x_1x_8x_9 < x_3x_6x_7$. However, both sides of the inequality (2) coincide with $x_1x_2 \cdots x_{10}$.

In high school mathematics, we learn that, given polynomials f and $g \neq 0$ in one variable x , there exist unique polynomials q and r such that $f = qg + r$, where either $r = 0$ or $\deg r < \deg g$. The division algorithm generalizes this well-known result.

Theorem 11 (DIVISION ALGORITHM). *Let $S = K[x_1, \dots, x_n]$ denote the polynomial ring in n variables over a field K and fix a monomial order $<$ on S . Let g_1, g_2, \dots, g_s be nonzero polynomials of S . Then, given a polynomial $0 \neq f \in S$, there exist polynomials f_1, f_2, \dots, f_s and f' of S with*

$$(3) \quad f = f_1g_1 + f_2g_2 + \cdots + f_sg_s + f'$$

such that the following conditions are satisfied:

- (i) if $f' \neq 0$ and if $u \in \text{supp}(f')$, then none of $\text{in}_<(g_1), \dots, \text{in}_<(g_s)$ divides u , i.e., no $u \in \text{supp}(f')$ belongs to $(\text{in}_<(g_1), \dots, \text{in}_<(g_s))$;
- (ii) if $f_i \neq 0$, then

$$\text{in}_<(f_i g_i) \leq \text{in}_<(f).$$

The right hand side of equation (3) is said to be a *standard expression* for f with respect to g_1, g_2, \dots, g_s , and the polynomial f' is called a *remainder* of f with respect to g_1, g_2, \dots, g_s .

Instead of giving a detailed proof of Theorem 11, we discuss a typical example which clearly explains the procedure to obtain a standard expression.

Example 12. Let $<_{\text{lex}}$ denote the lexicographic order on $S = K[x, y, z]$ induced by $x > y > z$. Let $g_1 = x^2 - z$, $g_2 = xy - 1$ and $f = x^3 - x^2y - x^2 - 1$. Each of

$$\begin{aligned} f &= x^3 - x^2y - x^2 - 1 = x(g_1 + z) - x^2y - x^2 - 1 \\ &= xg_1 - x^2y - x^2 + xz - 1 = xg_1 - (g_1 + z)y - x^2 + xz - 1 \\ &= xg_1 - yg_1 - x^2 + xz - yz - 1 = xg_1 - yg_1 - (g_1 + z) + xz - yz - 1 \\ &= (x - y - 1)g_1 + (xz - yz - z - 1) \end{aligned}$$

and

$$\begin{aligned} f &= x^3 - x^2y - x^2 - 1 = x(g_1 + z) - x^2y - x^2 - 1 \\ &= xg_1 - x^2y - x^2 + xz - 1 = xg_1 - x(g_2 + 1) - x^2 + xz - 1 \\ &= xg_1 - xg_2 - x^2 + xz - x - 1 = xg_1 - xg_2 - (g_1 + z) + xz - x - 1 \\ &= (x - 1)g_1 - xg_2 + (xz - x - z - 1) \end{aligned}$$

is a standard expression of f with respect to g_1 and g_2 , and each of $xz - yz - z - 1$ and $xz - x - z - 1$ is a remainder of f .

Example 12 says that a remainder of a nonzero polynomial may not be unique. However, we have the following fact.

Lemma 13. *If $\mathcal{G} = \{g_1, \dots, g_s\}$ is a Gröbner basis of $I = (g_1, \dots, g_s)$, then for any nonzero polynomial f of S , there is a unique remainder of f with respect to g_1, \dots, g_s .*

Proof. Suppose there exist remainders f' and f'' with respect to g_1, \dots, g_s with $f' \neq f''$. Since $0 \neq f' - f'' \in I$, the initial monomial $w = \text{in}_<(f' - f'')$ must belong to $\text{in}_<(I)$. However, since $w \in \text{supp}(f') \cup \text{supp}(f'')$, none of the monomials $\text{in}_<(g_1), \dots, \text{in}_<(g_s)$ divides w . Hence $\text{in}_<(I) \neq (\text{in}_<(g_1), \dots, \text{in}_<(g_s))$. \square

Given nonzero polynomials f and g of S , the notation $\text{lcm}(\text{in}_<(f), \text{in}_<(g))$ stands for the least common multiple of $\text{in}_<(f)$ and $\text{in}_<(g)$. Let c_f denote the coefficient of $\text{in}_<(f)$ in f and c_g the coefficient of $\text{in}_<(g)$ in g . The polynomial

$$S(f, g) = \frac{\text{lcm}(\text{in}_<(f), \text{in}_<(g))}{c_f \text{in}_<(f)} f - \frac{\text{lcm}(\text{in}_<(f), \text{in}_<(g))}{c_g \text{in}_<(g)} g$$

is called the *S-polynomial* of f and g .

We say that f has remainder 0 with respect to g_1, g_2, \dots, g_s if, in the division algorithm, there is a standard expression (3) of f with respect to g_1, g_2, \dots, g_s with $f' = 0$.

Lemma 14. *Let f and g be nonzero polynomials and suppose that $\text{in}_<(f)$ and $\text{in}_<(g)$ are relatively prime, i.e., $\text{lcm}(\text{in}_<(f), \text{in}_<(g)) = \text{in}_<(f) \text{in}_<(g)$. Then $S(f, g)$ has remainder 0 with respect to f, g .*

Proof. To simplify notation we will assume that each of the coefficients of $\text{in}_<(f)$ in f and $\text{in}_<(g)$ in g is equal to 1. Let $f = \text{in}_<(f) + f_1$ and $g = \text{in}_<(g) + g_1$. Since $\text{in}_<(f)$ and $\text{in}_<(g)$ are relatively prime, it follows that

$$\begin{aligned} S(f, g) &= \text{in}_<(g)f - \text{in}_<(f)g \\ &= (g - g_1)f - (f - f_1)g \\ &= f_1g - g_1f. \end{aligned}$$

We claim $(\text{in}_<(f_1) \text{in}_<(g) =) \text{in}_<(f_1g) \neq \text{in}_<(g_1f) (= \text{in}_<(g_1) \text{in}_<(f))$. In fact, if $\text{in}_<(f_1) \text{in}_<(g) = \text{in}_<(g_1) \text{in}_<(f)$, then, since $\text{in}_<(f)$ and $\text{in}_<(g)$ are relatively prime, it follows that $\text{in}_<(f)$ must divide $\text{in}_<(f_1)$. However, since $\text{in}_<(f_1) < \text{in}_<(f)$, this is impossible. Let, say, $\text{in}_<(f_1) \text{in}_<(g) < \text{in}_<(g_1) \text{in}_<(f)$. Then $\text{in}_<(S(f, g)) = \text{in}_<(g_1f)$ and $S(f, g) = f_1g - g_1f$ turns out to be a standard expression of $S(f, g)$ in terms of f and g . Hence $S(f, g)$ has remainder 0 with respect to f and g , and similarly for $\text{in}_<(g_1) \text{in}_<(f) < \text{in}_<(f_1) \text{in}_<(g)$. \square

We now come to the most fundamental theorem in the theory of Gröbner bases.

Theorem 15 (BUCHBERGER CRITERION). *Let I be a nonzero ideal of S and $\mathcal{G} = \{g_1, g_2, \dots, g_s\}$ a system of generators of I . Then \mathcal{G} is a Gröbner basis of I if and only if the following condition is satisfied:*

(*) *For all $i \neq j$, $S(g_i, g_j)$ has remainder 0 with respect to g_1, \dots, g_s .*

We refer the reader to a standard textbook on Gröbner bases, e.g., [1], [2] and [3] for a proof of the Buchberger criterion. However, for a (general) Gröbner basis “user,” it may not be required to understand a detailed proof of the Buchberger criterion.

In Example 6, by using Lemma 14 together with the Buchberger criterion, it follows immediately that the set $\{f, g\}$ is a Gröbner basis of $I = (f, g)$ with respect to the reverse lexicographic order $<_{\text{rev}}$ induced by $x_1 > x_2 > \cdots > x_7$.

The Buchberger criterion supplies an algorithm to compute a Gröbner basis starting from a system of generators of an ideal.

Let $\{g_1, g_2, \dots, g_s\}$ be a system of generators of a nonzero ideal I of S and suppose that $\{g_1, g_2, \dots, g_s\}$ is *not* a Gröbner basis of I . The Buchberger criterion then guarantees that there is an S -polynomial $S(g_i, g_j)$ such that *no* remainder of $S(g_i, g_j)$ with respect to g_1, g_2, \dots, g_s is 0. Let $h_{ij} \in I$ be a remainder of a standard expression of $S(g_i, g_j)$ with respect to g_1, g_2, \dots, g_s . Then $\text{in}_<(h_{ij})$ can be divided by none of the monomials $\text{in}_<(g_1), \text{in}_<(g_2), \dots, \text{in}_<(g_s)$. In other words, the inclusion

$$(\text{in}_<(g_1), \text{in}_<(g_2), \dots, \text{in}_<(g_s)) \subset (\text{in}_<(g_1), \text{in}_<(g_2), \dots, \text{in}_<(g_s), \text{in}_<(h_{ij})).$$

is strict. With setting $g_{s+1} = h_{ij}$, suppose that $\{g_1, g_2, \dots, g_s, g_{s+1}\}$ is not a Gröbner basis of I . Again, by using the Buchberger criterion, there is a S -polynomial $S(g_k, g_\ell)$ such that no remainder of $S(g_k, g_\ell)$ with respect to $g_1, g_2, \dots, g_s, g_{s+1}$ is 0. Let $h_{k\ell} \in I$ be a remainder of $S(g_k, g_\ell)$ with respect to $g_1, g_2, \dots, g_s, g_{s+1}$. Then the inclusion

$$\begin{aligned} (\text{in}_<(g_1), \text{in}_<(g_2), \dots, \text{in}_<(g_s), \text{in}_<(g_{s+1})) \\ \subset (\text{in}_<(g_1), \text{in}_<(g_2), \dots, \text{in}_<(g_s), \text{in}_<(g_{s+1}), \text{in}_<(h_{k\ell})). \end{aligned}$$

is strict. By virtue of Dickson’s lemma, these procedures must terminate after a finite number of steps, and a Gröbner basis of I can be obtained.

The above algorithm to find a Gröbner basis starting from a system of generators of an ideal is said to be the *Buchberger algorithm*.

Example 16. We continue Example 6. Let $S = K[x_1, x_2, \dots, x_7]$ and $<_{\text{lex}}$ the lexicographic order on S induced by $x_1 > x_2 > \cdots > x_7$. Let $f = x_1x_4 - x_2x_3$ and $g = x_4x_7 - x_5x_6$. Thus $\text{in}_{<_{\text{lex}}}(f) = x_1x_4$ and $\text{in}_{<_{\text{lex}}}(g) = x_4x_7$. Let $I = (f, g)$. Then $\{f, g\}$ is not a Gröbner basis of I with respect to $<_{\text{lex}}$. Now, as a remainder of $S(f, g) = x_7f - x_1g = x_1x_5x_6 - x_2x_3x_7$ with respect to f and g , we choose $S(f, g)$ itself. Let $h = x_1x_5x_6 - x_2x_3x_7$ with $\text{in}_{<_{\text{lex}}}(h) = x_1x_5x_6$. Then $\text{in}_{<_{\text{lex}}}(g)$ and $\text{in}_{<_{\text{lex}}}(h)$ are relatively prime. On the other hand, $S(f, h) = x_2x_3(x_4x_7 - x_5x_6)$ has remainder 0 with respect to f, g, h . It follows from the Buchberger criterion that $\{f, g, h\}$ is a Gröbner basis of I with respect to $<_{\text{lex}}$.

The following theorem is called *Elimination Theorem* and plays an important role when solving a system of equations.

Theorem 17. *Let $S' = K[x_{i_1}, \dots, x_{i_m}]$ be the subring of $S = K[x_1, \dots, x_n]$ where $1 \leq i_1 < \cdots < i_m \leq n$ and let $<$ a monomial order on S (and S'). Let \mathcal{G} denote a Gröbner basis of a nonzero ideal I of S with respect to $<$. If $<$ satisfies the condition*

$$(\sharp) \quad g \in \mathcal{G}, \text{in}_<(g) \in S' \implies g \in S'$$

then $\mathcal{G} \cap S'$ is a Gröbner basis of $I \cap S'$ with respect to $<$.

Proof. Let u be a monomial belonging to $\text{in}_<(I \cap S')$. Then there exists a polynomial ($0 \neq$) $f \in I \cap S'$ such that $\text{in}_<(f) = u$. Since $f \in I$, the initial monomial u belongs to $\text{in}_<(I)$. Hence there exists $g \in \mathcal{G}$ such that $\text{in}_<(g)$ divides u . Then $\text{in}_<(g)$ belongs to S' . Thanks to the condition (\sharp) , we have $g \in S'$ and hence $g \in \mathcal{G} \cap S'$. Thus $\text{in}_<(I \cap S')$ is generated by $\{\text{in}_<(g) : g \in \mathcal{G} \cap S'\}$ as desired. \square

Example 18. Let $\mathbf{a} = (a_1, a_2, \dots, a_n)$ and $\mathbf{b} = (b_1, b_2, \dots, b_n)$ be vectors belonging to \mathbb{Z}_+^n . We define the total order $<_{\text{plex}}$ on $\text{Mon}(S)$ by setting $\mathbf{x}^{\mathbf{a}} <_{\text{plex}} \mathbf{x}^{\mathbf{b}}$ if the left-most nonzero component of the vector $\mathbf{a} - \mathbf{b}$ is negative. It follows that $<_{\text{plex}}$ is a monomial order on S , which is called the *purely lexicographic order* on S induced by the ordering $x_1 > x_2 > \dots > x_n$. If $S' = K[x_m, x_{m+1}, \dots, x_n]$ is a subring of $S = K[x_1, \dots, x_n]$, then the condition (\sharp) in Theorem 17 holds for a Gröbner basis \mathcal{G} of an arbitrary ideal I of S with respect to $<_{\text{plex}}$.

Let $f_1, \dots, f_s, g_1, \dots, g_t \in S$. It is easy to see that, if $(f_1, \dots, f_s) = (g_1, \dots, g_t)$ holds, then the set of solutions of $f_1 = \dots = f_s = 0$ equals to that of $g_1 = \dots = g_t = 0$. Thus, one can eliminate the variables x_1, \dots, x_{m-1} from $f_1 = \dots = f_s = 0$ by computing a system of generators of $I \cap K[x_m, x_{m+1}, \dots, x_n]$. Thanks to Theorem 8, we can apply Elimination Theorem to eliminate variables from a system of equations.

Example 19 ([3]). Let $f_1 = x^2 + y + z - 1$, $f_2 = x + y^2 + z - 1$ and $f_3 = x + y + z^2 - 1$ and consider the system of equations $f_1 = f_2 = f_3 = 0$. Let $I = (f_1, f_2, f_3)$. Then $\{x + y + z^2 - 1, y^2 - y - z^2 + z, 2yz^2 + z^4 - z^2, z^6 - 4z^4 + 4z^3 - z^2\}$ is a Gröbner basis of I with respect to $<_{\text{plex}}$ induced by $x > y > z$. Thus, thanks to Theorem 17,

$$\begin{aligned} I \cap \mathbb{C}[z] &= (z^6 - 4z^4 + 4z^3 - z^2) \\ I \cap \mathbb{C}[y, z] &= (y^2 - y - z^2 + z, 2yz^2 + z^4 - z^2, z^6 - 4z^4 + 4z^3 - z^2) \end{aligned}$$

Note that $z^6 - 4z^4 + 4z^3 - z^2 = z^2(z-1)^2(z^2 + 2z - 1)$.

REFERENCES

- [1] W. Adams and P. Loustaunau, "An Introduction to Gröbner Bases," Amer. Math. Soc., Providence, RI, 1994.
- [2] T. Becker and V. Weispfenning, "Gröbner Bases," Springer-Verlag, Berlin, Heidelberg, New York, 1993.
- [3] D. Cox, J. Little and D. O'Shea, "Ideals, Varieties and Algorithms," Springer-Verlag, Berlin, Heidelberg, New York, 1992.
- [4] H. Ohsugi and T. Hibi, Toric ideals generated by quadratic binomials, *J. Algebra* **218** (1999), 509–527.

TAKAYUKI HIBI, DEPARTMENT OF PURE AND APPLIED MATHEMATICS, GRADUATE SCHOOL OF INFORMATION SCIENCE AND TECHNOLOGY, OSAKA UNIVERSITY, TOYONAKA, OSAKA 560-0043, JAPAN.

E-mail address: hibi@math.sci.osaka-u.ac.jp

HIDEFUMI OHSUGI, DEPARTMENT OF MATHEMATICS, COLLEGE OF SCIENCE, RIKKYO UNIVERSITY, TOSHIMA-KU, TOKYO 171-8501, JAPAN

E-mail address: ohsugi@rkmath.rikkyo.ac.jp

「理論的+観点からの導入」

グレブナー基底入門 K : 体

$$S = K[x_1, \dots, x_n]$$

$$\text{Mon}(S) = \{ x_1^{a_1} \cdots x_n^{a_n} \mid a_i \in \mathbb{Z}_{\geq 0} \}$$

$$\text{特に } 1 = x_1^0 \cdots x_n^0 \in \text{Mon}(S)$$

$$x^a = x_1^{a_1} \cdots x_n^{a_n}, x^b = x_1^{b_1} \cdots x_n^{b_n} \text{ のとき}$$

$$x^a \mid x^b \text{ (} x^a \text{ が } x^b \text{ を割り切れる)} \stackrel{\text{def}}{\iff} a_i \leq b_i \quad \forall i$$

$$\emptyset \neq M \subset \text{Mon}(S)$$

$$x^a \in M \text{ が 極小元} \stackrel{\text{def}}{\iff} x^b \in M, x^b \mid x^a \text{ ならば } x^b = x^a$$

 M^{\min} : M の極小元全体の集合定理 1 (Dickson の補題) M^{\min} は有限集合である S 上の 単項式順序 とは、 $\text{Mon}(S)$ 上の全順序であって

$$i) 1 < u \quad \forall u \in \text{Mon}(S) \setminus \{1\}$$

$$ii) u < v \implies wu < wv \quad \forall u, v, w \in \text{Mon}(S)$$

例 2

$$a = (a_1, \dots, a_n), b = (b_1, \dots, b_n) \text{ とするとき}$$

・辞書式順序

$$x^a <_{\text{lex}} x^b \stackrel{\text{def}}{\iff} (i) \sum a_i < \sum b_i$$

$$(ii) \sum a_i = \sum b_i \text{ かつ } a-b \text{ の成分で最も左にある } 0 \text{ ではないものは負}$$

・逆辞書式順序

$$x^a <_{\text{rev}} x^b \stackrel{\text{def}}{\iff} (i)$$

(ii)

"

"

右 "

" 正

練習 3 各自試みる。

補題4

$u, v \in \text{Mon}(S)$, $u \neq v$ とする。このとき

$$u \mid v \Rightarrow u < v$$

証明:

$v = wu$, $w \neq 1$ とする単項式 w がある。

$w \neq 1$ より $1 < w$

すると $1 \cdot u < w \cdot u \quad \therefore u < v$ □

• $0 \neq f = \sum_{u \in \text{Mon}(S)} a_u u \in S \quad (a_u \in K)$
 \sim 有限和

$\text{supp}(f) := \{u \mid a_u \neq 0\}$

$\text{in}_<(f) := \text{supp}(f)$ に含まれる u に関して最大の単項式, initial monomial と呼ぶ

• ICS イデアル

$\{f_\lambda\}_{\lambda \in \Lambda}$ が I の生成系 $\stackrel{\text{def}}{\iff} I = \left\{ \sum_{\text{有限和}} h_\lambda f_\lambda \mid h_\lambda \in S \right\}$

このとき $I = (\{f_\lambda\}_{\lambda \in \Lambda})$ と書く。

特に生成系が有限集合のときには $I = (f_1, \dots, f_s)$ と表す。

I は有限生成であると言う。

• 単項式から成る生成系を持つイデアルを 単項式イデアル と呼ぶ。すると

単項式イデアル I は有限生成である

実際 $(I \cap \text{Mon}(S))^{\text{min}}$ は I の生成系であり、Dicksonの補題より有限集合である。

• $0 \neq I \subset S$: イデアル

$\text{in}_<(I) := (\text{in}_<(f) \mid 0 \neq f \in I)$ I の initial ideal

定義5

有限集合 $\{g_1, \dots, g_s\} \subset I$ が I の グラーブ-基底 とは

$$\text{in}_<(I) = (\text{in}_<(g_1), \dots, \text{in}_<(g_s))$$

なるときから。

例6 (例16を参照)

補題7

単項式の無限減少列

$$u = u_0 > u_1 > u_2 > \dots$$

は存在しない

証明: Dicksonの補題 + 補題4

定理8

I のグレブナー基底は I の生成系である

証明:

$\{g_1, \dots, g_s\}$ を I の GB とする. $0 \neq f \in I$ があれば.

$$\text{in}_c(f) \in \text{in}_c(I) = (\text{in}_c(g_1), \dots, \text{in}_c(g_s))$$

$$\exists w \in \text{Mon}(S) \quad \exists i \quad \text{in}_c(f) = w \text{in}_c(g_i)$$

$$f = c_0 \text{in}_c(f) + \dots, \quad g_i = c_i \text{in}_c(g_i) + \dots \quad \text{とすれば}$$

$h := f - c_0 c_i^{-1} g_i$ とすれば $h \in I$ であり $h \neq 0$ ならば $\text{in}_c(h) < \text{in}_c(f)$ である. この操作は補題7 により有限回で終了し $f \in (g_1, \dots, g_s)$ を得る

系9

多項式環 S のイデアルは有限生成である. (Hilbertの基底定理)

例10 (大杉の3大反例の一つ)

定理11 (割り算アルゴリズム)

f の g_1, \dots, g_s に関する標準表示

g_1, \dots, g_s : 0 でない多項式

任意の $0 \neq f \in S$ に対して、次の表示が存在する.

$$f = f_1 g_1 + \dots + f_s g_s + f' \quad (f_1, \dots, f_s, f' \in S)$$

但し ① $f' \neq 0$ ならば $\forall u \in \text{supp}(f') \quad \forall i$ について $\text{in}_c(g_i) \nmid u$

② $f_i \neq 0$ ならば $\text{in}_c(f_i g_i) \equiv \text{in}(f)$

f' は f の g_1, \dots, g_s に関する余りという

例12

$$S = K[x, y, z] \quad <_{\text{lex}}$$

$$g_1 = (x^2 - z), \quad g_2 = (xy - 1), \quad f = x^3 - x^2y - x^2 - 1$$

$$f = x^3 - x^2y - x^2 - 1$$

$$(g_1 \text{ 割り算}) = x(g_1 + z) - x^2y - x^2 - 1$$

$$= xg_1 + xz - x^2y - x^2 - 1$$

g_1 で割り算 ← → g_2 で割り算

$$\begin{aligned}
 g_1 \text{ で割る} &= x \cdot g_1 + xz - (g_1 + z)y - x^2 - 1 \\
 &= (x-y)g_1 + xz - yz - \underline{x^2 - 1} \\
 &= (x-y)g_1 + xz - yz - (g_1 + z) - 1 \\
 &= (x-y-1)g_1 + (xz - yz - z - 1) \quad \text{余り}
 \end{aligned}$$

$$\begin{aligned}
 g_2 \text{ で割る} &= xg_1 + xz - x(g_2 + 1) - x^2 - 1 \\
 &= xg_1 - xg_2 + xz - \underline{x^2} - x - 1 \\
 &= xg_1 - xg_2 + xz - (g_1 + z) - x - 1 \\
 &= (x-1)g_1 - xg_2 + (xz - x - z - 1) \quad \text{余り}
 \end{aligned}$$

余りは一意の
でない

補題 13.

$\{g_1, \dots, g_s\}$ から $I = (g_1, \dots, g_s)$ の GB ならば、任意の $0 \neq f \in S$ の g_1, \dots, g_s に関する余りは一意のである

証明:

f' と f'' が f の余りで $f' \neq f''$ と せよ。すると $f - f' \in I$, $f - f'' \in I$ であるから $0 \neq f' - f'' \in I$ 。すると $\text{inc}(f' - f'') \in \text{inc}(I) = (\text{inc}(g_1), \dots, \text{inc}(g_s))$

$$\exists i \quad \text{inc}(g_i) \mid \text{inc}(f' - f'')$$

他方、 $\text{inc}(f' - f'') \in \text{supp}(f') \cup \text{supp}(f'')$ であるから

$$\forall i \quad \text{inc}(g_i) \nmid \text{inc}(f' - f'')$$

矛盾

□

$S = k[x_1, \dots, x_n]$
 $<$: 単項式順序

$(0 \neq) f, g \in S$ に対し

$\text{lcm}(\text{inc}(f), \text{inc}(g))$: $\text{inc}(f)$ と $\text{inc}(g)$ の最小公倍数

$C_f = f$ における $\text{inc}(f)$ の係数

$C_g = g$ における $\text{inc}(g)$ の係数

$$S(f, g) := \frac{\text{lcm}(\text{inc}(f), \text{inc}(g))}{C_f \text{inc}(f)} f - \frac{\text{lcm}(\text{inc}(f), \text{inc}(g))}{C_g \text{inc}(g)} g$$

f と g の S -多項式

例) $f = x_1 x_4 - x_2 x_3$ $\text{inc}_{lex} f$
 $g = 2x_4 x_7 - x_5 x_6$ $\text{inc}_{lex} g$

$<_{lex}: x_1 > x_2 > \dots > x_7$

$=0$ とは

$$\begin{aligned} S(f, g) &= \frac{x_1 x_4 x_7}{1 \cdot x_1 x_4} f - \frac{x_1 x_4 x_7}{2 \cdot x_4 x_7} g \\ &= \frac{x_1 x_4 x_7}{x_1 x_4} (x_1 x_4 - x_2 x_3) - \frac{x_1 x_4 x_7}{2 x_4 x_7} (2x_4 x_7 - x_5 x_6) \\ &= -x_2 x_3 x_7 + \frac{1}{2} x_1 x_5 x_6 \end{aligned}$$

先頭項が打ち消し合う

$f, g_1, g_2, \dots, g_s \in S$

f が g_1, \dots, g_s に関して 余り 0 \iff 割算アルゴリズムにおいて
 $f = f_1 g_1 + \dots + f_s g_s + 0$
 という標準表示が存在する

補題 14

$(0 \neq) f, g \in S$ に対して

$$\text{lcm}(\text{inc}(f), \text{inc}(g)) = \text{inc}(f) \text{inc}(g)$$

$$\implies S(f, g) \text{ は } f, g \text{ に関して余り } 0$$

証明:

簡単のため $C_f = C_g = 1$ とする

$$f = \text{inc}(f) + f_1 \quad \text{とか}$$

$$g = \text{inc}(g) + g_1$$

$$S(f, g) = \frac{\text{inc}(f) \text{inc}(g)}{\text{inc}(f)} f - \frac{\text{inc}(f) \text{inc}(g)}{\text{inc}(g)} g$$

$$= \text{inc}(g) f - \text{inc}(f) g$$

$$= (g - g_1) f - (f - f_1) g$$

$$= f_1 g - g_1 f$$

\leftarrow これが標準表示であるといふ

$\therefore \text{inc}(f_1 g), \text{inc}(g_1 f) \leq \text{inc}(S(f, g))$ を示せばよい

$inc(f, g) \neq inc(g, f)$ を示す.

$$\begin{matrix} inc(f_1) & inc(g) \\ || & || \\ inc(g_1) & inc(f) \end{matrix}$$
 $inc(f_1) inc(g) = inc(g_1) inc(f)$ と仮定すると、 $inc(f)$ と $inc(g)$ は互いに素だから $inc(f_1)$ は $inc(f)$ で割り切れる。 $\therefore inc(f_1) > inc(f)$ これは矛盾 //

定理15 (Buchberger 判定法)

$0 \neq I = (g_1, \dots, g_s) \subset S$
 $= 0$ とは

$G = \{g_1, \dots, g_s\}$ が I の GLT-基底
 \Leftrightarrow 任意の $i \neq j$ に対し $S(g_i, g_j)$ は g_1, \dots, g_s に関して余り 0

Buchberger アルゴリズム

Input: $\{g_1, \dots, g_s\} \subset S$
 $<$: 単項式順序

Output: $I = (g_1, \dots, g_s)$ の $<$ に関する GB

- ① $G = \{g_1, \dots, g_s\}$
- ② $G' = G$ とおく

各 $f_1, f_2 \in G'$ に対し、 $S(f_1, f_2)$ の G' に関する余りが 0 でない (f_1, f_2) ならば $G = G \cup \{f\}$ とおく。

- ③ $G = G'$ ならば G を出力
 $G \neq G'$ ならば ② に戻る。

① アルゴリズムは必ず停止する!

① ② において $t \neq 0$ のとき、 t は "余り" だから $inc(t)$ も割り切るおの $inc(g)$ ($g \in G'$) は存在しない。

$\therefore inc(t) \notin (inc(g) \mid g \in G')$
 $\therefore (inc(g) \mid g \in G') \subsetneq (inc(g) \mid g \in G' \cup \{t\})$

もし停止しないときは、

$(inc(g_1), \dots, inc(g_s))$
 $\subsetneq (inc(g_1), \dots, inc(g_s), inc(g_{s+1}))$
 $\subsetneq (inc(g_1), \dots, inc(g_s), inc(g_{s+1}), inc(g_{s+2}))$

と無限列が構成される。これは Dickson の補題に矛盾 (Hilbert の基底定理) //

Example 16

$S = K[x_1, \dots, x_7]$
 $<_{lex} : x_1 > \dots > x_7$
 $f = x_1 x_4 - x_2 x_3$
 $g = x_4 x_7 - x_5 x_6$
 $I = (f, g)$

この GLT-基底を求める

$G = \{f, g\}$

$$S(f, g) = x_7 f - x_1 g$$

$$= \underbrace{x_1 x_5 x_6 - x_2 x_3 x_7}_{\uparrow \text{余り}} = h$$

$$G = \{f, g, h\}$$

$$S(f, g) \text{ (済)}$$

$$S(g, h) \quad x_4 x_7 \text{ と } x_1 x_5 x_6 \text{ は互いに素}$$

$$S(f, h) = x_2 x_3 g \leftarrow f, g, h \text{ に関する余り 0}$$

$\therefore \{f, g, h\}$ は $\mathbb{I} \cap \mathbb{K}[x]$ の \mathbb{K} -基底

定理 17 (消去定理)

$$S = \mathbb{K}[x_1, \dots, x_n]$$

$$S' = \mathbb{K}[x_{i_1}, \dots, x_{i_m}] \quad (1 \leq i_1 < \dots < i_m \leq n)$$

$<$: 単項式順序 on S (on S')

G : $\mathbb{I} \cap \mathbb{K}[x]$ の \mathbb{K} -基底 $G \cap S'$ に関する GB

$<$ の条件

$$\lceil g \in G, \text{in}_<(g) \in S' \Rightarrow g \in S' \rceil$$

が満たされる。

$G \cap S'$ は $\mathbb{I} \cap S'$ の \mathbb{K} -基底に関する GB

Example 18

purely lex-order $x_1 > x_2 > \dots > x_n$

$x_i^a <_{\text{plex}} x_i^b \iff a < b$ の最も左にあるゼロでない成分が負成分で

$$S' = \mathbb{K}[x_m, x_{m+1}, \dots, x_n]$$

と \mathbb{K} の \mathbb{I} に関する消去定理が使える。

Example 19

$$\begin{cases} f_1 = x^2 + y + z - 1 = 0 \\ f_2 = x + y^2 + z - 1 = 0 \\ f_3 = x + y + z^2 - 1 = 0 \end{cases}$$

$$I = (f_1, f_2, f_3)$$

$<_{\text{plex}} : x > y > z$

$g_1 = x + y + z^2 - 1$	}	$\leftarrow x, y, z$
$g_2 = y^2 - y - z^2 + z$		$\leftarrow y, z$ のみ
$g_3 = 2yz^2 + z^4 - z^2$		
$g_4 = z^6 - 4z^4 + 4z^3 - z^2$		$\leftarrow z$ のみ

$\mathbb{K}[z]$ の GB

$$I = (g_1, \dots, g_4)$$

$$I \cap \mathbb{K}[z] = (g_4)$$

对应表

午前	午後
lex	全次数辞書式
reverse lex	全次数逆辞書式
purely lex	辞書式

9月14日(月曜日)

(準備2) 14:30~16:30

「計算機を使う観点からの導入」

野呂 正行(神戸大学大学院理学研究科・JST CREST)

濱田 龍義(福岡大学理学部・JST CREST)

[概要: グレブナー基底に関するアルゴリズムを実行できる環境として、KNOPPIX/Math 仮想マシンに習熟する。仮想マシン実行用ソフト (VMware Player、VMware Fusion)、仮想マシンのインストールを各自 PC に行ったあと、エディタの使い方などの基本操作を実習する。その後、(準備1) で学んだ諸概念を、幾つかのソフトウェアを使って実際に確認する。]

仮想マシンのインストール

野呂正行¹ 濱田龍義²

¹ 神戸大学/JST CREST

² 福岡大学/JST CREST

野呂, 濱田 仮想マシンのインストール

VMware Player/Fusion のインストール

- VMware Player (Windows 用)
時間の節約のため, 今回は配布します. 後日正式に取得してください.
 - ① Google で VMware Player を検索して, ダウンロードページに行く.
最新版は 9/13 時点で 2.5.3 である.
 - ② いくつかの質問に答える
email アドレスを含む質問があるが, 正しく入力しても, いままで実際に email が届いたことは一度もない.
 - ③ ダウンロードしたらインストール
- VMware Fusion (Mac 用)
VMware Fusion (残念ながら有料) を入手, インストールする. こちらは売り物なので, 説明書その他に従って下さい.

野呂, 濱田 仮想マシンのインストール

仮想マシンファイル

- ① KNOPPIX/Math DVD 版の ISO イメージ
knxm2008n-kobe.iso; 4GB
- ② 仮想マシン本体
次のいずれかを使う.
 - math2008n-crest-0912.exe
Windows 用自己解凍ファイル, NTFS 用 (通常はこちら)
 - math2008n-crest-0912.zip
Mac の内蔵ディスク用 (通常はこちら)
 - math2008n-crest-2G-0912.exe
Windows 用自己解凍ファイル, FAT32 用
 - math2008n-crest-2G-0912.zip
Mac の外付け FAT32 ディスク用

野呂, 濱田 仮想マシンのインストール

仮想マシンのインストール

- ① インストール先パーティションの選択
十分な空き領域 (最低 6GB) のあるものを選ぶ.
- ② インストール先ファイルシステムの確認
Windows (マイ) コンピュータ->当該ディスク右クリック->プロパティ
- ③ 仮想マシンファイルを選択したパーティションにコピー
USB メモリ, ポータブル HDD などから knxm2008n-kobe.iso と仮想マシン本体をコピーする.
- ④ Windows の場合, VMwarePlayer のインストール
VMware-Player-2.5.3-185404.exe をコピーしてインストールする.
- ⑤ 仮想マシン本体の展開
仮想マシン本体ファイルを実行して仮想マシンを展開
⇒ フォルダ math2008n-crest ができる.
- ⑥ iso ファイルを仮想マシンフォルダ内に移動する.

野呂, 濱田 仮想マシンのインストール

仮想マシンの起動, 終了

- math2008n-crest/math2008n.vmx をダブルクリック
これはメモリを 512MB 使用する設定.
ダイアログがでたら Enter を押せばよい.
- 実メモリが少ない場合には, math2008n-256M.vmx を使う (256MB 使用).
- 起動後にメモリ量を変更できる.
VMware Player->トラブルシューティング (Mac の場合仮想マシン->設定) から変更できる.
メモリがふんだんにある場合には, 適宜増やすのもよい.
- 終了は, K メニュー -> ログアウト
終了直前が出るダイアログには単に Enter を入力する.
「はい」を選ぶと, 再起動は失敗する
⇒ VMware Player -> トラブルシューティング -> パワーオフして終了. その後改めて起動すれば OK.

野呂, 濱田 仮想マシンのインストール

共有フォルダ

- ① 共有フォルダに指定するフォルダを作成
ホスト側で, 共有フォルダ用ディレクトリを作成
- ② 共有フォルダを有効にする.
仮想マシン起動後 VMware Player->共有フォルダ (Mac の場合仮想マシン ->共有フォルダ) で共有フォルダを有効にする.
- ③ 作成したディレクトリを共有フォルダに指定する.
Mac の場合, 一度デフォルトの設定を削除してから, 新規に共有フォルダを追加する.
フォルダを共有フォルダに指定してから, 名前を shared_folder に変更する.
- ④ デスクトップアイコンの変更
下部のペンギンから Mount Shared Folder を実行
実体は /mnt/hgfs/shared_folder である. シェルからアクセスする場合はこのパス名を用いる.

野呂, 濱田 仮想マシンのインストール

suspend, resume

- **suspend**
仮想マシンウィンドウを×で閉じると、現在の状態をセーブして **suspend** 状態となる。
この状態で、ホストマシンをシャットダウンすることができる。
仮想マシンフォルダ内に、一時停止のマークのついたアイコンが見える。
- **resume**
suspend 状態で、**vmx** ファイルをダブルクリックすると、**resume** する。
- 仮想マシンの移動、コピー
仮想マシンフォルダごと移動、コピーできる。

野呂, 濱田 仮想マシンのインストール

プリンタの設定 (PS プリンタの場合)

- ① ベンギン->Configure->Configure Printer を実行
- ② 追加->プリンタ/クラスを追加
- ③ バックエンド選択でリモート LPD キューを選択
- ④ LPD キュー情報でプリンタホスト、キュー名を入力
演習用プリンタ : B 棟 4 階中央の XEROX DCII-C4300
ホスト : p-418.math.kobe-u.ac.jp キュー : PS_DUP
- ⑤ プリンタ機種選択で **Postscript** プリンタを選択
- ⑥ プリンタテスト->設定
Page Size : **A4** Double-Sided Printing : **Long Edge**
Miscellaneous->GhostScript pre-filtering : **Convert to PS level2**
- ⑦ 一般情報で名前をつける
p-418 としておく。

以上により、`lpr -Pp-418 ...` でファイルが印刷できる。

野呂, 濱田 仮想マシンのインストール

数学ソフトウェアに関する文書の検索

数学ソフトウェアのマニュアル、参考書はいくつかの場所に分散している。

- /usr/share/doc
種々の文書がおかれるディレクトリ
- /usr/local/Math-ja
日本語文書がある。
knoppix-math からリンクされている。
- デスクトップの **Math-Doc-Search**
Math-Doc-Search を起動し、**Query** にキーワード (日本語 OK) を並べてサーチする。
⇒ 大抵のものを簡単に探し当てることができる。

野呂, 濱田 仮想マシンのインストール

その他

- 背景 (壁紙) の変更
背景上で右クリック -> デスクトップを設定からできる。キャラクターなしの壁紙は
/cdrom/KNOPPIX/background.jpg
- USB メモリの使用
 - ① USB メモリを挿す
ダイアログが出たら、「何もしない」を選んで OK
 - ② USB メモリのアイコンがデスクトップに現れる
クリックすれば開く
 - ③ 書き込み可にする
アイコン右クリックで **Change read/write mode**
 - ④ 取り外す前にアンマウント**Windows** 側に渡す場合には、上部のデバイスメニューから切断すればよい

野呂, 濱田 仮想マシンのインストール

Macaulay2 の起動

- \sqrt{x} メニュー、または **KNOPPIX-Math-Start** から起動する。
Konsole が起動し、その中で **Macaulay2** が起動する。
(**emacs**) を選ぶと **emacs** のバッファ内で **Macaulay2** が起動する。getting started で推奨されている使い方である。
- 端末エミュレータ (**Konsole** など) から起動する。
自分で立ち上げた端末エミュレータのシェルからコマンド **M2** を実行すると、その端末エミュレータ内で **Macaulay2** が起動する。

野呂, 濱田 仮想マシンのインストール

Macaulay2 : ヘルプその他

- ヘルプ, マニュアル
コマンド **viewHelp** を実行すると、ブラウザが起動する。最初は、
Macaulay 2 -> getting started -> a first Macaulay などをざっと眺めてみることをお勧めする。
個々のコマンドは、**index** から調べることができる。
- ファイルのロード
ファイルのロードは **load**、パッケージのロードは **loadPackage** で行う。

野呂, 濱田 仮想マシンのインストール

Asir の起動方法

- \sqrt{x} メニュー, または **KNOPPIX-Math-Start** から起動する.
(**openxm**) の方を起動すれば, 種々のライブラリファイルを自動的に読みこんで起動する.
- 端末エミュレータから起動する.
Asir 単体ではコマンドライン編集機能を持たないので, `openxm fep asir` を実行する.

Asir : ヘルプその他

- ヘルプ, マニュアル
ヘルプは `help("function")` で引ける. マニュアルはデスクトップの **Math-Doc-Search** で引くか, `helph()` コマンドでブラウザを立ち上げて **HTML** 形式のマニュアルを見るのが便利である.
- ファイルのロード
ファイルは `load` により行う. 環境変数 `ASIRLOADPATH` で指定されたディレクトリを順に探す. この値は, シェルから `openxm env` を実行すると見ることができる.

エディタ Emacs

濱田 龍義 (福岡大学/JST CREST)

平成 21 年 9 月 14 日

1 序

KNOPPIX/Math を用いると、世界中で開発されている数学ソフトウェアを利用することができます。収録されている数学ソフトウェアのうち、特に研究に活用されているシステムについて言えることですが、UNIX の基礎知識 (標準入出力、シェル) や Emacs や vi 等のエディタ (Editor) と呼ばれるアプリケーションの使い方に慣れておくと便利です。そこで、ここでは Emacs について簡単に解説します。

普段、我々が意識せずに行っている作業ですが、コンピュータに文字を入力したり、マウスを用いて絵を描いたりという操作は、すべてメモリー (主記憶) 上に確保された領域、すなわち バッファ (Buffer) に一旦データを書き込んだり、バッファからデータを読み込んだりすることで実行されます。

例えば Web メールを例に考えてみましょう。文章を入力している最中に、「下書き保存」をせずに、Web ブラウザを終了すると入力した文章は消えてしまいます。これは、入力されたデータがメモリー上のバッファに保存されているだけで、ハードディスクには保存されていないからです。「下書き保存」もしくは「送信」をマウスでクリックすることで、初めてハードディスクにデータが保存されます。

コンピュータ上でお絵書きソフトを使ったことがある方もいると思います。マウスで線を描く、色を塗るといった操作も、メモリー上に用意されたバッファに対して 2 進法の $\{0, 1\}$ データを書き込んでいます。

一般に、メモリーはデータを読み書きする速度が速いのですが高価です。ハードディスクはデータを読み書きする速度は遅いのですが、メモリーに比べれば安価にデータを保存することができます。そこで、メモリー上にバッファという緩衝地帯を設け、適当なタイミングでハードディスクのデータを読み書きすることで、コンピュータの操作性を高めています。また、メモリー上のデータはコンピュータの電源を切ると消えてしまいます。データを長期的に保存するためには、ハードディスク等の補助記憶装置が必要となります。


スクールではエディタを用いて命令を記述したテキストファイルを作成し、数学ソフトウェアに読み込むことで計算を実現します。Emacs の正式名称は GNU Emacs (ぐにゅーいーまっくす) です。フリーソフトウェア (Free Software) の概念を提唱した Richard Stallman が作成し、現在は FSF (Free Software Foundation) という組織で開発が進められフリーソフトウェアとして公開されています。Emacs はスクリーンエディタの一種ですが、Lisp というプログラミング言語によって拡張が可能であり、極めて多彩な能力を備えているため、1 つのコンピュータ環境と言っても良いでしょう。

2 GNU Emacs

2.1 引数無しで起動

この実習では Emacs が備えている豊富な機能のうち、ほんの一部だけ紹介します。既に Emacs を日常的に使いこなしている方は、軽く読み飛ばしてください。

Emacs について詳しいことを知りたい方は、「新 The UNIX Super Text[上]」の第 12 章、第 19 章を読むことで、かなりのことを学ぶことができます。さらに詳しいことを学びたいときは、第 8 章 8.6 を参照してください。参考書等も紹介されています。

まず KNOPPIX/Math 上でコンソール画面を開いてください。デスクトップ下部のメニューから左から 7 番めの黒い画面のアイコン  をシングルクリックすると、コンソールが起動します。ここでは、コンソールからコンピュータに命令を入力します。

コンソール画面を開くと

```
knoppix@Knoppix:~$
```

と表示されますが、knoppix@Knoppix:~\$ の部分を プロンプト(prompt) と呼びます。ここで ~ という記号は特別な意味を持っており、ホームディレクトリ (Home Directory) を表しています。各ユーザが自由にファイルを作成したり、計算結果を出力して保存することができる領域です。

今後、作業を円滑に進めるため、ホームディレクトリの中にディレクトリ (Windows で言うところのフォルダ) を作成して、作業を進めます。ディレクトリを作成するコマンドは mkdir です。

```
knoppix@Knoppix:~$ mkdir tutorial
```

コマンド mkdir の後ろに、新たに作成するディレクトリ名 (この場合は tutorial) を指定します。これをコマンド mkdir の 引数 (ひきすう) としてディレクトリ名を指定すると言います。コマンドでディレクトリ tutorial を作成したらカレントワーキングディレクトリを ~/tutorial に変更して Emacs を起動します。

```
knoppix@Knoppix:~$ cd tutorial
knoppix@Knoppix:~/tutorial$ emacs
```

ここでは emacs を引数無しで起動しています。Emacs を起動すると Welcome メッセージが出ます。まず一番に重要なことは、Emacs の終了の仕方です。Emacs を終了するには、C-x C-c を行います。これは、「Ctrl」キーを押しながら、2 個のキー「x」、「c」を続けて押すという動作です。

次に適当なキーを押してみましょう。すると、

```
;; This buffer is for notes you don't want to save, and for Lisp evaluation.
;; If you want to create a file, visit that file with C-x C-f,
;; then enter the text in that file's own buffer.
```

と出ます。このバッファは、今の段階では使いません。一応、簡単に訳しておく

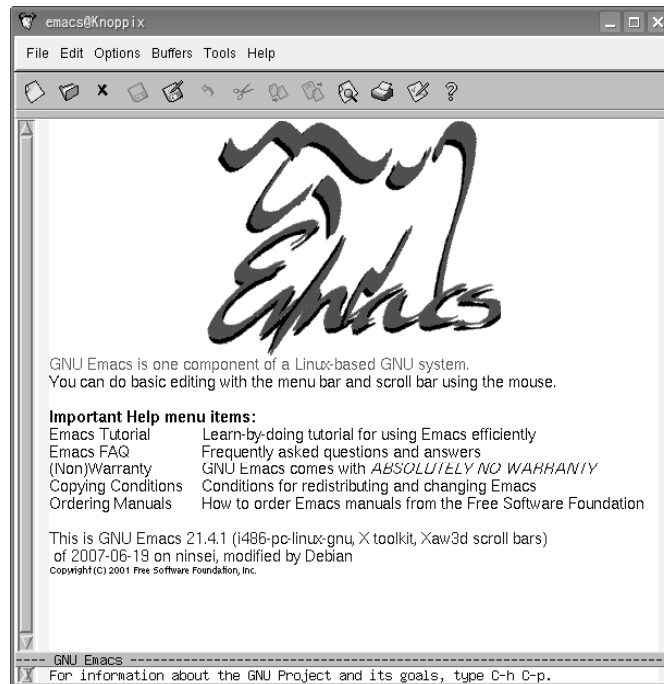


図 1 KNOPPIX/Math における Emacs の起動メッセージ

```
;; このバッファは、ちょっとしたメモや、Lisp の評価用です。
;; もし、ファイルを作成したいならば、C-x C-f を用いてファイルを開き、
;; そのファイルのバッファに文字を入力してください。
```

と書いてあります。それでは、エディタ Emacs を用いて、内容が “Emacs is a screen editor.” という 1 行だけのファイル emacs.txt を作成します。上記にも書いてあったように、まずは、ファイルを作成するためのバッファを用意します。上のメッセージにもあったように、ファイルを作成するバッファを用意する命令は C-x C-f です。「Ctrl」キーを押しながら、「x」と「f」のキーを続けて押すと、エコー領域に

```
Find file: ~/tutorial/
```

と表示されるはずですが、そこで、次のように入力して Enter キーを押します。

```
Find file: ~/tutorial/emacs.txt
```

ファイル~/tutorial/emacs.txt を作成するためのバッファが用意されました。エコー領域を見ると (New file) と表示されているはずですが、あとは入力したい文字列をバッファに入力するだけです。

まずは、1 行だけ

```
Emacs is a screen editor.
```


と入力します。行末のピリオドを入力したら「Enter」キーを押して改行してください。

Emacs では、「Ctrl」キーや「Meta」キーを用いて命令します。C-x という記号は「Ctrl」キーを押しながら「x」を押す操作です。例えば C-x o と書かれていたら、「Ctrl」キーを押しながら「x」を押し、「Ctrl」キーから指を離してから「o」を押す動作となります。

また、M-x という記号は「Meta」キーを押しながら「x」を押す操作です。しかし、一般の PC には「Meta」キーは存在しないので、「Alt」キーを「Meta」キーの代わりに用います。また、一般的には、「Esc」キーをポンと押してから指を離して「x」キーを押す操作で代用することの方が多ようです。

それでは、emacs.txt という名前が付いたバッファのデータをファイルに保存しましょう。ファイルに保存する命令は、C-x C-s です。この場合、C-s の s は save (保存する) という意味です。C-x C-s という命令を与えるとエコー領域に

```
Wrote /home/knoppix/tutorial/emacs.txt
```

というメッセージが出ます。これでメモリー上のバッファのデータがハードディスクに書き込まれました。

一旦、Emacs を終了します。終了の命令は C-x C-c でした。

それではファイルが作成されているか確認します。

```
knoppix@Knoppix:~/tutorial$ ls emacs.txt
emacs.txt
```

ファイルの大きさを調べます。ファイルの大きさが 26byte であることを確認してください。

```
knoppix@Knoppix:~/tutorial$ ls -l emacs.txt
合計 4
-rw-r--r-- 1 knoppix knoppix 26 2009-09-14 10:34 emacs.txt
```

次にファイルの内容を表示しましょう。

```
knoppix@Knoppix:~/tutorial$ cat emacs.txt
Emacs is a screen editor.
```

ここで、コマンド od を用いてみましょう。od は octal dump の省略形です。octal とは 8 進法のこと、dump とは、本来は「投げ捨てる、放り出す」などの意味ですが、コンピュータ用語では記憶されたデータを出力するという意味になります。まずは以下の表で普段使い慣れている 10 進法との対応を確認しましょう。2 進法は {0, 1}, 8 進法は {0, 1, 2, 3, 4, 5, 6, 7}, 10 進法は {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}, 16 進法は {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, a, b, c, d, e, f} という文字で数字を表す方法です。

2進法	8進法	10進法	16進法
0000	0	0	0
0001	1	1	1
0010	2	2	2
0011	3	3	3
0100	4	4	4
0101	5	5	5
0110	6	6	6
0111	7	7	7
1000	10	8	8
1001	11	9	9
1010	12	10	a
1011	13	11	b
1100	14	12	c
1101	15	13	d
1110	16	14	e
1111	17	15	f

od はオプションを付けることでファイルの内容を 8 進法だけでなく 16 進法でも表示することができます。(下の表示では、メモを書き込めるように行を空けています。)

```
[sm090001@pcs1p001-( ~/tutorial )-1008]od -Ad -tx1 emacs.txt
0000000 45 6d 61 63 73 20 69 73 20 61 20 73 63 72 65 65

0000016 6e 20 65 64 69 74 6f 72 2e 0a

0000026
```

ここで表示されているのは、ハードディスク装置に ASCII コード と呼ばれる形式で記録されている情報です。ファイル名 `emacs.txt` は、ファイル自身には含まれていないことに注意してください。左側の 6 桁の数字は保存されているデータの 1byte 単位のアドレスを 10 進法 で表しています。つまりデータに対して 0000000 から 0000025 までのアドレスが振られているということです。さらにデータは 16 進法で表されています。45='E', 6d='m', 61='a', 63='c', 73='s'... と対応しているのがわかるでしょうか? 16 進法で記述された 2 桁の数字 1 個が 1byte に相当します。全部で 26 個あることを確認してください。また、ところどころに空白を表す 20=' ' や最後が 0a='LF'、つまり「改行」を表す制御文字で終了していることにも注意してください。

2.2 起動時に引数でファイル名を指定

コマンド `emacs` は引数としてファイル名を指定すると、指定されたファイル名のファイルを編集することができます。

```
[sm090001@pcs1p001-( ~/tutorial )-1009]emacs emacs.txt
```

Emacs はファイルを開くと、ファイルの内容を表示します。

```
Emacs is a screen editor.
```

2.3 テキストの削除、挿入

それでは、screen という単語を text に修正しましょう。ファイルを開いたとき、行頭の E の字が点滅しているはずですが、キーボードで文字を入力すると、この位置に文字が表示されます。この点滅している部分を カーソル(cursor) と呼びます。カーソルは、コントロールキーを利用して移動することができます。C-f を用いて screen の n の字の次の位置まで移動したら、BS キーで screen という文字を削除し、text と入力してください。バッファが修正されて、以下ようになります。

```
Emacs is a text editor.
```

この時点でカーソルは text という単語末の t の字の次の位置にあるはずですが。

2.4 複数行の編集

今度は複数行の編集をしましょう。C-n で次の行に移動します。editor. という文字の後ろには改行の制御文字がありますので、カーソルキーの移動により、2 行目に入力することができます。

```
Emacs is a text editor.  
Emacs is a computer environment.
```

Emacs では編集機能の多くは Emacs-Lisp というプログラミング言語の一種によって実現されています。文字の入力やカーソルの移動などの編集操作は対応する Lisp 関数 を呼び出すことによって実現されています。関数名を指定して Lisp 関数を呼び出すには M-x を使います。M-x と入力すると、エコー領域にミニバッファが開いて、関数名の入力を促します。

```
M-x
```

例えば、M-x forward-char と入力すると、カーソルが次の文字へ移動します。

しかし、カーソルの移動のために、いちいち関数名を入力していたのでは面倒なので、良く使う関数に対してはキー入力が割り当てられています。「キーに Lisp 関数を結びつける」という意味で、このようなキー割り当ての事を キーバインディング(key binding) と言います。コンピュータの入力において効率を求めるのであれば、キーボードからは手を離さない方が良いことは明白です。そこで、Emacs では上の表のようなキーバインディングでカーソルを移動させることができます。

これまでに紹介したファイル操作も、それぞれ Lisp 関数をキーに結び付けたものです。

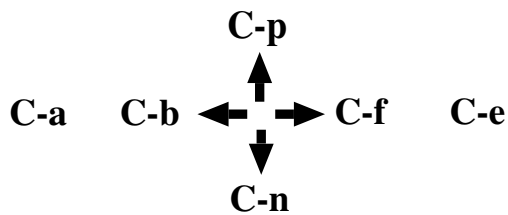


図2 Emacs のカーソル移動

キー操作	意味	関数
C-f	次の文字へ	forward-char
C-b	前の文字へ	backward-char
C-p	前の行へ	previous-line
C-n	次の行へ	next-line
C-a	行頭へ	beginning-of-line
C-e	行末へ	end-of-line

キー操作	意味	関数
C-x C-c	Emacs の終了	save-buffers-kill-emacs
C-x C-f	ファイルを開く	find-file
C-x C-s	ファイルの保存	save-buffer
M-x	Lisp 関数の呼出し	execute-extended-command

練習 1 先ほどの文章に続けて以下の文章を入力してください。

```
Emacs is a text editor.
Emacs is a computer environment.

At its core is an interpreter for Emacs Lisp,
a dialect of the Lisp programming language
with extensions to support text editing.
```

ここまで入力したら、ファイルを保存します。ファイルを保存する命令は、C-x C-s です。この命令も重要ですので、必ず忘れないようにしてください。

練習 2 C-p 等でカーソルを動かして 3 行目に 1 行挿入して

```
Emacs is a text editor.
Emacs is a computer environment.
Emacs is the extensible, customizable, self-documenting, real-time display editor.
At its core is an interpreter for Emacs Lisp,
a dialect of the Lisp programming language
with extensions to support text editing.
```

としてください。

ここでモードラインに注目してみましょう。モードラインには現在の行番号が表示されています。

練習 3 カーソルを上下に動かして行番号を確認しなさい。

Emacs では全ての編集操作にキーが割り当てられているわけではありません。例えば、行番号を表示する Lisp 関数は “what-line” です。M-x what-line と入力すると、エコー領域に行番号が表示されます。ま

た、行番号を指定して移動するには、“goto-line” という Lisp 関数を利用します。実際に

```
M-x goto-line
```

を実行すると、エコー領域にミニバッファ

```
Goto line:
```

が表示されますので、行番号を数字で入力して Enter キーを押せば、任意の行に移動することができます。

練習 4 Lisp 関数 goto-line を利用して、5 行目に移動しなさい。また、Lisp 関数 what-line を利用して、行番号をエコー領域に表示しなさい。

練習 5 ファイルを保存せずに、終了の命令 C-x C-c を実行しなさい。どのようなメッセージがエコー領域に表示されますか？

おそらく、

```
Save file /home/knoppix/tutorial/emacs.txt? (y, n, !, ., q, C-r or C-h)
```

と表示されます。このメッセージは、変更されているにも係わらず保存されていないファイル emacs.txt が存在するためです。従って、変更を保存する場合は y、変更を保存しない場合は n を入力します。ここでは y を入力して保存します。

2.5 削除ふたたび

再度、テキストファイル emacs.txt を Emacs で開きます。

```
knoppix@KNOPPIX:~/tutorial$ emacs emacs.txt &
```

命令の最後に & を付けると、バックグラウンドジョブで Emacs を実行できます。つまり、コンソールをそのまま使うことができるわけです。カーソルの位置は、最初の行の先頭の文字、「E」にあるはずですが、カーソル位置の文字を削除するには、Emacs の命令 C-d を用います。C-d を 1 回だけ押して、「E」の字を 1 字だけ消してください。

練習 6 C-d をさらに 4 回押して Emacs という文字を消しなさい。

1 行全てを削除したいときに、その回数分だけ C-d や BS を押すのは面倒です。カーソル位置から行末までの削除命令として、C-k が用意されています。ここで、C-k はカーソル位置からの削除であることに注意してください。行頭への移動 C-a と組み合わせることで、迅速に行を消すことができます。

キー操作	意味	関数
BS	カーソル位置の前の文字を削除	backward-delete-char
C-d	カーソル位置の文字の削除	delete-char
C-k	カーソル位置から行末までの削除	kill-line

それでは、複数行に渡って削除するにはどうしたら良いのでしょうか。

2.6 ポイント/マーク/リージョン

Emacs では、編集を行う際、ポイント(point)、マーク(mark)、リージョン(region) という 3 つの概念を利用します。

概念	意味
ポイント	現在のカーソル位置と前の文字との間
マーク	ユーザがバッファに 1 つだけ記憶させることができるポイント
リージョン	ポイントとマークで挟まれた部分 (矩形とは限らない)

下記のドキュメントを例に説明します。今、カーソル位置が最初の不定冠詞 “a” にあるとします。

```
Emacs is a computer environment.  
Emacs is the extensible, customizable, self-documenting, real-time display editor.
```

このとき、ポイントは空白 “ ” と “a” の間にあります。ここで、マークを設定します。マークを設定する命令は、C-SPC です。SPC はスペースキーを表しています。

カーソルを移動して、定冠詞 the の “t” まで移動します。

```
Emacs is a computer environment.  
Emacs is t he extensible, customizable, self-documenting, real-time display  
editor.
```

このとき、リージョンに含まれるのは、最後の空白も含めて以下の部分となります。

```
    a computer environment.  
Emacs is
```

Emacs では、標準の状態ではマークやリージョンは目に見える形では表示されません。教科書のコラムでは、マーク位置を確認する方法の一つとして C-x C-x を 2 度入力する方法が紹介されています。その他の方法としては、transient-mark-mode という Lisp 関数が存在し、リージョンの表示モードの On, Off を行うことができます。

リージョンを確認したら、削除してみましょう。リージョンの削除は、C-w という命令で行います。削除されたリージョンは 削除リング(kill ring) と呼ばれる領域に保存されます。削除リングに保存されたリージョンは、C-y という命令でカーソル位置に挿入できます。また、リージョンを削除するのではなく、削除リングに複製する命令 M-w もあります。

一般に C-w や C-k による文字列の削除を カット(cut)、M-w による文字列の複製を コピー(copy)、C-y による文字列の挿入を ペースト(paste) と呼び、テキストの移動を「カット&ペースト」、テキストの複製を「コピー&ペースト」と呼びます。

ところで、削除リングは、その名のとおりリング (輪) 状に削除された文字列が記憶されています。C-y の後に M-y を実行することで、順次古い削除内容を呼び出すことができます。

キー操作	意味	関数
C-w	リージョンを削除	kill-region
M-w	リージョンを削除リングに複製	kill-ring-save
C-y	削除リングのリージョンをカーソル位置に挿入	yank
M-y	削除リング内の古いデータを順番に呼び出す	yank-pop

3 編集操作のキャンセル

Emacs を使っている最中にキー操作を間違えたり、関数呼び出しを間違えたりということがあります。また、操作によっては、ミニバッファにメッセージが表示されて意味が理解できないということがあるかもしれません。その場合、操作をキャンセルする命令を行ってください。

操作をキャンセルする方法は C-g です。大変重要な命令です。場合によっては、何度か C-g を実行しないといけないかもしれませんので、注意しておいてください。

4 その他

今回紹介した機能の他に必要と思われる機能を表にまとめておきます。最近の Emacs はマウスでも操作できますが、キーボードからの操作を使いこなせれば、使いこなせるほど、今後の実習の効率が上がりますので是非挑戦してみてください。

キー操作	意味	関数
C-x i	ファイルの内容をバッファに挿入	insert-buffer
C-x C-w	ファイル名を指定して保存	write-file
C-s	前方逐次検索 (前方インクリメンタルサーチ)	isearch-forward
C-r	後方逐次検索 (後方インクリメンタルサーチ)	isearch-backward
M-%	置換	query-replace
C-x o	カーソルを他のウィンドウに移動	other-window
C-x u	Undo(編集取消)	advertised-undo
C-_	Undo(編集取消)	undo

また、Emacs 上で M-x help-with-tutorial を実行すると Emacs の簡単な案内が表示されます。こちらも一度読んでみると良いでしょう。

5 日本語の入力

たぶんスクールでは必要ありませんが、最後に Emacs 上で日本語入力を切替える方法を紹介しておきます。Emacs 上で日本語入力を切替えるには C-¥ を用いてください。

参考文献

[1] 新 The UNIX Super Text [上] 改訂増補版, 山口和紀 他著, 技術評論社

計算機を使う観点からの導入

野呂正行¹ 濱田龍義²

¹ 神戸大学/JST CREST

² 福岡大学/JST CREST

野呂, 濱田

計算機を使う観点からの導入

VMware Player/Fusion のインストール

● VMware Player (Windows 用)

- 1 Google で VMware Player を検索して、ダウンロードページに行く。
最新版は 9/3 時点で 2.5.3 である。
- 2 いくつかの質問に答える
email アドレスを含む質問があるが、正しく入力しても、いままで実際に email が届いたことは一度もない。
- 3 ダウンロードしたらインストール

● VMware Fusion (Mac 用)

VMware Fusion (残念ながら有料) を入手、インストールする。こちらは売り物なので、説明書その他に従って下さい。

野呂, 濱田

計算機を使う観点からの導入

仮想マシンファイル

1 KNOPIX/Math DVD 版の ISO イメージ

knxm2008n-kobe.iso; 4GB

2 仮想マシン本体

次のいずれかを使う。

- math2008n-crest-0912.exe
Windows 用自己解凍ファイル, NTFS 用 (通常はこちら)
- math2008n-crest-0912.zip
Mac の内蔵ディスク用 (通常はこちら)
- math2008n-crest-2G-0912.exe
Windows 用自己解凍ファイル, FAT32 用
- math2008n-crest-2G-0912.zip
Mac の外付け FAT32 ディスク用

野呂, 濱田

計算機を使う観点からの導入

仮想マシンのインストール

- 1 インストール先の選択
十分な空き領域 (余裕をみて最低 6GB) のあるパーティションを選ぶ。
- 2 インストール先ファイルシステムの確認
Windows なら
(マイ) コンピュータ->当該ディスク右クリック-> プロパティ
- 3 仮想マシンファイルを選択したパーティションにコピー
USB メモリ, SD カード, USB ポータブル HDD などから仮想マシンファイル (iso と仮想マシン本体) をコピーする。
- 4 仮想マシン本体の展開
仮想マシン本体ファイルを実行して仮想マシンを展開
⇒ フォルダ math2008n-crest ができる。
- 5 iso ファイルを仮想マシンフォルダ内に移動する。

野呂, 濱田

計算機を使う観点からの導入

仮想マシンの起動, 終了

- 仮想マシンフォルダ内の math2008n.vmx ファイルをダブルクリックする。
これはメモリを 512MB 使用する設定。
ダイアログがでたら Enter を押せばよい。
- 実メモリが少ない場合には, math2008n-256M.vmx を使う (256MB 使用)。
- 起動後にメモリ量を変更できる。
VMware Player->トラブルシューティング (Mac の場合仮想マシン->設定) から変更できる。
メモリがふんだんにある場合には, 適宜増やすのもよい。
- 終了は, K メニュー -> ログアウト
終了直前が出るダイアログには単に Enter を入力
⇒ 「はい」を選ぶと, 再起動は失敗する
⇒ VMware Player -> トラブルシューティング -> パワーオフして終了。その後改めて起動すれば OK。

野呂, 濱田

計算機を使う観点からの導入

共有フォルダ

- 1 共有フォルダに指定するフォルダを作成
ホスト側で, 共有フォルダに指定するディレクトリを作成
- 2 共有フォルダを有効にする。
仮想マシン起動後 VMware Player->共有フォルダ (Mac の場合仮想マシン ->共有フォルダ) で共有フォルダを有効にする。
- 3 作成したディレクトリを共有フォルダに指定する。
Mac の場合, 一度デフォルトの設定を削除してから, 新規に共有フォルダを追加する。
フォルダを共有フォルダに指定してから, 名前を shared_folder に変更する。
- 4 デスクトップアイコンの変更
下部のペンギンから Mount Shared Folder を実行
実体は /mnt/hgfs/shared_folder である。シエルからアクセスする場合はこのパス名を用いる。

野呂, 濱田

計算機を使う観点からの導入

suspend, resume

- **suspend**
仮想マシンウィンドウを×で閉じると、現在の状態をセーブして **suspend** 状態となる。
この状態で、ホストマシンをシャットダウンすることができる。
仮想マシンフォルダ内に、一時停止のマークのついたアイコンが見える。
- **resume**
suspend 状態で、**vmx** ファイルをダブルクリックすると、**resume** する。
- 仮想マシンの移動、コピー
仮想マシンフォルダごと移動、コピーできる。

野呂, 濱田 計算機を使う観点からの導入

プリンタの設定 (PS プリンタの場合)

- ① ペンギン->Configure->Configure Printer を実行
- ② 追加->プリンタ/クラスを追加
- ③ バックエンド選択でリモート LPD キューを選択
- ④ LPD キュー情報でプリンタホスト、キュー名を入力
ホスト : p-418.math.kobe-u.ac.jp キュー : PS_DUP
- ⑤ プリンタ機種選択で **Postscript** プリンタを選択
- ⑥ プリンタテスト->設定
Page Size : **A4** Double-Sided Printing : **Long Edge**
Miscellaneous->GhostScript pre-filtering : **Convert to PS level2**
- ⑦ 一般情報で名前をつける
p-418 としておく。

以上により, `lpr -Pp-418 ...` でファイルが印刷できる。

野呂, 濱田 計算機を使う観点からの導入

数学ソフトウェアに関する文書の検索

数学ソフトウェアのマニュアル、参考書はいくつかの場所に分散している。

- `/usr/share/doc`
種々の文書がおかれるディレクトリ
- `/usr/local/Math-ja`
日本語文書がある。
`knoppix-math` からリンクされている。
- デスクトップの **Math-Doc-Search**
Math-Doc-Search を起動し、**Query** にキーワード (日本語 OK) を並べてサーチする。
⇒ 大抵のものを簡単に探し当てることができる。

野呂, 濱田 計算機を使う観点からの導入

その他

- 背景 (壁紙) の変更
背景上で右クリック -> **デスクトップを設定** からできる。キャラクターなしの壁紙は
`/cdrom/KNOPPIX/background.jpg`
- **USB メモリ** の使用
 - ① **USB メモリ** を挿す
ダイアログが出たら、「何もしない」を選んで OK
 - ② **USB メモリ** のアイコンがデスクトップに現れる
クリックすれば開く
 - ③ 書き込み可にする
アイコン右クリックで **Change read/write mode**
 - ④ 取り外す前にアンマウント**Windows** 側に渡す場合には、上部のデバイスメニューから切断すればよい

野呂, 濱田 計算機を使う観点からの導入

Macaulay2 の起動

- \sqrt{x} メニュー、または **KNOPPIX-Math-Start** から起動する。
Konsole が起動し、その中で **Macaulay2** が起動する。
(**emacs**) を選ぶと **emacs** のバッファ内で **Macaulay2** が起動する。getting started で推奨されている使い方である。
- 端末エミュレータ (**Konsole** など) から起動する。
自分で立ち上げた端末エミュレータのシェルからコマンド **M2** を実行すると、その端末エミュレータ内で **Macaulay2** が起動する。

野呂, 濱田 計算機を使う観点からの導入

Macaulay2 : ヘルプその他

- ヘルプ, マニュアル
コマンド `viewHelp` を実行すると、ブラウザが起動する。最初は、
`Macaulay 2 -> getting started -> a first Macaulay` などをざっと眺めてみることをお勧めする。
個々のコマンドは、`index` から調べることができる。
- ファイルのロード
ファイルのロードは `load`、パッケージのロードは `loadPackage` で行う。

野呂, 濱田 計算機を使う観点からの導入

Macaulay2 : 基礎環の宣言と多項式の入力

- Macaulay2 では、基礎環を明確に宣言する必要がある。
- 係数体として、有理数体は \mathbb{Q} , 有限体 $\mathbb{F}_p = \mathbb{Z}/p\mathbb{Z}$ は \mathbb{Z}/p により入力できる。
- ある基礎環が設定されている場合、そこに含まれない不定元を使用しようとしても拒否される。
- 各多項式はそれが生成された時の環に属する。異なる環に属する多項式は、含まれる不定元が一致していても、一方を他方の環に写す必要がある。

野呂, 演田 計算機を使う観点からの導入

Macaulay2 : 基礎環および項順序

基礎環の宣言と多項式の入力

```
i1 : R=QQ[x,y,z];
i2 : f=(x+y+z)^2
      2      2      2
o2 = x  + 2x*y + y  + 2x*z + 2y*z + z
o2 : R
i3 : g=y+u
stdio:3:4:(1):[0]: error: no method for binary operator + applied to
objects:
--      y (of class R)
--      +  u (of class Symbol)
i4 : S=QQ[x,y,z,u]
i5 : f+u
stdio:5:2:(1):[0]: error: expected pair to have a method for '+'
i6 : h=(map(S,R))(f);
i7 : h+u
      2      2      2
o7 = x  + 2x*y + y  + 2x*z + 2y*z + z  + u
o7 : S
```

野呂, 演田 計算機を使う観点からの導入

Macaulay2 : 項順序

項順序は基礎環を定義するときに指定する。デフォルトは全次数逆辞書式順序であり、それ以外の順序を指定する場合には MonomialOrder により指定する。代表的なものを挙げる。

- 辞書式順序
例: $\mathbb{Q}\langle x, y, z, \text{MonomialOrder} \Rightarrow \text{Lex} \rangle$
この例は, $x > y > z$ なる辞書式順序を持つ多項式環を宣言している。
- ブロック項順序
例: $\mathbb{Z}/37\langle x, y, z, u, v, \text{MonomialOrder} \Rightarrow \{2, 3\} \rangle$
この例は, $\{x, y\} \gg \{z, u, v\}$ で、各ブロックに全次数逆辞書式を適用するブロック順序を持つ多項式環を宣言している。

多項式の先頭項は Macaulay2 では lead monomial と呼ばれ、leadMonomial で取り出せる。その係数は leadCoefficient、係数つきの先頭項は leadTerm で取り出せる。

野呂, 演田 計算機を使う観点からの導入

Macaulay2 : グレブナー基底の計算

- Macaulay2 でのイデアルの生成は $\text{ideal}(p_1, \dots, p_l)$ による。
- グレブナー基底は gb で行う。
項順序は環が知っているので引数はイデアル 1 つのみである。結果はグレブナー基底というオブジェクトで返される。
- 生成系は gens により行列 (行ベクトル) として取り出せる。
生成系 g の i 番目の要素は, g_i により取り出せる。 i は 0 から始まることに注意する。

野呂, 演田 計算機を使う観点からの導入

例 : cyclic-7

次の例は、cyclic-7 の全次数逆辞書式順序によるグレブナー基底計算である。

Macaulay2 によるグレブナー基底計算

```
i1 : R=QQ[c0,c1,c2,c3,c4,c5,c6];
i2 : I=ideal(c6*c5*c4*c3*c2*c1*c0-1,...
o2 : Ideal of R
i3 : G=gb I;
i4 : g=gens G;
      1      209
o4 : Matrix R  <--- R
i5 : g_0
o5 = | c0+c1+c2+c3+c4+c5+c6 |
      1
o5 : R
```

野呂, 演田 計算機を使う観点からの導入

Macaulay2 : イニシャルイデアルの計算

- イデアル $I \subset R$ ($R = K[x_1, \dots, x_n]$) のグレブナー基底 G の先頭項から I のイニシャルイデアル $\text{in}(I)$ が得られる。
- R/I は G の先頭項で割れない単項式全体 M で K 上張られる。
 M の元を標準単項式 (standard monomial) と呼ぶ。
- I の次元が 0 なら標準単項式は有限個である。
これは $R/\text{in}(I)$ の \mathbb{Q} -基底を与える basis により得られる。
- $\dim_K R/\text{in}(I) = \dim_K R/I$ であり、この値は、 \bar{K}^r における I の零点の重複度込みの個数に等しい。

野呂, 演田 計算機を使う観点からの導入

例：イニシャルイデアルの計算

Macaulay2によるイニシャルイデアルの計算

```
i1 : R=QQ[x,y,z];
i2 : I=ideal(x^2*y^2-z^2,x^3-y*z^2,x^2*z^4-y^2);
o2 : Ideal of R
i3 : J=ideal leadTerm I
      3 2 2 3 2 5 6 2 4
o3 = ideal (x , x y , y z , y , z , x z )
o3 : Ideal of R
i4 : dim I
o4 = 0
i5 : S=R/J
o5 = S
o5 : QuotientRing
i6 : basis S
o6 = | 1 x x2 x2y x2yz x2yz2 x2yz3 x2z x2z2 x2z3 xy xy2 xy3 xy4 ...
      |-----|
      xy2z2 xy2z3 xy2z4 xy2z5 xyz xyz2 xyz3 xyz4 xyz5 xz xz2 xz3 xz4 ...
      |-----|
      y4 y4z y3z y2z y2z2 y2z3 y2z4 y2z5 yz yz2 yz3 yz4 yz5 z z2 z3 z4 z5 |
      1 52
o5 : Matrix S <--- S
```

野呂 満田

計算機を使う観点からの導入

Macaulay2：商および剰余の計算

Macaulay2では、多項式をグレブナー基底または行列で割った商および剰余が計算できる。

- $\text{remainder}(f, g)$: f を g で割った剰余 r を返す。
- $\text{quotient}(f, g)$: f を g で割った商 q を返す。
- $\text{quotientRemainder}(f, g)$: f を g で割った商 q , 剰余 r に対し $\text{sequence}(q, r)$ を返す。
- 引数 f は行列, g はグレブナー基底または行列
- g がグレブナー基底の場合, 商は 0
- g が行列の場合, $gq + r = f$ を満たす q, r が計算される。 g がイデアルの生成系を並べた行ベクトルの場合, $q_0g_0 + \dots + q_lg_l = f$ を満たす q_0, \dots, q_l が列ベクトルとして返される。
- 剰余計算の最も簡単な応用=メンバーシップ, 包含関係
 $f \in I \Leftrightarrow f$ を I のグレブナー基底による剰余が 0
 $I \subset J \Leftrightarrow I$ の生成系の各元の, J のグレブナー基底による剰余

野呂 満田

計算機を使う観点からの導入

例：メンバーシップテスト

商および剰余の計算

```
i1 : R=QQ[x,y,z];
i2 : I=ideal(x^4*y^2+z^2-4*x*y^3*z-2*y^5*z,x^2+2*x*y^2+y^4);
i3 : G=gb I;
i4 : g=gens G;
i5 : f=y*z-x^3;
i6 : remainder(matrix{{f}},G)
o6 = | -x3+yz |
i7 : remainder(matrix{{f^2}},G)
o7 = | 2x2y3z+2x3yz+2y2z2+2xz2 |
i8 : remainder(matrix{{f^3}},G)
o8 = 0
i9 : qr=quotientRemainder(matrix{{f^3}},g);
o9 : Sequence
i10 : q=qr_0;
i11 : g^q
o11 = | -x9+3x6yz-3x3y2z2+y3z3 |
i12 : g^q-f^3
o12 = 0
```

野呂 満田

計算機を使う観点からの導入

Macaulay2：消去法

- 消去イデアルの計算
 I を多項式環 $K[Z]$ ($Z = X \cup Y, X \cap Y = \emptyset$) のイデアルとすると, $I_Y = I \cap K[Y]$ の生成系は, $X \gg Y$ なる任意の消去順序 $<$ に関する I のグレブナー基底 G に対し $G_Y = G \cap K[Y]$ により与えられる。
- G_Y は既にグレブナー基底
 G_Y は I_Y の $<_Y = <_{|K[Y]}$ に関するグレブナー基底になっている。
- 消去順序としては, 計算効率の問題から, 通常はブロック順序を使うのが望ましい。
- selectInSubring
 G から G_Y を求める。
- $\text{selectInSubring}(i, m)$
行列 m から i 番目 (この場合は $i \geq 1$) までのブロックに属する変数を含まない列のみを取り出した行列を返す。

野呂 満田

計算機を使う観点からの導入

例：消去イデアル

次の例では $I \cap \mathbb{Q}[z]$ の生成系を計算している。 $\{x, y\} \gg \{z\}$ で, 各ブロックで全次数逆辞書式順序を適用するブロック順序を設定してグレブナー基底 G を計算したあと, 変数が z のみからなる多項式を G から取り出して G_z としている。

消去イデアルのグレブナー基底の計算

```
i1 : R=QQ[x,y,z,MonomialOrder=>{2,1}];
i2 : I=ideal(x^2-z,x*y-1,x^3-x^2*y-x^2-1);
o2 : Ideal of R
i3 : G=gens gb I;
      1 3
o3 : Matrix R <--- R
i4 : Gz=selectInSubring(1,G)
o4 = | z3-3z2-z-1 |
```

野呂 満田

計算機を使う観点からの導入

Macaulay2：イデアル演算 1

$R = K[x_1, \dots, x_n]$ とする。

- イデアルの共通部分
 R のイデアル $I = \langle f_1, \dots, f_k \rangle, J = \langle g_1, \dots, g_l \rangle$ に対し, t を新しい変数とすれば

$$I \cap J = \langle tf_1, \dots, tf_k, (1-t)g_1, \dots, (1-t)g_l \rangle \cap R$$

ただし, 右辺のイデアルは $K[x_1, \dots, x_n, t]$ で考える。よって, 消去イデアル計算により共通部分が計算できる。

- イデアル商
 R のイデアル I, J に対し $I : J = \{f \mid fJ \subset I\}$ である。
 $J = \langle g_1, \dots, g_l \rangle$ なら $I : J = \bigcap_{i=1}^l I : \langle g_i \rangle$ である。 $I : \langle g \rangle$ を $I : g$ と書く。 $I : g = (I \cap \langle g \rangle) / g$ である。右辺は, $I \cap \langle g \rangle$ の生成系の各元を g で割ったもので生成されるイデアルである。よって, $I : J$ は共通部分計算により計算できる。

野呂 満田

計算機を使う観点からの導入

Macaulay2 : イdeal演算 2

● saturation

R のイdeal I, J に対し $I : J^\infty = \bigcup_{m=1}^{\infty} (I : J^m)$ である。

$J = \langle g_1, \dots, g_l \rangle$ なら $I : J = \bigcap_{i=1}^l (I : \langle g_i \rangle^\infty)$ である。 $I : \langle g \rangle^\infty$

を $I : g^\infty$ と書く。 $I : g^\infty = (I + \langle tg - 1 \rangle) \cap R$ である。ただし右辺のイdealは $K[x_1, \dots, x_n, t]$ で考える。よって saturation は共通部分計算により計算できる。

● $f \in \sqrt{I}$ の判定 (radical メンバーシップ判定)

R のイdeal $I, f \in R$ に対し、

$f \in \sqrt{I} \Leftrightarrow I + \langle tf - 1 \rangle = K[x_1, \dots, x_n, t]$ である。

$I + \langle tf - 1 \rangle = K[x_1, \dots, x_n, t]$ は $I + \langle tf - 1 \rangle$ の (任意項順序に関する) 簡約グレブナー基底が $\{1\}$ であることと同値だから、 $f \in \sqrt{I}$ か否かはグレブナー基底を計算することで判定できる。

野呂, 濱田 計算機を使う観点からの導入

例 : イdeal演算

イdealの演算

```

i1 : R=QQ[x,y];
i2 : I=ideal(x^4-y^5,x^3-y^7);
o2 : Ideal of R
i3 : I1=quotient(I,x)
      5 4 3 2 2 7 2 3
o3 : ideal (y - x , x y - x , x - x y )
o3 : Ideal of R
i4 : I2=quotient(I,x^2)
      2 2 5 4 6 3
o4 : ideal (x y - x , y - x , x - x y )
i5 : I3=quotient(I,x^3)
      2 5 4 5 3
o5 : ideal (x^2 y - 1 , y - x , x - y )
i6 : I4=quotient(I,x^4)
      2 5 4
o6 : ideal (x^2 y - 1 , y - x )
i7 : J=saturate(I,x)
      2 5 4 5 3
o7 : ideal (x^2 y - 1 , y - x , x - y )
i8 : I3==I4
o8 : true
i9 : I2==I3
o9 : false

```

野呂, 濱田 計算機を使う観点からの導入

例 : radical メンバーシップ

radical メンバーシップ判定

```

i1 : R=QQ[t,x,y,z];
i2 : I=ideal(x^4*y^2+z^2-4*x*y^3*z-2*y^5*z,
      x^2+2*x*y^2+y^4);
o2 : Ideal of R
i3 : f=y*z-x^3;
i4 : gens gb (I+ideal(t*f-1))
o4 = | 1 |

```

グレブナー基底が $\{1\}$ なので、 $f \in \sqrt{I}$ と判定できる。

野呂, 濱田 計算機を使う観点からの導入

Asir の起動方法

- \sqrt{x} メニュー, または **KNOPPIX-Math-Start** から起動する。
(openxm) の方を起動すれば, 種々のライブラリファイルを自動的に読みこんで起動する。
- 端末エミュレータから起動する。
Asir 単体ではコマンドライン編集機能を持たないので, openxm fep asir を実行する。

野呂, 濱田 計算機を使う観点からの導入

Asir : ヘルプその他

- ヘルプ, マニュアル
ヘルプは help("function") で引ける。マニュアルはデスクトップの **Math-Doc-Search** で引くか, help() コマンドでブラウザを立ち上げて HTML 形式のマニュアルを見るのが便利である。
- ファイルのロード
ファイルは load により行う。環境変数 ASIRLOADPATH で指定されたディレクトリを順に探す。この値は, シェルから openxm env を実行すると見ることができる。

野呂, 濱田 計算機を使う観点からの導入

Asir : 多項式の入力

- Asir ではアルファベット小文字で始まり, アルファベット, 数字, _ (アンダースコア) からなる文字列が不定元である。
- 入力された多項式は再帰表現により保持されている。再帰表現とは, 多項式を, 主変数に関する一変数多項式として表現するもので, 係数は, 主変数を含まない多項式である。

多項式の入力

```

[1518] F=(x+y+z)^2;
x^2+(2*y+2*z)*x+y^2+2*z*y+z^2
[1519] G=F+u;
x^2+(2*y+2*z)*x+y^2+2*z*y+z^2+u

```

野呂, 濱田 計算機を使う観点からの導入

Asir : 分散表現多項式と項順序

- グレブナー基底に関連する計算は、分散表現で行われる。グレブナー基底関連計算を行う場合、暗黙あるいは明示的に分散表現への変換を行う。
- グレブナー基底関連計算など、項順序が必要が計算の都度、項順序を指定する必要がある。
- **Asir** においては、項順序は変数順序と項順序型により指定される。
変数順序は不定元を並べたリストで表現する。
この順序は単項式を指数ベクトルで表示する場合の各指数のインデックスを決める。
例えば、変数順序が $[x, y, z, u, v, w]$ で与えられた場合、 $x^a y^b z^c u^d v^e w^f$ は (a, b, c, d, e, f) で表示される。

野呂 演田 計算機を使う観点からの導入

Asir : 項順序型の設定

変数リストに対し、次のような項順序型が設定できる。

- 単純な項順序型
 - 0 : 全次数逆辞書式順序
 - 1 : 全次数辞書式順序
 - 2 : 辞書式順序
- ブロック項順序型
 - $[[O_1, n_1], [O_2, n_2], \dots, [O_l, n_l]]$ なるリストのリスト
 - 典型例: $[[0, n_1], [0, n_2]]$: 先頭の n_1 変数を消去するための消去順序
 - ① 変数リストを左から n_1, n_2, \dots, n_l ($n_1 + \dots + n_l = n$) ずつのブロックに分ける。
 - ② i 番目のブロックに単純項順序型 O_i を適用する
 - ③ 1 番目のブロックから、大小が決まるまで順に行う。

野呂 演田 計算機を使う観点からの導入

Asir : 分散表現多項式の操作

- `dp_ord(Ord)`
項順序型の設定。項順序型は関数の引数として与える場合もある。
- `dp_ptod(F, V)`
設定されている項順序型、変数順序 V で定まる項順序で、多項式 F を分散表現に変換する。
- `dp_ht`
先頭項 (係数 1) の取り出し
- `dp_hc`
先頭係数の取り出し
- `dp_hm`
係数つきの先頭項の取り出し

用語は初期に使われていたものを採用しており、最近の用法と異なることに注意された。

野呂 演田 計算機を使う観点からの導入

例 : 分散表現多項式の演算

分散表現への変換、演算

```
[1532] F=x^2*y+y^3*z+x*x*z+x+1;
y*x^2+(z+1)*x+z*y^3+1
[1533] dp_ord(0)$
[1534] DF0=dp_ptod(F, [x, y, z]);
(1)*<<0, 3, 1>>+(1)*<<2, 1, 0>>+(1)*<<1, 0, 1>>+(1)*<<1, 0, 0>>
+(1)*<<0, 0, 0>>
[1535] dp_ord(2)$
[1536] DF2=dp_ptod(F, [x, y, z]);
(1)*<<2, 1, 0>>+(1)*<<1, 0, 1>>+(1)*<<1, 0, 0>>+(1)*<<0, 3, 1>>
+(1)*<<0, 0, 0>>
[1537] G=F+u;
y*x^2+(z+1)*x+z*y^3+u+1
[1538] DG=dp_ptod(G, [u, x, y, z]);
(1)*<<1, 0, 0, 0>>+(1)*<<0, 2, 1, 0>>+(1)*<<0, 1, 0, 1>>
+(1)*<<0, 1, 0, 0>>+(1)*<<0, 0, 3, 1>>+(1)*<<0, 0, 0, 0>>
[1539] dp_ht(DG);
(1)*<<1, 0, 0, 0>>
```

野呂 演田 計算機を使う観点からの導入

Asir : グレブナー基底の計算

gr をロードしておく。(KNOPPIX/Math では不要)

Plist : イdealを表す多項式リストである。
Vlist : 変数リスト
Ord : 項順序型

- `nd_gr(Plist, Vlist, Char, Ord)`
 $\langle Plist \rangle$ の簡約グレブナー基底を計算する。
 $Char = 0$ のとき有理数体係数,
 $Char$ が素数のとき有限体 \mathbb{F}_{Char} 上で計算する。
結果は多項式のリストである。リスト G の i 番目の要素は $G[i]$ (i は 0 から始まる) で取り出せる。
- `nd_gr_trace(Plist, Vlist, Homo, Prime, Ord)`
 $\langle Plist \rangle \subset \mathbb{Q}[\langle Vlist \rangle]$ の簡約グレブナー基底を計算する。
 $Prime$ は 1 を指定しておく。
 $Homo$ が 1 のとき、斉次化を経由して計算する。
 $Homo$ が 0 のとき、斉次化を経由しないで計算する。
ほとんどの場合 $Homo = 1$ が安全

野呂 演田 計算機を使う観点からの導入

例 : グレブナー基底計算

Asir によるグレブナー基底計算

```
[1517] load("cyclic")$
[1527] C=cyclic(7);
[c6*c5*c4*c3*c2*c1*c0-1, ...]
[1528] V=vars(C);
[c0, c1, c2, c3, c4, c5, c6]
[1529] nd_gr(C, V, 31991, 0)$
...
2.016sec + gc : 0.072sec(2.089sec)
[1530] nd_gr(C, V, 0, 0)$
(5分待つて中断)
[1530] G=nd_gr_trace(C, V, 1, 1, 0)$
...
19.54sec + gc : 5.428sec(25.02sec)
[1531] G[0];
((238539226659020007130662*c6*c4-...
[1532] length(G);
209
nd_gr : 係数膨張のため計算が進まなくなる。
nd_gr_trace * Homo = 1 : 25秒で計算が終了する。
```

野呂 演田 計算機を使う観点からの導入

Asir : イニシャルイデアルの計算

- グレブナー基底を計算する.
 - 基底の各元を `dp_ptod` で分散表現に変換
 - `dp_ht` で先頭項を取り出す
 - 必要があれば, `dp_dtop` で再帰表現に戻す
- 0 次元 \Rightarrow `dp_mbase` により, 標準単項式全体を計算できる.

Asir によるイニシャルイデアルの計算

```
[1517] B=[x^2*y^2-z^2,x^3-y^2z^2,x^2*z^4-y^2];
[1518] V=[x,y,z]$
[1519] G=nd_gr(B,V,0,0);
[1520] D=map(dp_ptod,G,V)$ H=map(dp_ht,D)$
[1521] [1522] map(dp_dtop,H,V);
[1523] map(dp_dtop,dp_mbase(H),V);
[1524] length(@@);
52
```

野呂, 演田 計算機を使う観点からの導入

Asir : 剰余計算

`p_nf` : 剰余の分母を払って整数係数で返す
剰余が 0 かどうかの判定に用いる.
`p_true_nf` : `[num,den]` なるリストを返す
`num/den` が真の剰余となる.

剰余計算

```
[1517] B=[u2*u0-2*u2+3,(2*u1-1)*u0^2-u0-2*u2,
2*u1^3+u2+4]$
[1518] V=[u0,u1,u2]$
[1519] G=nd_gr(B,V,0,0);
[1520] Q=p_nf(u0^5+u1^5+u2^5,G,V,0);
2851262910*u2^3+30078832770*u2^2+(22194374760*u1-...
[1521] QR=p_true_nf(u0^5+u1^5+u2^5,G,V,0);
[2851262910*u2^3+30078832770*u2^2+... ,35373600]
```

野呂, 演田 計算機を使う観点からの導入

Asir : 消去法: $I_Y = I \cap K[Y]$ の生成系の計算

- ブロック順序によるグレブナー基底を使う.
有理数体上で計算する場合には `nd_gr_trace` を
`Home = 1` で使う.
- の消去順序グレブナー基底 G から I_Y のグレブナー基底
 G_Y を取り出す.
`elimination` (ライブラリ `primdec_mod` に定義されて
いるがマニュアルにはない) を使う.

消去イデアルの計算

```
[1518] load("primdec_mod")$
[1664] B=[u2*u0-2*u2+3,(2*u1-1)*u0^2-u0-2*u2,2*u1^3+u2+4]$
[1665] V=[u0,u1,u2]$
[1666] G1=nd_gr_trace(B,V,1,1,[[0,2],[0,1]])$
[1667] elimination(G1,[u2]);
[8*u2^9+72*u2^8+292*u2^7-2036*u2^6-198*u2^5+20682*u2^4-...]
```

野呂, 演田 計算機を使う観点からの導入

Asir : 最小多項式の計算

`minipoly(G, V, Ord, F, T) (in gr)`

- 有理数体係数多項式環の 0 次元イデアル $I = \langle G \rangle$ およ
び多項式 F に対し, $m(f) \in I$ を満たすような 0 でない
最小次数の多項式 $m(T)$ を計算する.
- G は項順序 (V, Ord) でのグレブナー基底, F は多項式,
 T は V に含まれない変数.

katsura-7 での $u7$ の最小多項式の計算

```
[1518] load("katsura")$
[1522] B=katsura(7)$
[1523] V=[u0,u1,u2,u3,u4,u5,u6,u7]$
[1524] G=nd_gr_trace(B,V,1,1,0)$
[1525] minipoly(G,V,0,u7,t)$
[1526] deg(@@,t);
128
```

消去順序グレブナー基底計算と比較してみるとおもしろい.

野呂, 演田 計算機を使う観点からの導入

Asir : 0 次元イデアルの項順序変換

辞書式順序グレブナー基底を Buchberger アルゴリズムで
直接計算するのは一般に効率が大変悪い.

\Rightarrow 項順序変換が有効

`tolex(G, V, Ord, W) (in gr)`

0 次元イデアルの項順序 (V, Ord) でのグレブナー基底 G から,
項順序 (W, lex) のグレブナー基底を計算する.

次の例は, `katsura-7` の辞書式順序グレブナー基底を項順序
変換で計算したものである.

項順序変換による辞書式順序グレブナー基底の計算

```
[1523] V=[u0,u1,u2,u3,u4,u5,u6,u7]$
[1524] G=nd_gr_trace(katsura(7),V,1,1,0)$
2.676sec + gc : 1.356sec(4.032sec)
[1525] G2=tolex(G,V,0,V)$
279.5sec + gc : 57.68sec(337.5sec)
```

野呂, 演田 計算機を使う観点からの導入

Asir : イデアル演算

イデアルの共通部分, イデアル商, `saturation` を計算する関
数は, ライブラリのおちこちで定義され使われている. (例:
`primdec`)

マニュアルに書かれていないので, 消去イデアル計算を用い
てこれらを実装してみると, よい練習になるであろう.

野呂, 演田 計算機を使う観点からの導入

練習問題

- ① 以下の各項を行う方法を, Macaulay2, Asir それぞれについて調べよ.
 - ① ファイルに結果を書き出す.
 - ② 繰り返しを行う.
 - ③ 多項式の因数分解を行う.
- ② $\mathbb{Q}[x, y, z]$ のイデアル

$$I = \langle x^2 + z, xy + y^2 + z, xz - y^3 - 2yz, y^4 + 3y^2z + z^2 \rangle,$$

$$J = \langle x^2 + z, xy + y^2 + z, x^3 - yz \rangle$$
 の包含関係を調べよ.
- ③ $f_1 = 3x^2yz^2 + 3z + (-2x + 2)y + 2x,$
 $f_2 = 3yz^5 + (-xy^2 + 2)z - 2y^4 + 2y,$
 $f_3 = xy^3z^3 - 2yz^2 - z - 2y + x^2$ に対し,
 $I = \langle f_1, f_2, f_3 \rangle \subset \mathbb{Q}[x, y, z]$ とおく.
 - ① $\dim I = 0$ を確かめよ.
 - ② $\dim_{\mathbb{Q}} \mathbb{Q}[x, y, z]/I$ を求めよ.
 - ③ I の $x > y > z$ なる辞書式順序での簡約グレブナー基底が $\{g_0(z), x - g_1(z), y - g_2(z)\}$ という形であることを確かめよ.

野呂, 濱田

計算機を使う観点からの導入

練習問題 (つづき)

- ① $f_1 = x^2 + y^2 + z^2 - 9, f_2 = 3x^2 - y^2z, f_3 = x^2z - 2y^2 + 2$ に対し, $f_1 = f_2 = f_3 = 0$ を満たす $(x, y, z) \in \mathbb{C}^3$ を全て求めよ.
- ② $\alpha = 3^{\frac{1}{3}}, \beta = 5^{\frac{1}{3}}$ とする.
 - ① $\alpha + \beta$ の \mathbb{Q} 上の最小多項式を求めよ.
 - ② $\frac{1}{\alpha + \beta}$ を α, β の有理数係数多項式で表せ.
- ③ (Asir でのプログラミング経験がある人向け) イデアルの共通部分, イデアル商, saturation の計算および radical メンバーシップを判定する関数を記述せよ.

野呂, 濱田

計算機を使う観点からの導入

参考書

- W. Adams, P. Loustaunau, An Introduction to Gröbner Bases. Graduate Studies in Mathematics, Vol. 3, AMS (1994).
- D. Cox, J. Little, D. O'Shea, Using Algebraic Geometry. GTM Vol. 185, Springer (2005).
- D. Eisenbud, D. Grayson, M. Stillman, B. Sturmfels (Eds.), Computations in Algebraic Geometry with Macaulay 2. Algorithms and Computation in Mathematics 8, Springer-Verlag (2000).
- G.-M. Greuel, G. Pfister, A Singular Introduction to Commutative Algebra. Springer (2007).
- M. Kreuzer, L. Robbiano, Computational Commutative Algebra 1. Springer (2008).

野呂, 濱田

計算機を使う観点からの導入

グレブナー基底：計算機を使う観点からの導入

野呂 正行 (神戸大学大学院理学研究科・JST CREST)

濱田 龍義 (福岡大学理学部・JST CREST)

平成 21 年 9 月 14 日

1 各種ソフトウェアの準備

1.1 VMware Player または VMware Fusion のインストール

1.1.1 VMware Player のインストール

Windows の場合, VMware Player (無料) を入手, インストールする. Google で VMware Player を検索して, ダウンロードページに行く. 最新版は 9/3 時点で 2.5.3 である. 入手の際に, email アドレスを含む質問があるが, 正しく入力しても, いままで実際に email が届いたことは一度もない. ダウンロードできたらあとはインストールするだけである.

1.1.2 VMware Fusion のインストール

Intel Mac の場合, VMware Fusion (残念ながら有料) を入手, インストールする. こちらは売り物なので, 説明書その他に従って下さい.

1.2 仮想マシンのインストール

仮想マシンは, KNOPPIX/Math DVD 版の ISO イメージ (knxm2008n-kobe.iso; 4GB) と, 仮想マシン本体 (math2008n-crest フォルダ) からなる. ISO イメージは 4GB 弱のファイルである. 仮想マシン本体は, Windows 用の自己解凍 exe ファイル (math2008n-crest.exe 160MB) または Mac 用 zip ファイル (+verb+math2008n-crest.zip+; 240MB) を用いる.

外付け HDD を購入したままの状態を使っている場合, ファイルシステムが FAT32 の場合がある. また, Windows の C: 以外のパーティションも, FAT32 の場合がある. この場合, 仮想 HDD が 2GB ファイルに分割されている math2008n-crest-2G.exe または math2008n-crest-2G.zip を使う必要がある.

1. 仮想マシン本体の展開

配布 DVD, USB メモリ, SD カード, USB ポータブル HDD などのうち, 都合のよいものから各自の PC の HDD に仮想マシン本体を展開する. 展開するパーティションは, 十分な空き領域のあるものを選ぶこと.

2. ISO イメージのコピー

これも, なんらかのメディアから, 仮想マシンフォルダ内にコピーする.

1.3 仮想マシンの起動

仮想マシンフォルダ内の math2008n.vmx ファイルをダブルクリックする。こちらはメモリを 512MB 使用する設定であるが、もし実メモリが少ない場合には、math2008n-256M.vmx を使うこともできる。VMware Player->トラブルシューティング (Mac の場合仮想マシン->設定) から適当な大きさに変更することができる。メモリがふんだんにある場合には、適宜増やすのもよい。

1.4 共有フォルダ

デフォルトでは共有フォルダは無効になっている。仮想マシンが起動したあと、Windows または Mac 側で、共有フォルダに指定するディレクトリを作成し、VMware Player->共有フォルダ (Mac の場合 仮想マシン->共有フォルダ) で共有フォルダを有効にし、作成したディレクトリを共有フォルダに指定する。そのあと、仮想マシンの下部のペンギンから Mount Shared Folder を実行すると、デスクトップの shared_folder アイコンがフォルダの形に変わる。このフォルダに KNOPPIX 側からファイルをドロップすると、ホスト側の対応するディレクトリにコピーされたことになる。逆も同様である。実体は /mnt/hgfs/shared_folder である。シェルからアクセスする場合はこのパス名を用いる。

Mac の場合、一度デフォルトの設定を削除してから、新規に共有フォルダを追加する必要がある。この場合、実在のフォルダを共有フォルダに指定してから、名前をクリックして shared_folder に変更する。

1.5 suspend, resume

仮想マシンウィンドウを x で閉じると、現在の状態をセーブして suspend 状態となる。この状態で、ホストマシンをシャットダウンすることができる。この状態では、仮想マシンフォルダ内に、一時停止のマークのついたアイコンが見える。この状態で、vmx ファイルをダブルクリックすると、resume する。

1.6 プリンタの設定

ペンギン->Configure->Configure Printer を実行すると、プリンタ設定のためのダイアログが現れる。PS プリンタを設定する場合には次の手順を実行する。

1. 追加->プリンタ/クラスを追加
2. バックエンド選択でリモート LPD キューを選択
3. LPD キュー情報でプリンタホスト、キュー名を入力
実習で使用する B 棟 4 階のプリンタの場合
ホスト : p-418.math.kobe-u.ac.jp
キュー : PS_DUP
4. プリンタ機種選択で Postscript プリンタを選択

5. プリンタテスト->設定

Page Size : A4

Double-Sided Printing : Long Edge

Miscellaneous->GhostScript pre-filtering : Convert to PS level2

6. 一般情報で名前をつける

p-418 としておく.

以上により, `lpr -Pp-418 ...` でファイルが印刷できる.

1.7 数学ソフトウェアに関する文書の検索

数学ソフトウェアのマニュアル, 参考書はいくつかの場所に分散している. `/usr/share/doc` は, 種々の文書がおかれるディレクトリである. また, 日本語文書は `/usr/local/Math-ja` におかれていて, デスクトップの `knoppix-math` からリンクされている. これらの中から目的のものを探するのは骨が折れが, デスクトップの `Math-Doc-Search` を使えば, 大抵のものは簡単に探し当てることができる. 使い方は簡単で, `Math-Doc-Search` を起動し, `Query` にキーワード (日本語 OK) を並べてサーチするだけである.

2 Macaulay2 によるグレブナー基底の計算

2.1 Macaulay2 について

2.1.1 起動方法

- \sqrt{x} メニュー, または `KNOPPIX-Math-Start` から起動する.

`Konsole` が起動し, その中で `Macaulay2` が起動する. (`emacs`) を選ぶと `emacs` のバッファ内で `Macaulay2` が起動する. `getting started` で推奨されている使い方である.

- 端末エミュレータ (`Konsole` など) から起動する.

自分で立ち上げた端末エミュレータのシェルからコマンド `M2` を実行すると, その端末エミュレータ内で `Macaulay2` が起動する.

2.1.2 ヘルプ, マニュアル

コマンド `viewHelp` を実行すると, ブラウザが起動する. 最初は,

```
Macaulay 2 -> getting started -> a first Macaulay 2 session
```

などをざっと眺めてみることをお勧めする. 個々のコマンドは, `index` から調べることができる.

2.1.3 ファイルのロード

ファイルのロードは `load`, パッケージのロードは `loadPackage` で行う.

2.2 基礎環および項順序

2.2.1 基礎環の宣言と多項式の入力

Macaulay2 では、基礎環を明確に宣言する必要がある。係数体として、有理数体は \mathbb{Q} 、有限体 $\mathbb{F}_p = \mathbb{Z}/p\mathbb{Z}$ は \mathbb{Z}/p により入力できる。ある基礎環が設定されている場合、そこに含まれない不定元を使用しようとしても拒否される。各多項式はそれが生成された時の環に属するため、異なる環に属する多項式は、たとえ含まれる不定元が一致していても、一方を他方の環に写す (map を用いる) 必要がある。

基礎環の宣言と多項式の入力

```
i1 : R=QQ[x,y,z];
i2 : f=(x+y+z)^2
      2      2      2
o2 = x  + 2x*y + y  + 2x*z + 2y*z + z
o2 : R
i3 : g=y+u
stdio:3:4:(1):[0]: error: no method for binary operator + applied to
objects:
--          y (of class R)
--      +   u (of class Symbol)

i4 : S=QQ[x,y,z,u]
i5 : f+u
stdio:5:2:(1):[0]: error: expected pair to have a method for '+'
i6 : h=(map(S,R))(f);
i7 : h+u
      2      2      2
o7 = x  + 2x*y + y  + 2x*z + 2y*z + z  + u
o7 : S
```

2.2.2 項順序

項順序は基礎環を定義するときに指定する。デフォルトは全次数逆辞書式順序であり、それ以外の順序を指定する場合には `MonomialOrder` により指定する。代表的なものを挙げる。

- 辞書式順序

例: `QQ[x,y,z,MonomialOrder=>Lex]`

この例は、 $x > y > z$ なる辞書式順序を持つ多項式環を宣言している。

- ブロック項順序

例: `ZZ/37[x,y,z,u,v,MonomialOrder=>{2,3}]`

この例は、 $\{x,y\} \gg \{z,u,v\}$ で、各ブロックに全次数逆辞書式を適用するブロック順序を持つ多項式環を宣言している。

多項式の先頭項は Macaulay2 では lead monomial と呼ばれ, leadMonomial で取り出せる. その係数は leadCoefficient, 係数付きの先頭項は leadTerm で取り出せる.

2.3 グレブナー基底の計算

本節では, イデアルの入力およびグレブナー基底計算について述べる. Macaulay2 でのイデアルの生成は $\text{ideal}(p_1, \dots, p_l)$ による. グレブナー基底は gb で行う. この場合, 項順序は環が知っているので引数はイデアル 1 つのみである. 結果はグレブナー基底というオブジェクトで返される. 生成系は gens により行列 (行ベクトル) として取り出せる.

次の例は, cyclic-7 の全次数逆辞書式順序によるグレブナー基底計算である. 計算方法の制御を行うためのスイッチが多数用意されているが, 特に指定を行わなくても比較的高速に計算が終了する. gens により生成系が 1×209 行列 g として得られる. その i 番目の要素は, g_i により取り出せる. i は 0 から始まることに注意する.

———— Macaulay2 によるグレブナー基底計算 ————

```
i1 : R=QQ[c0,c1,c2,c3,c4,c5,c6];
i2 : I=ideal(c6*c5*c4*c3*c2*c1*c0-1,...
o2 : Ideal of R
i3 : G=gb I;
i4 : g=gens G;
           1      209
o4 : Matrix R  <--- R
i5 : g_0
o5 = | c0+c1+c2+c3+c4+c5+c6 |
      1
o5 : R
```

2.4 基本的な応用

2.4.1 イニシャルイデアルの計算

イデアル $I \subset R (R = K[x_1, \dots, x_n])$ のグレブナー基底 G の先頭項から I のイニシャルイデアル $\text{in}(I)$ が得られる. R/I は G の先頭項で割れない単項式全体 M で K 上張られる. M の元を標準単項式 (standard monomial) と呼ぶ. 次の例では, まず I のイニシャルイデアル J の計算している. 実際には, グレブナー基底の計算が行われ, グレブナー基底の先頭項で生成されるイデアルとして J が得られる. I の次元を調べると 0 なので, 標準単項式は有限個である. これは R/J の \mathbb{Q} -基底を与える basis により得られる. $\dim_K R/J = \dim_K R/I$ であり, この値は, \bar{K}^n における I の零点の重複度込みの個数に等しい.

```

i1 : R=QQ[x,y,z];
i2 : I=ideal(x^2*y^2-z^2,x^3-y*z^2,x^2*z^4-y^2);
o2 : Ideal of R
i3 : J=ideal leadTerm I
          3  2 2   3 2   5   6   2 4
o3 = ideal (x , x y , y z , y , z , x z )
o3 : Ideal of R
i4 : dim I
o4 = 0
i5 : S=R/J
o5 = S
o5 : QuotientRing
i6 : basis S
o6 = | 1 x x2 x2y x2yz x2yz2 x2yz3 x2z x2z2 x2z3 xy xy2 xy3 xy4 ...
----- ...
xy2z2 xy2z3 xy2z4 xy2z5 xyz xyz2 xyz3 xyz4 xyz5 xz xz2 xz3 xz4 ...
----- ...
y4 y4z y3z y2z y2z2 y2z3 y2z4 y2z5 yz yz2 yz3 yz4 yz5 z z2 z3 z4 z5 |
          1          52
o5 : Matrix S <--- S

```

2.4.2 商および剰余の計算

Macaulay2 では、多項式をグレブナー基底または行列 (イデアルの場合、生成系を並べた行ベクトル) で割った商および剰余が計算できる。関連する関数は以下の通りである。

- $\text{remainder}(f, g)$: f を g で割った剰余 r を返す。
- $\text{quotient}(f, g)$: f を g で割った商 q を返す。
- $\text{quotientRemainder}(f, g)$: f を g で割った商 q , 剰余 r に対し $\text{sequence}(q, r)$ を返す。

引数 f は行列, g はグレブナー基底または行列である。 g がグレブナー基底の場合, 商は 0 が返されるようである。 g が行列の場合, $gq + r = f$ を満たす q, r が計算される。例えば g がイデアルの生成系を並べた行ベクトルの場合, $q_0g_0 + \dots + q_lg_l = f$ を満たす q_0, \dots, q_l が列ベクトルとして返される。剰余計算の最も簡単な応用として, イデアル I, J に対する $I \subset J$ のテストがある。これは, J の任意項順序でのグレブナー基底 G を計算しておけば, I の生成系の各元の G による剰余が 0 となることを確かめることに帰着される。

```

i1 : R=QQ[x,y,z];
i2 : I=ideal(x^4*y^2+z^2-4*x*y^3*z-2*y^5*z,x^2+2*x*y^2+y^4);
o2 : Ideal of R
i3 : G=gb I;
i4 : g=gens G;
      1      3
o4 : Matrix R <--- R
i5 : f=y*z-x^3;
i6 : remainder(matrix{{f}},G)
o6 = | -x3+yz |
      1      1
o6 : Matrix R <--- R
i7 : remainder(matrix{{f^2}},G)
o7 = | 2x2y3z+2x3yz+2y2z2+2xz2 |
      1      1
o7 : Matrix R <--- R
i8 : remainder(matrix{{f^3}},G)
o8 = 0
      1      1
o8 : Matrix R <--- R
i9 : qr=quotientRemainder(matrix{{f^3}},g);
o9 : Sequence
i10 : q=qr_0;
      3      1
o10 : Matrix R <--- R
i11 : g*q
o11 = | -x9+3x6yz-3x3y2z2+y3z3 |
      1      1
o11 : Matrix R <--- R
i12 : g*q-f^3
o12 = 0

```

この例では、イデアル I による f, f^2, f^3 の剰余を計算して、 $f^3 \in I$ を示している。 f^3 を G で割ることで商が得られる。さらに得られた商 q を G の生成系に掛けることで、 $f^3 = gq$ が確かめられる。この場合には $f \notin I$ だが $f^3 \in I$ である。よって $f \in \sqrt{I}$ が示せたことになる。このやり方では、 $f \notin \sqrt{I}$ を示すことはできない。 $f \in \sqrt{I}$ の判定法は 2.4.4 節で説明する。

2.4.3 消去法

I を多項式環 $K[Z]$ ($Z = X \cup Y, X \cap Y = \emptyset$) のイデアルとすると、 $I_Y = I \cap K[Y]$ の生成系は、 $X \gg Y$ なる任意の消去順序 $<$ に関する I のグレブナー基底 G に対し $G_Y = G \cap K[Y]$ により与えられる。さらに、 G_Y は I_Y の $<_Y = <|_{K[Y]}$ に関するグレブナー基底になっている。消去順序としては辞書式順序も使えるが、計算効率の問題から、通常はブロック順序を使うのが望

ましい。 G を計算した後 G_Y を求めるには `selectInSubring` を用いる。

次の例では $I \cap \mathbb{Q}[z]$ の生成系を計算している。 $\{x, y\} \gg \{z\}$ で、各ブロックで全次数逆辞書式順序を適用するブロック順序を設定してグレブナー基底 G を計算したあと、変数が z のみからなる多項式を G から取り出して G_z としている。 `selectInSubring(i, m)` は、行列 m から i 番目 (この場合は $i \geq 1$) までのブロックに属する変数を含まない列のみを取り出した行列を返す。

消去イデアルのグレブナー基底の計算

```

i1 : R=QQ[x,y,z,MonomialOrder=>{2,1}];

i2 : I=ideal(x^2-z,x*y-1,x^3-x^2*y-x^2-1);
o2 : Ideal of R
i3 : G=gens gb I;
      1      3
o3 : Matrix R <--- R
i4 : Gz=selectInSubring(1,G)
o4 = | z3-3z2-z-1 |

```

2.4.4 イデアルの共通部分, イデアル商, saturation, radical メンバーシップ

イデアルの共通部分, イデアル商, saturation, $f \in \sqrt{I}$ の判定はは次の方法により行うことができる。 $R = K[x_1, \dots, x_n]$ とする。

- イデアルの共通部分

R のイデアル $I = \langle f_1, \dots, f_k \rangle, J = \langle g_1, \dots, g_l \rangle$ に対し, t を新しい変数とすれば

$$I \cap J = \langle tf_1, \dots, tf_k, (1-t)g_1, \dots, (1-t)g_l \rangle \cap R$$

ただし, 右辺のイデアルは $K[x_1, \dots, x_n, t]$ で考える。よって, 消去イデアル計算により共通部分が計算できる。

- イデアル商

R のイデアル I, J に対し $I : J = \{f \mid fJ \subset I\}$ である。 $J = \langle g_1, \dots, g_l \rangle$ なら $I : J = \bigcap_{i=1}^l I : \langle g_i \rangle$ である。 $I : \langle g \rangle$ を $I : g$ と書く。 $I : g = (I \cap \langle g \rangle) / g$ である。右辺は, $I \cap \langle g \rangle$ の生成系の各元を g で割ったもので生成されるイデアルである。よって, $I : J$ は共通部分計算により計算できる。

- saturation

R のイデアル I, J に対し $I : J^\infty = \bigcup_{m=1}^{\infty} (I : J^m)$ である。 $J = \langle g_1, \dots, g_l \rangle$ なら $I : J = \bigcap_{i=1}^l (I : \langle g_i \rangle^\infty)$ である。 $I : \langle g \rangle^\infty$ を $I : g^\infty$ と書く。 $I : g^\infty = (I + \langle tg - 1 \rangle) \cap R$ である。ただし右辺のイデアルは $K[x_1, \dots, x_n, t]$ で考える。よって saturation は共通部分計算により計算できる。

- $f \in \sqrt{I}$ の判定 (radical メンバーシップ判定)

R のイデアル $I, f \in R$ に対し, $f \in \sqrt{I} \Leftrightarrow I + \langle tf - 1 \rangle = K[x_1, \dots, x_n, t]$ である. $I + \langle tf - 1 \rangle = K[x_1, \dots, x_n, t]$ は $I + \langle tf - 1 \rangle$ の (任意項順序に関する) 簡約グレブナー基底が $\{1\}$ であることと同値だから, $f \in \sqrt{I}$ か否かはグレブナー基底を計算することで判定できる.

Macaulay 2 においては, イデアル I_1, I_2, \dots の共通部分は $\text{intersect}(I_1, I_2, \dots)$, イデアル I, J に対するイデアル商 $I : J$ は $\text{quotient}(I, J)$, saturation $I : J^\infty$ は $\text{saturate}(I, J)$ により計算できる.

イデアルの演算

```

i1 : R=QQ[x,y];
i2 : I=ideal(x^4-y^5,x^3-y^7);
o2 : Ideal of R
i3 : I1=quotient(I,x)
      5   4   3 2   2   7   2 3
o3 = ideal (y  - x , x y  - x , x  - x y )
o3 : Ideal of R
i4 : I2=quotient(I,x^2)
      2 2   5   4   6   3
o4 = ideal (x y  - x, y  - x , x  - x*y )
o4 : Ideal of R
i5 : I3=quotient(I,x^3)
      2   5   4   5   3
o5 = ideal (x*y  - 1, y  - x , x  - y )
o5 : Ideal of R
i6 : I4=quotient(I,x^4)
      2   5   4
o6 = ideal (x*y  - 1, y  - x )
o6 : Ideal of R
i7 : J=saturate(I,x)
      2   5   4   5   3
o7 = ideal (x*y  - 1, y  - x , x  - y )
o7 : Ideal of R
i8 : I3==I4
o8 = true
i9 : I2==I3
o9 = false

```

次の例は, 2.4.2 節の例における $f \in \sqrt{I}$ の判定を, ここで述べた方法で行ったものである. グレブナー基底が $\{1\}$ なので, $f \in \sqrt{I}$ と判定できる.


```

i1 : R=QQ[t,x,y,z];
i2 : I=ideal(x^4*y^2+z^2-4*x*y^3*z-2*y^5*z,x^2+2*x*y^2+y^4);
o2 : Ideal of R
i3 : f=y*z-x^3;
i4 : gens gb (I+ideal(t*f-1))
o4 = | 1 |

```

3 Asir によるグレブナー基底の計算

3.1 Asir について

3.1.1 起動方法

- \sqrt{x} メニュー, または KNOPPIX-Math-Start から起動する.
(openxm) の方を起動すれば, 種々のライブラリファイルを自動的に読みこんで起動する.
- 端末エミュレータから起動する.
Asir 単体ではコマンドライン編集機能を持たないので, openxm fep asir を実行する.

3.1.2 ヘルプ, マニュアル

ヘルプは help("function") で引ける. マニュアルはデスクトップの Math-Doc-Search で引くか, helph() コマンドでブラウザを立ち上げて HTML 形式のマニュアルを見るのが便利である.

3.1.3 ファイルのロード

ファイルは load により行う. 環境変数 ASIRLOADPATH で指定されたディレクトリを順に探す. この値は, シェルから openxm env を実行すると見ることができる.

3.2 基礎環および項順序

3.2.1 多項式の入力

Asir ではアルファベット小文字で始まり, アルファベット, 数字, _ (アンダースコア) からなる文字列が不定元である. 不定元を含む多項式が入力された場合, システムが内部的に保持する不定元の順序に従い内部形式に変換され保持される. 通常係数は自動的に有理数と判断される. すなわち, この場合には, 有理数係数の無限変数のユニバーサルな多項式環が設定されていると考えてよい. よって, 保持している多項式は, 内部の不定元順序が変更されない限り, 自由に加減乗算が可能である. しかし, 項順序に関する情報はなんら保持されていないため, グレブナー基底関連計算など, 項順序が必要が計算の都度, 項順序を指定する必要がある.

```
[1518] F=(x+y+z)^2;
x^2+(2*y+2*z)*x+y^2+2*z*y+z^2
[1519] G=F+u;
x^2+(2*y+2*z)*x+y^2+2*z*y+z^2+u
```

この表示から分かるように, Asir における多項式は再帰表現により保持されている. 再帰表現とは, 多項式を, 主変数に関する一変数多項式として表現するもので, 係数は, 主変数を含まない多項式である. これに対し, グレブナー基底に関連する計算では, 多項式を単項式の和として表現するのが便利である. これを分散表現と呼ぶ. Asir においては, グレブナー基底関連計算を行う場合, 暗黙あるいは明示的に分散表現への変換を行う. これについては後述する.

3.2.2 項順序

Asir においては, 項順序は変数順序と項順序型により指定される. 変数順序は不定元を並べたリストで表現する. この順序は単項式を指数ベクトルで表示する場合の各指数のインデックスを決める. 例えば, 変数順序が $[x, y, z, u, v, w]$ で与えられた場合, $x^a y^b z^c u^d v^e w^f$ は (a, b, c, d, e, f) で表示される. n 変数の変数リストが与えられているとき, 次のような項順序型が設定できる.

- 単純な項順序型

整数値で表される. 0 は全次数逆辞書式, 1 は全次数辞書式, 2 は辞書式順序を表す. これらは, 上の変数順序によりベクトルで表された単項式に対し適用される.

- ブロック項順序型

$[[O_1, n_1], [O_2, n_2], \dots, [O_l, n_l]]$ なるリストのリストで表される. これは, 変数リストを左から n_1, n_2, \dots, n_l ($n_1 + \dots + n_l = n$) ずつのブロックに分け, i 番目のブロックに単純項順序型 O_i を適用する項順序である. 項順序比較は, 1 番目のブロックから, 大小が決まるまで順に行う. よく使われるのは $[[0, n_1], [0, n_2]]$ なるもので, 先頭の n_1 変数を消去するための消去順序の一つである.

- 行列による項順序型

$m \times n$ 整数行列 M で表される. これは, 二つの単項式 $e = (e_1, \dots, e_n)$, $f = (f_1, \dots, f_n)$ に対し

$$e > f \Leftrightarrow M(e - f) \text{ の } 0 \text{ でない最も上の要素が正}$$

として定義される. M が項順序を表すためには,

- 整数ベクトル e に対し $Me = 0 \Leftrightarrow e = 0$
- 各列の 0 でない最も上の要素が正

という条件を満たす必要があるが, これを保証するのはユーザの責任である.

項順序型は, dp_ord により設定できる. あるいは関数の引数として与える場合もある.

分散表現への変換

```
[1532] F=x^2*y+y^3*z+x*z+x+1;
y*x^2+(z+1)*x+z*y^3+1
[1533] dp_ord(0)$
[1534] DF0=dp_ptod(F, [x,y,z]);
(1)*<<0,3,1>>+(1)*<<2,1,0>>+(1)*<<1,0,1>>+(1)*<<1,0,0>>+(1)*<<0,0,0>>
[1535] dp_ord(2)$
[1536] DF2=dp_ptod(F, [x,y,z]);
(1)*<<2,1,0>>+(1)*<<1,0,1>>+(1)*<<1,0,0>>+(1)*<<0,3,1>>+(1)*<<0,0,0>>
[1537] G=F+u;
y*x^2+(z+1)*x+z*y^3+u+1
[1538] DG=dp_ptod(G, [u,x,y,z]);
(1)*<<1,0,0,0>>+(1)*<<0,2,1,0>>+(1)*<<0,1,0,1>>+(1)*<<0,1,0,0>>
+(1)*<<0,0,3,1>>+(1)*<<0,0,0,0>>
[1539] dp_ht(DG);
(1)*<<1,0,0,0>>
```

この例では, 多項式 F, G を明示的に分散表現に変換している. $\text{dp_ord}(0)$ により $DF0$ は全次数逆辞書式で整列され, $\text{dp_ord}(2)$ により $DF2$ は辞書式に整列される. また, DG は, u が最大の辞書式となるため, $\langle\langle 1, 0, 0, 0 \rangle\rangle$ が先頭となっている. dp_ht は先頭項 (係数 1), dp_hc は先頭係数, dp_hm は係数付きの先頭項を返す. (用語は初期に使われていたものを採用しており, 最近の用法と異なることに注意されたい.) 分散表現多項式間の演算は, 同じ項順序で変換されたものに限られる. Macaulay2 と異なり, これを守るのはユーザの責任である.

3.3 グレブナー基底の計算

本節では, イデアルの入力およびグレブナー基底計算について述べる. 以下で述べる関数はライブラリ `gr` に定義されているものも `gr` をロードしておく必要がある. ただし, KNOPPIX/Math にインストールされている環境では, デフォルトでロードされているので操作は不要である.

Asir では, イデアルは多項式のリストで表現される. この段階では基礎環にあたるものはまだ未定である. グレブナー基底計算の際に指定される変数リストと項順序型, および係数体を指定する引数により基礎環が係数体を含めて決定される. 主なグレブナー基底計算関数を挙げる.

- $\text{nd_gr}(Plist, Vlist, Char, Ord)$

$Plist$ はイデアルを表す多項式リストである. 変数リスト $Vlist$, 項順序型 Ord で指定される項順序をもつ多項式環のイデアル $\langle Plist \rangle$ の簡約グレブナー基底を計算する. $Char = 0$ のとき有理数体係数, $Char$ が素数のとき有限体 \mathbb{F}_{Char} 上で計算する. 結果は多項式のリストである.

- $\text{nd_gr_trace}(Plist, Vlist, Homo, Prime, Ord)$

この関数は有理数体上のグレブナー基底を、有限体上でのショートカット計算を用いて効率よく計算するための関数である。

Plist はイデアルを表す多項式リストである。変数リスト *Vlist*, 項順序型 *Ord* で指定される項順序をもつ多項式環のイデアル $\langle Plist \rangle$ の簡約グレブナー基底を計算する。 *Prime* は 1 を指定しておく。(他の値の場合はマニュアルを参照。) *Homo* が 1 のとき、斉次化を経由して計算する。 *Homo* が 0 のとき斉次化を経由しないで計算する。どちらも結果は同一であるが、中間係数が膨張する可能性がある場合、 *Homo* = 1 で計算するのが安全である。(Macaulay2 のソースコードで確かめたわけではないが、中間基底の生成の様子を見ると、Macaulay2 はデフォルトで斉次化を行っているように見える。)

Asir によるグレブナー基底計算

```
[1517] load("cyclic")$
[1527] C=cyclic(7);
[c6*c5*c4*c3*c2*c1*c0-1,...]
[1528] V=vars(C);
[c0,c1,c2,c3,c4,c5,c6]
[1529] nd_gr(C,V,31991,0)$
...
2.016sec + gc : 0.072sec(2.089sec)
[1530] nd_gr(C,V,0,0)$
(5分待つて中断)
[1530] G=nd_gr_trace(C,V,1,1,0)$
...
19.54sec + gc : 5.428sec(25.02sec)
[1531] G[0];
(((238539226659020007130662*c6*c4-282765997082979724500242*c5^2-...
[1532] length(G);
209
```

この例では、有名なベンチマーク問題 *cyclic-7* の全次数逆辞書式順序グレブナー基底計算を有限体 \mathbb{F}_{31991} および有理数体上で行っている。計算時間は、Intel Xeon X5470 (3.33GHz) のものである(以下同様)。最初の *nd_gr* は有限体上の計算で、2秒程で終わっているが、これを有理数体上で行う(2番目の *nd_gr*)と、計算途中で係数膨張のため計算が進まなくなる。一方で、*nd_gr_trace* を *Homo* = 1 で実行すると、25秒で計算が終了する。この例に限らず、有理数体上の計算の場合、常に係数膨張の危険があるため、*Homo* = 1 で計算することを推奨する。

計算結果のリストは、入力と同一のイデアルを生成するグレブナー基底である。リスト *G* の *i* 番目の要素は *G*[*i*] (*i* は 0 から始まる) で取り出せる。

3.4 基本的な応用

3.4.1 イニシャルイデアルの計算

Asir でイニシャルイデアルを計算するには、まずグレブナー基底を計算し、その各生成元先頭項を取り出す。このためには、再帰表現された多項式を *dp_ptod* で分散表現に変換し、

dp_ht で先頭項を取り出すという操作が必要となる. さらに, 例では dp_dtop により各先頭単項式を再帰表現に逆変換している. また, 0次元の場合のみ, dp_mbase により, 標準単項式全体を計算できる.

Asir によるイニシャルイデアルの計算

```
[1517] B=[x^2*y^2-z^2,x^3-y*z^2,x^2*z^4-y^2];
[1518] V=[x,y,z]$
[1519] G=nd_gr(B,V,0,0);
[1520] D=map(dp_ptod,G,V)$ H=map(dp_ht,D)$
[1521] [1522] map(dp_dtop,H,V);
[1523] map(dp_dtop,dp_mbase(H),V);
[1524] length(@@);
52
```

3.4.2 剰余計算

多項式を多項式リストに現れる多項式で割った剰余を計算する関数が p_nf および p_true_nf である. 前者は, 剰余の分母を払って, 整数係数で返すもので, 剰余が 0 か否かの判定に用いる. 後者は, [num, den] なるリストを返す. num は p_nf が返すもので, num/den が真の剰余となる.

剰余計算

```
[1517] B=[u2*u0-2*u2+3,(2*u1-1)*u0^2-u0-2*u2,2*u1^3+u2+4]$
[1518] V=[u0,u1,u2]$
[1519] G=nd_gr(B,V,0,0);
[1520] Q=p_nf(u0^5+u1^5+u2^5,G,V,0);
2851262910*u2^3+30078832770*u2^2+(22194374760*u1-21995962245)*u2-...
[1521] QR=p_true_nf(u0^5+u1^5+u2^5,G,V,0);
[2851262910*u2^3+30078832770*u2^2+(22194374760*u1-...,35373600]
```

3.4.3 消去法

2.4.3 節で述べたように, $K[Z]$ ($Z = X \cup Y, X \cap Y = \emptyset$) のイデアル I に対する消去イデアル $I_Y = I \cap K[Y]$ の生成系の計算には消去順序によるグレブナー基底を使う. 有理数体上で計算する場合には nd_gr_trace を斉次化ありで使うことをお勧めする. I の消去順序によるグレブナー基底 G から I_Y のグレブナー基底 G_Y を取り出すには, elimination (ライブラリ primdec_mod に定義されているがマニュアルにはない) を使う.

次の例では, B が生成するイデアルの $\{u_0, u_1\} \gg \{u_2\}$ なる消去順序によるグレブナー基底を計算し, elimination により u_2 のみを含む多項式を取り出している.

消去イデアルの計算

```
[1518] load("primdec_mod")$
[1664] B=[u2*u0-2*u2+3,(2*u1-1)*u0^2-u0-2*u2,2*u1^3+u2+4]$
[1665] V=[u0,u1,u2]$
[1666] G1=nd_gr_trace(B,V,1,1,[[0,2],[0,1]])$
[1667] elimination(G1,[u2]);
[8*u2^9+72*u2^8+292*u2^7-2036*u2^6-198*u2^5+20682*u2^4-57429*u2^3+...]
```

3.4.4 最小多項式の計算

消去法の特別な例として、イデアルと一変数多項式環の共通部分 $I \cap K[z]$ の計算がある。これも前節で述べた一般的な方法で計算できるが、 I が 0 次元イデアルの場合、より直接的な方法により $I \cap K[z]$ の単項生成元のみを計算することができる。gr で定義されている `minipoly` は、有理数体係数多項式環の 0 次元イデアル I および多項式 f に対し、 $m(f) \in I$ を満たすような 0 でない最小次数の多項式 m を計算する。

次の例では、これも有名なベンチマーク問題 *katsura - 7* において、 u_7 の最小多項式を `minipoly` により求めている。`minipoly` の引数は、最初の $(G, V, 0)$ で、グレブナー基底と項順序、 u_7 が、最小多項式を計算したい多項式、 t が、最小多項式の変数の指定である。最後の変数は、 V に現れないものを指定する必要がある。この例を、消去順序グレブナー基底計算で行ってみれば、その大変さがよくわかる。

最小多項式の計算

```
[1518] load("katsura")$
[1522] B=katsura(7)$
[1523] V=[u0,u1,u2,u3,u4,u5,u6,u7]$
[1524] G=nd_gr_trace(B,V,1,1,0)$
[1525] minipoly(G,V,0,u7,t)$
[1526] deg(@@,t);
128
```

3.4.5 0次元イデアルの項順序変換

代数方程式の求解を行う場合、辞書式、あるいはそれに近い消去順序でグレブナー基底を計算する必要が出て来るが、これらを Buchberger アルゴリズムで直接計算するのは一般に効率が大変悪い。このような場合、例えば全次数辞書式順序でグレブナー基底を計算し、それから線形代数的手法により別の項順序に関するグレブナー基底を計算する方法がいくつかある。gr で定義されている `tolex` は、0 次元イデアルの任意項順序のグレブナー基底を入力として、辞書式順序のグレブナー基底を計算する関数である。この場合、出力の変数順序のみ変更可能である。次の例は、*katsura - 7* の辞書式順序グレブナー基底を項順序変換で計算したものである。これを `nd_gr_trace` で直接辞書式順序で計算することは、途中の係数膨張の様子からみてほぼ不可能であろう。

```
[1524] G=nd_gr_trace(katsura(7),V=[u0,u1,u2,u3,u4,u5,u6,u7],1,1,0)$
2.676sec + gc : 1.356sec(4.032sec)
[1525] G2=tolex(G,V,0,V)$
279.5sec + gc : 57.68sec(337.5sec)
```

3.4.6 イdeal演算

イdealの共通部分, イdeal商, saturation を計算する関数は, ライブラリのあちこちで定義され使われているが, マニュアルに書かれていないので, 消去イdeal計算を用いてこれらを実装してみると, よい練習になるであろう. 計算方法は 2.4.4 節にある.

4 練習問題

参考文献にあげた図書には, 種々のソフトウェアの使用法の解説とともに, 多数の練習問題が収録されている. 以下の問題のいくつかはこれらから引用した.

- 以下の各項を行う方法を, Macaulay2, Asir それぞれについて調べよ.
 - ファイルに結果を書き出す.
 - 繰り返しを行う.
 - 多項式の因数分解を行う.
- $\mathbf{Q}[x, y, z]$ のイdeal $I = \langle x^2 + z, xy + y^2 + z, xz - y^3 - 2yz, y^4 + 3y^2z + z^2 \rangle$, $J = \langle x^2 + z, xy + y^2 + z, x^3 - yz \rangle$ の包含関係を調べよ.
- $f_1 = 3x^2yz^2 + 3z + (-2x + 2)y + 2x$, $f_2 = 3yz^5 + (-xy^2 + 2)z - 2y^4 + 2y$, $f_3 = xy^3z^3 - 2yz^2 - z - 2y + x^2$ に対し, $I = \langle f_1, f_2, f_3 \rangle \subset \mathbf{Q}[x, y, z]$ とおく.
 - $\dim I = 0$ を確かめよ.
 - $\dim_{\mathbf{Q}} \mathbf{Q}[x, y, z]/I$ を求めよ.
 - I の $x > y > z$ なる辞書式順序での簡約グレブナー基底が $\{g_0(z), x - g_1(z), y - g_2(z)\}$ という形であることを確かめよ.
- $f_1 = x^2 + y^2 + z^2 - 9$, $f_2 = 3x^2 - y^2z$, $f_3 = x^2z - 2y^2 + 2$ に対し, $f_1 = f_2 = f_3 = 0$ を満たす $(x, y, z) \in \mathbf{C}^3$ を全て求めよ.
- $\alpha = 3^{\frac{1}{5}}, \beta = 5^{\frac{1}{3}}$ とする.
 - $\alpha + \beta$ の \mathbf{Q} 上の最小多項式を求めよ.
 - $\frac{1}{\alpha + \beta}$ を α, β の有理数係数多項式で表せ.
- (Asir でのプログラミング経験がある人向け) イdealの共通部分, イdeal商, saturation の計算および radical メンバーシップを判定する関数を記述せよ.

参考文献

- [1] W. Adams, P. Lounstau, An Introduction to Gröbner basis Bases. Graduate Studies in Mathematics, Vol. 3, AMS (1994).
- [2] D. Cox, J. Little, D. O'Shea, Using Algebraic Geometry. GTM Vol. 185, Springer (2005).
- [3] D. Eisenbud, D. Grayson, M. Stillman, B. Sturmfels (Eds.), Computations in Algebraic Geometry with Macaulay 2. Algorithms and Computation in Mathematics **8**, Springer-Verlag (2000).
- [4] G.-M. Greuel, G. Pfister, A Singular Introduction to Commutative Algebra. Springer (2007).
- [5] M. Kreuzer, L. Robbiano, Computational Commutative Algebra 1. Springer (2008).

9月15日(火曜日)

(講義1) 10:00~12:00

「マルコフ基底と実験計画」

竹村 彰通(東京大学大学院情報理工学系研究科・JST CREST)

青木 敏(鹿児島大学大学院理工学研究科・JST CREST)

[概要: 実験計画法は、グレブナー基底理論の統計学への応用として、早くから注目されてきた分野の一つである。本講義では、まず、一部実施計画の別名関係とイデアル所属問題の対応を解説する。次に、実験計画データが計数値の場合の仮説検定を、グレブナー基底(マルコフ基底)を利用したモンテカルロ法で遂行する方法について解説する。]

(演習1) 14:00~

「マルコフ基底と実験計画」

中山 洋将(神戸大学大学院理学研究科・JST CREST)

西山 絢太(神戸大学大学院理学研究科・JST CREST)

[概要: トーリックイデアルを使った、整数計画の解法、実行可能集合の数え上げの方法を実習する。システムとしては CoCoA、4ti2、Risa/Asir 等を利用する。数え上げの方法を基礎として、マルコフ基底を利用したモンテカルロ法を実習する。]

マルコフ基底と実験計画

東京大 情報理工・JST CREST 竹村彰通
鹿児島大 理工・JST CREST 青木敏

2009年9月15日
JST CREST Gröbner School

1

目次

1. 分割表の条件付検定
 - 十分統計量
 - 2×2 分割表の独立モデル
 - 相似検定
 - $I \times J$ 分割表の独立性の検定
2. マルコフ基底
 - マルコフ基底
 - 2元分割表の独立モデルのマルコフ基底
 - 3元分割表の無3因子交互作用モデルのマルコフ基底
 - マルコフ基底とイデアール

2

3. 実験計画とマルコフ基底

- 2水準実験・組合せ配置
- 2水準実験・一部実施計画

3

1. 分割表の条件付検定

4

十分統計量

- X : 離散確率変数
- x : X の実現値
- X, x は、非負整数 $\{0, 1, 2, \dots\}$ の値をとる。
多次元確率変数は太字で、個々の変数は添字で表現する。
例えば、

$$\mathbf{X} = (X_i) = (X_1, X_2, \dots, X_n)$$

$$\mathbf{X} = (X_{ij}) = (X_{11}, \dots, X_{1J}, \dots, X_{I1}, \dots, X_{IJ})$$

$$\Pr(\mathbf{X} = \mathbf{x}) = \Pr(X_{ij} = x_{ij} \text{ for all } i, j)$$

という具合。

5

- $p(\mathbf{x}) = \Pr(\mathbf{X} = \mathbf{x})$: X の確率分布 (確率関数)
 $p(\mathbf{x})$ は、パラメータ (母数) で特徴付けられる。
- [定義] 一般に、多次元確率変数 X の関数 (統計量) を
 $T(\mathbf{X}) = (T_1(\mathbf{X}), \dots, T_k(\mathbf{X}))$, パラメータを $\theta = (\theta_1, \dots, \theta_\nu)$ とする。
 T を与えたときの X の条件付分布

$$p(\mathbf{x} \mid \mathbf{t}) = \Pr(\mathbf{X} = \mathbf{x} \mid T(\mathbf{X}) = \mathbf{t})$$

が θ によらないとき、 T を θ の十分統計量とよぶ。

- つまり、 T の値を知っていれば、 X はそれ以上 θ に関する情報をもたない (T を知れば十分)、という意味。

6

- [定理](分解定理) T が θ の十分統計量であるための必要十分条件は、 X の確率分布が

$$p(x) = h(x)g(T(x), \theta)$$

の形で分解できることである。

- [定義] $p(x)$ が,

$$p(x) = h(x) \exp \left(\sum_{j=1}^k T_j(x) \psi_j(\theta) - c(\theta) \right)$$

と書けるとき、(k 母数) 指数型分布族とよぶ。

- 分解定理より、 (T_1, \dots, T_k) は k 次元十分統計量である。

7

- 本講義で扱う分布は、すべて指数型分布族に含まれ、多次元の母数をもつ。

- 本講義では、母数 θ を、興味のある母数 λ と興味のない母数 (局外母数) ψ に分解する変数変換

$$\theta \leftrightarrow (\lambda, \psi)$$

を考え、

$$H_0 : \lambda = (0, \dots, 0)$$

$$H_1 : \lambda \neq (0, \dots, 0)$$

の形の検定問題を考える。

- まず、 2×2 分割表の例で説明する。

8

2 × 2 分割表の独立モデル

- 観測値が 2×2 分割表として得られる 3 つの例を紹介する。
- いずれの場合も、興味があるのは「ふたつの要因に関連があるか」、つまり「行と列の独立性」である。
- 「行と列が独立である」というモデル (帰無仮説) の下では、局外母数に関する十分統計量が行和と列和になることを確認する。

9

例 1: 観測値が独立な 2 項分布にしたがう場合

胃ガンに対する喫煙のリスクを調べるため、胃ガン患者 20 人と健常者 100 人に対し、過去の喫煙習慣の有無を調査したところ、以下の結果を得た。喫煙習慣は胃ガンに関係あるといえるか?

	喫煙歴あり	喫煙歴なし	合計
胃ガン患者	14	6	20
健常者	56	44	100

(数値は架空)

10

- X_1 : 胃ガン患者の喫煙者数
 X_2 : 健常者の喫煙者数

$$X_1 \sim \text{Bin}(n_1, p_1), X_2 \sim \text{Bin}(n_2, p_2), X_1 \perp X_2$$

p_1 : 胃ガン患者の喫煙確率

p_2 : 健常者の喫煙確率

観測値	喫煙	非喫煙	合計	確率	喫煙	非喫煙	合計
胃ガン	x_1	$n_1 - x_1$	n_1	胃ガン	p_1	$1 - p_1$	1
健常者	x_2	$n_2 - x_2$	n_2	健常者	p_2	$1 - p_2$	1

- $X = (X_1, X_2)$ の確率分布

$$p(x) = \binom{n_1}{x_1} \binom{n_2}{x_2} p_1^{x_1} (1 - p_1)^{n_1 - x_1} p_2^{x_2} (1 - p_2)^{n_2 - x_2},$$

$$x_1 = 0, 1, \dots, n_1, x_2 = 0, 1, \dots, n_2$$

11

- 母数は (p_1, p_2) であり、2 次元。

- 母数変換

$$\psi = \log \frac{p_2}{1 - p_2}, \quad \lambda = \log \frac{p_1(1 - p_2)}{p_2(1 - p_1)}$$

を考える。逆変換は

$$p_1 = \frac{e^{\psi + \lambda}}{1 + e^{\psi + \lambda}}, \quad p_2 = \frac{e^{\psi}}{1 + e^{\psi}}$$

- 「行 (胃ガンか否か) と列 (喫煙歴の有無) が独立」という仮説 (独立モデル) は、 $\lambda = 0$ と表される。

λ は、対数オッズ比とよばれる量。医学統計でよく使われる。

12

- 新しい母数に関する確率分布 (指数型分布族) :

$$p(\mathbf{x}) = \binom{n_1}{x_1} \binom{n_2}{x_2} \exp((x_1 + x_2)\psi + x_1\lambda - n_1 \log(1 + e^{\psi+\lambda}) - n_2 \log(1 + e^{\psi}))$$

- 従って, $H_0: \lambda = 0$ の下では, $X_1 + X_2$ が母数 ψ の十分統計量である.
- n_1, n_2 が固定なので, 十分統計量を固定することは, 「 2×2 分割表の行和と列和」をすべて固定することと同値.

13

例 2: 観測値が多項分布にしたがう場合

若者の政治への関心を調べるため, 街頭で 100 人に,

- 与野党のいずれを支持するか
- 今回の選挙で投票に行くか

を聞いた. 結果は以下のようになった.

支持政党と投票率には関連があるといえるか?

	与党支持	野党支持	合計
投票する	22	43	65
棄権する	14	21	35
合計	36	64	100

(数値は架空)

14

- $\mathbf{X} = (X_{11}, X_{12}, X_{21}, X_{22})$ は, 多項分布 (4 項分布) $M(n, (p_{11}, p_{12}, p_{21}, p_{22}))$ にしたがう, と仮定するのが自然.
- x_{ij}, p_{ij} は以下 ($i, j = 1, 2$)

観測値	与党	野党	合計	確率	与党	野党	合計
投票	x_{11}	x_{12}		投票	p_{11}	p_{12}	
棄権	x_{21}	x_{22}		棄権	p_{21}	p_{22}	
合計			n	合計			1

- $\mathbf{X} = (X_{11}, X_{12}, X_{21}, X_{22})$ の確率分布

$$p(\mathbf{x}) = \frac{n!}{x_{11}!x_{12}!x_{21}!x_{22}!} p_{11}^{x_{11}} p_{12}^{x_{12}} p_{21}^{x_{21}} p_{22}^{x_{22}},$$

$$p_{11} + p_{12} + p_{21} + p_{22} = 1,$$

$$x_{11} + x_{12} + x_{21} + x_{22} = n$$

15

- 母数 $p = (p_{11}, p_{12}, p_{21}, p_{22})$ は 3 次元.
- 母数変換

$$\psi_1 = \log \frac{p_{12}}{p_{22}}, \quad \psi_2 = \log \frac{p_{21}}{p_{22}}, \quad \lambda = \log \frac{p_{11}p_{22}}{p_{12}p_{21}}$$

を考える. 逆変換は

$$p_{11} = \frac{e^{\psi_1 + \psi_2 + \lambda}}{1 + e^{\psi_1} + e^{\psi_2} + e^{\psi_1 + \psi_2 + \lambda}},$$

$$p_{12} = \frac{e^{\psi_1}}{1 + e^{\psi_1} + e^{\psi_2} + e^{\psi_1 + \psi_2 + \lambda}},$$

$$p_{21} = \frac{e^{\psi_2}}{1 + e^{\psi_1} + e^{\psi_2} + e^{\psi_1 + \psi_2 + \lambda}},$$

$$p_{22} = \frac{1}{1 + e^{\psi_1} + e^{\psi_2} + e^{\psi_1 + \psi_2 + \lambda}}$$

16

- 「行 (投票に行くか) と列 (支持政党) が独立」という仮説 (独立モデル) は, $\lambda = 0$ と表される.

- 新しい母数に関する確率分布 (指数型分布族) :

$$p(\mathbf{x}) = \frac{n!}{x_{11}!x_{12}!x_{21}!x_{22}!} \exp((x_{11} + x_{12})\psi_1 + (x_{21} + x_{22})\psi_2 + x_{11}\lambda - n \log(1 + e^{\psi_1} + e^{\psi_2} + e^{\psi_1 + \psi_2 + \lambda}))$$

- 従って, $H_0: \lambda = 0$ の下では, $(X_{11} + X_{12}, X_{21} + X_{22})$ が母数 (ψ_1, ψ_2) の十分統計量である.
- n が固定なので, 十分統計量を固定することは, 「 2×2 分割表の行和と列和」をすべて固定することと同値.

17

例 3: 観測値がポアソン分布にしたがう場合

ある工場では, ある鑄造工程で作られる製品の不良品の個数が少なくなる条件を調べるために, 2 通りの熱処理時間 (長, 短) と 2 通りの触媒 (A, B) の組合せについて, 不良品の個数を計測した. 結果は以下のようになった. 熱処理時間の長短と触媒の種類を, どのように設定するのが適切といえるか?

	触媒 A	触媒 B
長時間	5	12
短時間	7	6

(数値は架空)

18

- 一見、もっとも不良品数が少ない、(長時間, 触媒 A) の組合せが最適と思える。
- しかし、「熱処理時間と触媒の種類の影響は独立」と考えれば、「長時間」における不良品数 (5 + 12 = 17) よりも「短時間」における不良品数 (7 + 6 = 13) の方が少ないので、(短時間, 触媒 A) の組合せの方が適切かもしれない。
- 結局、前の 2 例と同様、「独立モデルの当てはまり具合」の評価が問題となる。

19

- X_{ij} , $i, j = 1, 2$: 各組合せにおける不良品の個数

$$X_{ij} \sim \text{Po}(\mu_{ij}), \quad i, j = 1, 2, \quad X_{ij} \text{ はすべて独立}$$

$$\mu_{ij}: \text{各組合せにおける不良品の期待値} \quad (E(X_{ij}) = \mu_{ij})$$

観測値	触媒 A	触媒 B	期待値	触媒 A	触媒 B
長時間	x_{11}	x_{12}	長時間	μ_{11}	μ_{12}
短時間	x_{21}	x_{22}	短時間	μ_{21}	μ_{22}

- $X = (X_{11}, X_{12}, X_{21}, X_{22})$ の確率分布

$$p(\mathbf{x}) = \prod_{i=1}^2 \prod_{j=1}^2 \frac{\mu_{ij}^{x_{ij}} e^{-\mu_{ij}}}{x_{ij}!},$$

$$x_{ij} = 0, 1, 2, \dots$$

20

- 母数 $(\mu_{11}, \mu_{12}, \mu_{21}, \mu_{22})$ は 4 次元。

- 母数変換

$$\psi_0 = \log \mu_{22}, \quad \psi_1 = \log \frac{\mu_{12}}{\mu_{22}}, \quad \psi_2 = \log \frac{\mu_{21}}{\mu_{22}}, \quad \lambda = \log \frac{\mu_{11}\mu_{22}}{\mu_{12}\mu_{21}}$$

を考える。逆変換は

$$\mu_{11} = e^{\psi_0 + \psi_1 + \psi_2 + \lambda}, \quad \mu_{12} = e^{\psi_0 + \psi_1}, \quad \mu_{21} = e^{\psi_0 + \psi_2}, \quad \mu_{22} = e^{\psi_0}$$

- 「行 (熱処理時間) と列 (触媒の種類) が独立」という仮説 (独立モデル) は、 $\lambda = 0$ と表される。

21

- 新しい母数に関する確率分布 (指数型分布族):

$$p(\mathbf{x}) = \frac{1}{x_{11}! x_{12}! x_{21}! x_{22}!} \exp((x_{11} + x_{12} + x_{21} + x_{22})\psi_0 + (x_{11} + x_{12})\psi_1 + (x_{11} + x_{21})\psi_2 + x_{11}\lambda - (e^{\psi_0 + \psi_1 + \psi_2 + \lambda} + e^{\psi_0 + \psi_1} + e^{\psi_0 + \psi_2} + e^{\psi_0}))$$

- 従って、 $H_0: \lambda = 0$ の下では、

$$(X_{11} + X_{12} + X_{21} + X_{22}, X_{11} + X_{12}, X_{21} + X_{22})$$

が母数 (ψ_0, ψ_1, ψ_2) の十分統計量である。

- この場合も、十分統計量を固定することは、「 2×2 分割表の行和と列和」をすべて固定することと同値。

22

まとめ

- 3 つの例は、データの取り方 (サンプルサイズを固定するか否か、など) が異なるため、仮定する確率分布 (2 項分布, 多項分布, ポアソン分布) も、母数の次元も異なる。
- しかし、興味の対象はすべて、「行と列との独立性」と表現できる。
- 適切な変数変換により、「行と列の独立モデル」を、1 次元母数 λ に関する帰無仮説

$$H_0: \lambda = 0$$

として表すことができる。

- このとき、残りの「興味のない母数」(これは**局外母数**とよばれる) に関する十分統計量は、 2×2 分割表の行和と列和となる。

23

- 「何も構造を仮定しないモデル」では、母数の次元はデータの次元に等しい。例えば、

- 例 1 (独立な 2 項分布) では、母数 (p_1, p_2) は 2 次元。

- 例 2 (4 項分布) では、母数 $(p_{11}, p_{12}, p_{21}, p_{22})$ は 3 次元 ($\sum p_{ij} = 1$ の制約があるので)。

- 例 3 (独立なポアソン分布) では、母数 $(\mu_{11}, \mu_{12}, \mu_{21}, \mu_{22})$ は 4 次元。

このようなモデルを、**飽和モデル**という。

- 本講義で扱う問題は、

H_0 : 「母数の次元が少ない、飽和モデルのサブモデル」 vs

H_1 : 「飽和モデル」

の検定問題。

24

Remark

- 統計学では、伝統的に、**確率の対数をとって考える**ことが多い。
これまで見てきた変数変換は、**対数線形モデル**

$$\log p_{ij} = \psi_0 + \psi_{1i} + \psi_{2j} + \lambda_{ij}$$

において、母数の次元が合うように適当な制約を入れて書き直したものに他ならない。

- 最近では、母数の対数をとらずに、単項式のままで考えることも多い (**toric model**)。例えば、2元分割表の独立モデルであれば、

$$p_{ij} = \alpha_i \beta_j$$

と書けばよい。

この場合も、母数の次元を合わせるために適当な制約を入れなければならない点は同じなので、結局、同じような変換を考えることになる。しかし、例えば多項分布の場合、確率関数は

$$p(\mathbf{x}) = \frac{n!}{\prod_{i,j} x_{ij}!} \prod_{i,j} p_{ij}^{x_{ij}} = \frac{n!}{\prod_{i,j} x_{ij}!} \left(\prod_i \alpha_i^{x_{i+}} \right) \left(\prod_j \beta_j^{x_{+j}} \right)$$

となり、行和と列和が十分統計量になることが分かる。

- 対数で (対数線形モデル) を考えることのメリット:

母数の解釈がしやすい。

本講義では、主に、統計学の伝統的な記法である、対数線形モデルで説明する。

相似検定

- 本講義では、帰無仮説

$$H_0 : \lambda = (0, \dots, 0)$$

の検定問題を、**局外母数に関する十分統計量を固定した条件付分布にもとづく条件付検定**として解く手法を紹介する。

- このような検定は、**相似検定**とよばれる。
- 2 × 2 分割表の3つの例では、観測値とその行和、列和を統一的に

x_{11}	x_{12}	x_{1+}
x_{21}	x_{22}	x_{2+}
x_{+1}	x_{+2}	x_{++}

と書けば、条件付分布はすべて (例えば x_{11} の) 1 変数関数となり、

$$p(x_{11} | x_{++}, x_{1+}, x_{+1}) = \frac{x_{1+}! x_{2+}! x_{+1}! x_{+2}!}{x_{++}!} \prod_{i=1}^2 \prod_{j=1}^2 \frac{1}{x_{ij}!}$$

$$= \frac{\binom{x_{1+}}{x_{11}} \binom{x_{++} - x_{1+}}{x_{+1} - x_{11}}}{\binom{x_{++}}{x_{+1}}}$$

と書ける。これは、超幾何分布とよばれる。

- 超幾何分布: 「壺の中に、1 と書いた球を x_{1+} 個、2 と書いた球を x_{2+} 個入れ、そこから**目隠しして** x_{+1} 個を取り出す。このとき、取り出した球のうち 1 と書かれた球の数 X_{11} の分布」

- このように、条件付分布を考えることにより、局外母数によらず、また、(データの取り方に依存して決まる) 最初に仮定した確率分布にもよらない議論をする。

- 2 × 2 分割表の例であれば、例えば片側検定

$$H_0 : \lambda = 0$$

$$H_1 : \lambda > 0$$

であれば、

$$X_{11} > c \implies H_0 \text{ を棄却}$$

という検定方式が自然。

- この検定の**有意確率 (p 値)**は、

$$p \text{ 値} = \Pr(X_{11} \geq x_{11} | H_0) = \sum_{x \geq x_{11}} p(x | x_{++}, x_{1+}, x_{+1})$$

と書ける。この値を計算し、有意水準 α (例えば 0.05) と比較して、

$$p \text{ 値} \leq 0.05 \implies H_0 \text{ を棄却}$$

とすればよい。

- p 値: H_0 が真と仮定したときに、実際に得られたデータが (H_1 の方向に) 「どれだけ珍しいか」を評価したもの (確率)。この値が小さいときには、「今回の測定で、たまたまこんな珍しい出来事が起こったと考えるのは不自然だ。つまり、 H_0 は誤りである」と判断するのが、統計的仮説検定の考え方。

- 胃ガン患者と喫煙歴の例:

	喫煙歴あり	喫煙歴なし	合計
胃ガン患者	14	6	20
健常者	56	44	100
合計	70	50	120

$$p(x_{11} | 120, 20, 70) = \frac{\binom{20}{x_{11}} \binom{100}{70 - x_{11}}}{\binom{120}{70}}$$

$$p \text{ 値} = \sum_{x=14}^{20} p(x | 120, 20, 70) = 0.1818$$

H_0 は棄却できない。「胃ガン患者の方が喫煙率が高い」(あるいは、「喫煙者の方が胃ガンになりやすい」とはいえない。

31

- R での計算

```
> data <- matrix(c(14,6,56,44),2,2,byrow=T)
> data
      [,1] [,2]
[1,]  14   6
[2,]  56  44
> fisher.test(data,alternative="greater")

Fisher's Exact Test for Count Data

data: data
p-value = 0.1818
alternative hypothesis: true odds ratio is greater than 1
95 percent confidence interval:
 0.697064      Inf
sample estimates:
odds ratio
 1.824451
>
```

32

- もう少し小さい例で考えてみる (演習問題 5 の参考).

	喫煙 なし	合計
胃ガン	2	3
健常者	2	5
合計	4	8

$$p(x_{11} | 8, 3, 4) = \frac{\binom{3}{x_{11}} \binom{5}{4 - x_{11}}}{\binom{8}{4}}$$

超幾何分布 (独立性の帰無仮説) の下では、**十分統計量が等しい** 以下の 4 つの結果

3	0	2	1	1	2	0	3
1	4	2	3	3	2	4	1

が得られる確率は、順に $\left(\frac{1}{14}, \frac{6}{14}, \frac{6}{14}, \frac{1}{14}\right)$.

$$p \text{ 値} = \Pr(\text{実際の観測値 } x_{11} = 2 \text{ と同程度以上に珍しい結果が得られる}) \\ = p(2 | 8, 3, 4) + p(3 | 8, 3, 4) = 0.5$$

33

$I \times J$ 分割表の独立性の検定

- 次に、一般の $I \times J$ 分割表の独立モデルの検定について述べる。この場合、 λ は多次元であり、 H_0 は複合仮説となる。
- 考え方は 2×2 分割表の場合と同じ。つまり、**適当な母数変換により、母数を (ψ, λ) と変換し、局外母数 ψ に関する十分統計量を固定した条件付検定 (相似検定) を行う。**
- ただし、条件付分布は多次元分布となり、**適当な検定統計量を考える必要がある。**

34

2元分割表の独立性の検定の例 (演習問題 6)

例：幾何数理工学と推測数理工学の試験成績

幾何 \ 推測	5	4	3	2	1-	計
5	2	1	1	0	0	4
4	8	3	3	0	0	14
3	0	2	1	1	1	5
2	0	0	0	1	1	2
1-	0	0	0	0	1	1
計	10	6	5	2	3	26

- これは、5 行 5 列の 2 元分割表
- H_0 : 「幾何数理工学と推測数理工学の成績は独立」

35

- $X = (X_{11}, \dots, X_{15}, \dots, X_{51}, \dots, X_{55})$
- この場合の自然なモデルは、**総度数 n を固定した多項分布モデル**

$$p(x) = \frac{n!}{\prod_{i,j} x_{ij}!} \prod_{i,j} p_{ij}^{x_{ij}}, \quad \sum_{i,j} p_{ij} = 1, \\ \sum_{i,j} x_{ij} = n$$

- p_{ij} は、 (i, j) セルの生起確率を表す母数。
- 飽和モデルの母数の次元は、 $I, J - 1 = 24$ 。

36

- 母数変換: 対数線形モデル

$$\log p_{ij} = \psi_0 + \psi_{1i} + \psi_{2j} + \lambda_{ij}$$

- 母数には,

$$\sum_{i=1}^I \psi_{1i} = \sum_{j=1}^J \psi_{2j} = \sum_{i=1}^I \lambda_{ij} = \sum_{j=1}^J \lambda_{ij} = 0$$

の制約を入れる。飽和モデルの次元は確かに,

$$1 + (I - 1) + (J - 1) + (I - 1)(J - 1) = IJ - 1$$

と一致し, (p_{ij}) と $(\psi_0, \psi_{1i}, \psi_{2j}, \lambda_{ij})$ の変換は 1 対 1.

- 逆変換はやや複雑 (省略)

37

- 対数線形モデルでは, 母数はそれぞれ,

- ψ_{1i} : 因子 1 の水準 i の主効果 (幾何の成績 i の主効果)
- ψ_{2j} : 因子 2 の水準 j の主効果 (推測の成績 j の主効果)
- λ_{ij} : 因子 1, 2 の水準組合せ (i, j) の交互作用 (幾何と推測の成績が (i, j) の交互作用)

と解釈できる. (これも, 統計学の伝統的な用語.)

- 独立性の仮説は,

$$H_0 : p_{ij} = \alpha_i \beta_j \text{ for all } (i, j)$$

と書ける. これは, 新しい母数について,

$$H_0 : \lambda_{ij} = 0 \text{ for all } (i, j)$$

と書き直せる.

38

- 局外母数は

$$\psi_0, \psi_{11}, \dots, \psi_{1I}, \psi_{21}, \dots, \psi_{2J}$$

- [定理] 局外母数の十分統計量は,

$$x_{++}, x_{1+}, \dots, x_{I+}, x_{+1}, \dots, x_{+J}$$

つまり, 2 元分割表の行和と列和になる.

- [定理] H_0 のもとでの条件付分布は, 多項超幾何分布

$$p(\mathbf{x} \mid x_{i+}, x_{+j}, \lambda_{ij} = 0) = \frac{\left(\prod_i x_{i+}! \right) \left(\prod_j x_{+j}! \right)}{x_{++}! \prod_{i,j} x_{ij}!}$$

になる.

- ここまでは, 2×2 のときと同じ.

39

- 本講義では, 対立仮説 H_1 は飽和モデルの場合のみを考える.

- つまり, 飽和モデルの母数を適当に母数変換して (ψ, λ) と書いたとき, 局外母数 ψ の十分統計量を固定した条件付分布にもとづき, 検定問題

$$H_0 : \lambda = (0, \dots, 0)$$

$$H_1 : \lambda \neq (0, \dots, 0)$$

を考える.

- 応用上は, 対立仮説が飽和モデルよりも小さい場合の検定, 例えば, 母数が $(\psi, \lambda_1, \lambda_2)$ と書けるときに

$$H_0 : \lambda_1 = \lambda_2 = 0$$

$$H_1 : \lambda_1 \neq 0, \lambda_2 = 0$$

のような検定も重要であるが, 本講義では省略する.

40

- 対立仮説が飽和モデルである検定は, 一般に, 適合度検定とよばれる.

- H_0 が表すモデルの自由度は,

飽和モデルの自由度 $- \lambda$ の次元,

つまり, 局外母数 ψ の次元.

- 検定統計量の自由度は, H_1, H_0 の表すモデルの自由度の差, つまり, λ の次元となる.

41

- 2 元分割表の独立性の問題の場合, 検定問題は

$$H_0 : \lambda_{ij} = 0 \text{ for all } (i, j)$$

$$H_1 : \lambda_{ij} \neq 0 \text{ for some } (i, j)$$

となる.

- 代表的な検定:

- カイ 2 乗適合度検定

$$\chi^2(\mathbf{x}) = \sum_i \sum_j \frac{(x_{ij} - m_{ij})^2}{m_{ij}} > c_\alpha \Rightarrow H_0 \text{ reject}$$

- 尤度比検定

$$G^2(\mathbf{x}) = 2 \sum_i \sum_j x_{ij} \log \frac{x_{ij}}{m_{ij}} > c_\alpha \Rightarrow H_0 \text{ reject}$$

42

- ここで、 $m = (m_{ij})$ は、 H_0 のもとでの当てはめ値 (fitted value). 多項分布モデルでは $E(X_{ij}) = np_{ij}$ であるから、 m は

$$m_{ij} = n\hat{p}_{ij}$$

と定義される。ただし、 \hat{p}_{ij} は H_0 のもとでの p_{ij} の最尤推定値。つまり、対数尤度関数

$$L = \text{Const} + \sum_{i,j} x_{ij} \log p_{ij}$$

を、 $\lambda = (0, \dots, 0)$ の制約下で最大化したもの。

- 当てはめ値は、一般には、**比例反復法**で求める。2元分割表の独立モデルでは、

$$m_{ij} = \frac{x_{i+}x_{+j}}{x_{++}}$$

となる。

43

観測値 $x^o = (x_{ij}^o)$

	5	4	3	2	1-	計
5	2	1	1	0	0	4
4	8	3	3	0	0	14
3	0	2	1	1	1	5
2	0	0	0	1	1	2
1-	0	0	0	0	1	1
計	10	6	5	2	3	26

44

H_0 のもとでの当てはめ値 $m = (m_{ij})$

	5	4	3	2	1-	計
5	1.54	0.92	0.77	0.31	0.46	4
4	5.38	3.23	2.69	1.08	1.62	14
3	1.92	1.15	0.96	0.38	0.58	5
2	0.77	0.46	0.38	0.15	0.23	2
1-	0.38	0.23	0.19	0.08	0.12	1
計	10	6	5	2	3	26

$$\chi^2(x^o) = \sum_i \sum_j \frac{(x_{ij}^o - m_{ij})^2}{m_{ij}} = \frac{(2 - 1.54)^2}{1.54} + \dots + \frac{(1 - 0.12)^2}{0.12} = 25.338$$

45

- 有意性の評価 (p 値の計算)

... $\chi^2(x^o) = 25.338$ は有意に大きいか?

- 漸近分布論の利用
- 正確法
- モンテカルロ法

46

- 漸近分布論の利用

- H_0 のもとで、 $\chi^2(x)$ は自由度 $(5-1)(5-1) = 16$ のカイ2乗分布に漸近的にしたがう。
(漸近的: ある正則条件のもとで、 $n \rightarrow \infty$ とした極限)

↓

- 自由度 16 のカイ2乗分布の上側 α パーセント点と $\chi^2(x^o)$ を比較する。
 $\chi_{16,0.05} = 26.30$ なので、有意水準 5% では H_0 は棄却されない

- 欠点: 疎 (sparse) な分割表に対する漸近分布論の当てはまりは悪い。

47

- R での計算

```
> data <- matrix(c(2,1,1,0,0,8,3,3,0,0,0,2,1,1,1,0,0,0,1,1,0,0,0,0,1),
  byrow=T,ncol=5)
> data
  [,1] [,2] [,3] [,4] [,5]
[1,]  2  1  1  0  0
[2,]  8  3  3  0  0
[3,]  0  2  1  1  1
[4,]  0  0  0  1  1
[5,]  0  0  0  0  1
> res <- chisq.test(data)
Warning message:
カイ自乗近似は不正確かもしれません in: chisq.test(data)
> res

Pearson's Chi-squared test

data: data
X-squared = 25.3376, df = 16, p-value = 0.06409
```

48

```

> qchisq(0.95,16)
[1] 26.29623
> res$expected
      [,1]      [,2]      [,3]      [,4]      [,5]
[1,] 1.5384615 0.9230769 0.7692308 0.30769231 0.4615385
[2,] 5.3846154 3.2307692 2.6923077 1.07692308 1.6153846
[3,] 1.9230769 1.1538462 0.9615385 0.38461538 0.5769231
[4,] 0.7692308 0.4615385 0.3846154 0.15384615 0.2307692
[5,] 0.3846154 0.2307692 0.1923077 0.07692308 0.1153846
>

```

49

(b) 正確法

- \mathbf{H}_0 のもとでの \mathbf{x} の分布 (超幾何分布) :

$$p(\mathbf{x}) = \frac{\left(\prod_i x_{i+}!\right) \left(\prod_j x_{+j}!\right)}{x_{++}! \prod_{i,j} x_{ij}!}$$

- 行和, 列和が \mathbf{x}^o と等しい分割表の集合 :

$$\mathcal{F} = \left\{ \mathbf{x} \mid x_{i+} = x_{i+}^o, x_{+j} = x_{+j}^o, x_{ij} \in \{0, 1, 2, \dots\} \right\}$$

50

- 適合度カイ 2 乗検定の正確な p 値 :

$$p = \Pr(\chi^2(\mathbf{x}) \geq \chi^2(\mathbf{x}^o) \mid \mathbf{H}_0) = \sum_{\mathbf{x} \in \mathcal{F}} g(\mathbf{x}) p(\mathbf{x}),$$

$$g(\mathbf{x}) = \begin{cases} 1, & \text{if } \chi^2(\mathbf{x}) \geq \chi^2(\mathbf{x}^o), \\ 0, & \text{otherwise} \end{cases}$$

- 欠点 : $\#\mathcal{F}$ が大きいと計算量が膨大になる。

51

$$p(\mathbf{x}) = \frac{(4!14!5!2!1!)(10!6!5!2!3!)}{26!} \prod_{i=1}^5 \prod_{j=1}^5 \frac{1}{x_{ij}!}$$

$$\mathcal{F} = \left\{ \mathbf{x} = (x_{ij}) \begin{array}{|c|c|c|c|c|c|} \hline x_{11} & x_{12} & x_{13} & x_{14} & x_{15} & 4 \\ \hline x_{21} & x_{22} & x_{23} & x_{24} & x_{25} & 14 \\ \hline x_{31} & x_{32} & x_{33} & x_{34} & x_{35} & 5 \\ \hline x_{41} & x_{42} & x_{43} & x_{44} & x_{45} & 2 \\ \hline x_{51} & x_{52} & x_{53} & x_{54} & x_{55} & 1 \\ \hline 10 & 6 & 5 & 2 & 3 & 26 \\ \hline \end{array} \right\} (\#\mathcal{F} = 229, 174)$$

52

$$\chi^2(\mathbf{x}^o) = 25.338$$

$$p = \Pr(\chi^2(\mathbf{x}) \geq 25.338 \mid \mathbf{H}_0) = \sum_{\mathbf{x} \in \mathcal{F}} g(\mathbf{x}) p(\mathbf{x}) = 0.0609007$$

$$g(\mathbf{x}) = \begin{cases} 1, & \text{if } \chi^2(\mathbf{x}) \geq 25.338, \\ 0, & \text{otherwise} \end{cases}$$

53

(c) モンテカルロ法

$p = \sum_{\mathbf{x} \in \mathcal{F}} g(\mathbf{x}) p(\mathbf{x})$ の値を, $p(\mathbf{x})$ からのサンプル x_1, \dots, x_N を用いて

$\frac{1}{N} \sum_{i=1}^N g(x_i)$ と推定する。

- N を大きくすれば, 任意の精度で推定可能。
- 任意の統計量に適用可能。
- 欠点 : $p(\mathbf{x})$ の形が複雑だと, サンプルの発生方法は簡単でない。

本講義では, マルコフ連鎖を利用したサンプルの発生方法を紹介する。
(マルコフ連鎖・モンテカルロ法)

54

● マルコフ連鎖・モンテカルロ法 (MCMC 法)

定常分布が H_0 のもとでの条件付分布 $p(x)$ に一致するマルコフ連鎖を構成する。

適当な初期値から、十分大きな回数 (例えば 50,000 回, 100,000 回など) の状態推移を行えば、その時点ではマルコフ連鎖は定常分布に収束しているとみなせるので、そこから先の状態推移を定常分布からのサンプルとして使用する。

$$x^{(0)} \rightarrow x^{(1)} \rightarrow x^{(2)} \rightarrow \dots \rightarrow \underbrace{x^{(100000)} \rightarrow x^{(100001)} \rightarrow \dots}_{\text{定常分布からのサンプル}}$$

Metropolis-Hastings アルゴリズムにより、

\mathcal{F} 上で連結な任意のマルコフ連鎖から、定常分布が $p(x)$ に一致するマルコフ連鎖を構成できる。

- $\mathcal{F} = \{x_1, \dots, x_s\}$ (適当に番号付けられているとする.)
- $\pi = (\pi_1, \dots, \pi_s) : H_0$ のもとでの条件付分布 $\pi_i = p(x_i)$
- $\{Z_t, t = 0, 1, 2, \dots\} : \text{確率過程}$
- 推移確率行列 $Q = (q_{ij}) :$

$$q_{ij} = \Pr(Z_{t+1} = x_j \mid Z_t = x_i)$$

- π が平衡方程式

$$\pi = \pi Q$$

を満たすとき、 π を定常分布とよぶ。

- [定理](Metropolis-Hastings)

$R = (r_{ij})$ を、 \mathcal{F} 上の連結、対称な任意のマルコフ連鎖の推移確率行列とする。このとき、 $Q = (q_{ij})$ を、

$$q_{ij} = r_{ij} \min\left(1, \frac{\pi_j}{\pi_i}\right), \quad i \neq j,$$

$$q_{ii} = 1 - \sum_{j \neq i} q_{ij}$$

と定義すれば、 Q は平衡方程式を満たす。

- つまり、 \mathcal{F} 上で連結、対称であるようなマルコフ連鎖が (どんなものでもよいからひとつ) 作ればよい。
- 条件付分布の基準化定数の計算は不要。

2. マルコフ基底

マルコフ基底の定義

- $x = (x_{11}, \dots, x_{IJ})'$: 分割表 (頻度ベクトル. 列ベクトルとする)

- $t = Ax$: 固定する十分統計量 (行列の形で書く)

例えば、 2×3 分割表の独立モデルであれば、

$$\begin{pmatrix} x_{1+} \\ x_{2+} \\ x_{+1} \\ x_{+2} \\ x_{+3} \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_{11} \\ x_{12} \\ x_{13} \\ x_{21} \\ x_{22} \\ x_{23} \end{pmatrix}$$

- $\mathcal{F}_t = \{x \geq 0 \mid Ax = t\}$: 十分統計量が t に等しい分割表の集合 (t -fiber)

MCMC 法によって相似検定の p 値を計算するために、観測値 x^o が含まれる t -fiber 上に、連結なマルコフ連鎖を構成したい。

- \mathcal{M} を以下で定義する.

$$\begin{aligned} \mathcal{M} &= \text{Ker}(A) \\ &= \{z \mid Az = 0, z \text{ の成分 } \in \{0, \pm 1, \pm 2, \dots\}\} \\ &= \{ \text{固定する十分統計量がすべてゼロである整数ベクトル} \} \end{aligned}$$

- \mathcal{M} の要素を move と呼ぶ.
- $x, y \in \mathcal{F}_t$ が $B \subset \mathcal{M}$ によって相互到達可能

$$\begin{aligned} \stackrel{\text{def}}{\Leftrightarrow} N > 0, z_j \in B, \varepsilon_j \in \{-1, 1\}, j = 1, \dots, N \text{ が存在して} \\ y = x + \sum_{j=1}^N \varepsilon_j z_j \text{ かつ } x + \sum_{j=1}^n \varepsilon_j z_j \in \mathcal{F}_t, n = 1, \dots, N \\ \text{が成り立つ} \end{aligned}$$

61

- 相互到達可能性は, \mathcal{F}_t 内の同値関係であり, \mathcal{F}_t は B によって排反な同値類に分解される. この同値類を \mathcal{F}_t の B -同値類と呼ぶ.

62

- マルコフ基底 (Diaconis and Sturmfels, 1998)

move の有限集合 $B \subset \mathcal{M}$ が $(A$ に対する) マルコフ基底

$\stackrel{\text{def}}{\Leftrightarrow}$ 任意の t に対して \mathcal{F}_t がそれぞれ自身ひとつの B -同値類となる

- マルコフ基底が求まれば, 任意の観測値に対する t -fiber 上に連結なマルコフ連鎖が構成できるので, MCMC 法により相似検定の p 値が計算できる.
- 任意の A に対して, マルコフ基底が存在する. (ヒルベルトの基底定理)
- 任意の A に対して, マルコフ基底を計算するアルゴリズムがある.
- 本講義では, まず, 分割表のふたつの問題に対して, マルコフ基底を紹介する. その後, マルコフ基底の計算アルゴリズムを紹介する.

63

2元分割表の独立モデルのマルコフ基底

- $x = (x_{11}, \dots, x_{IJ})' : I \times J$ の分割表, $x_{ij} \in \{0, 1, 2, \dots\}$

x_{11}	\dots	x_{1J}
\vdots		\vdots
x_{I1}	\dots	x_{IJ}

- 固定する十分統計量は行和と列和 :

$$t = (x_{1+}, \dots, x_{I+}, x_{+1}, \dots, x_{+J})'$$

- 任意に与えられた $t = Ax^0$ に対して,

$$\mathcal{F}_t = \{x \geq 0 \mid Ax = t\}$$

上に連結なマルコフ連鎖を作りたい.

これは簡単

64

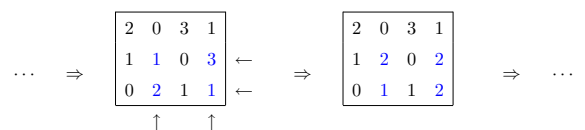
\mathcal{F}_t 上の連結なマルコフ連鎖の構成 :

$\begin{matrix} + & - \\ - & + \end{matrix}$ の形の推移のみで十分 (マルコフ基底をなす).

アルゴリズムの例 :

- $x \in \mathcal{F}_t$: 現在の状態
- 行のペア, 列のペアをランダムに選ぶ.
- $\begin{matrix} + & - \\ - & + \end{matrix}$ or $\begin{matrix} - & + \\ + & - \end{matrix}$ をランダムに選ぶ.
- x の 4 つのセル頻度を更新 $\rightarrow y$
- $y \in \mathcal{F}_t$ ならば y が次の状態
- $y \notin \mathcal{F}_t$ ならば (どこかの頻度が負になったら) x に留まる

65



どんな t に対しても, \mathcal{F}_t 上の連結なマルコフ連鎖となる.

66

3元分割表の無3因子交互作用モデルのマルコフ基底

$$\mathbf{x} = (x_{ijk}), \quad 1 \leq i \leq I, 1 \leq j \leq J, 1 \leq k \leq K$$

$$x_{ijk} \in \{0, 1, 2, \dots\}$$

x_{111}	...	x_{11K}	x_{211}	...	x_{21K}	...	x_{J11}	...	x_{J1K}
\vdots		\vdots	\vdots		\vdots		\vdots		\vdots
x_{1J1}	...	x_{1JK}	x_{2J1}	...	x_{2JK}		x_{JJ1}	...	x_{JJK}

- $t = Ax$: すべての2次元周辺度数 $\{x_{ij+}, \{x_{i+k}, \{x_{+jk}\}$
- 任意に与えられた $t = Ax^o$ に対して,

$$\mathcal{F}_t = \{x \geq 0 \mid Ax = t\}$$

上に連結なマルコフ連鎖を作りたい。

surprisingly difficult !

Remark : これは以下の問題に対応する。

- 対数線形モデル

$$\log p_{ijk} = \mu + \alpha_i + \beta_j + \gamma_k + (\alpha\beta)_{ij} + (\alpha\gamma)_{ik} + (\beta\gamma)_{jk} + (\alpha\beta\gamma)_{ijk}$$

- 帰無仮説は「三因子交互作用が存在しない」

$$H_0 : (\alpha\beta\gamma)_{ijk} = 0, \quad \text{for all } i, j, k$$

- 局外母数 : $\mu, \alpha_i, \beta_j, \gamma_k, (\alpha\beta)_{ij}, (\alpha\gamma)_{ik}, (\beta\gamma)_{jk}$
- 局外母数に対する十分統計量 :

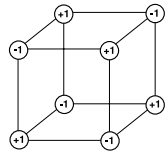
$$x_{+++}, x_{i++}, x_{+j+}, x_{++k}, x_{ij+}, x_{i+k}, x_{+jk}$$

- 条件つき確率関数 : $p(x_{ijk} \mid x_{ij+}, x_{i+k}, x_{+jk}) \propto \left(\prod_{i,j,k} x_{ijk}! \right)^{-1}$

H_0 は分解可能モデルではないため, $p(x_{ijk} \mid x_{ij+}, x_{i+k}, x_{+jk})$ からの直接的なサンプリングも難しい \Rightarrow MCMC 法が使えれば嬉しい

2x2のmoveの単純な拡張 (2x2x2 basic move)

2元分割表における $\begin{matrix} + & - \\ - & + \end{matrix}$ の形の move を3元分割表に自然に拡張し, 十分統計量の値を保存する $2 \times 2 \times 2$ の move を考える。



	$i = i_1$		$i = i_2$		
$j \setminus k$	k_1	k_2	$j \setminus k$	k_1	k_2
j_1	+1	-1	j_1	-1	+1
j_2	-1	+1	j_2	+1	-1

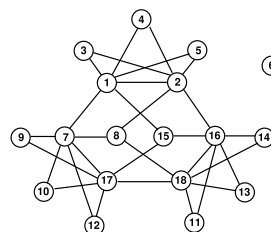
→ 構成されるマルコフ連鎖は, 連結になるとは限らない。
(これだけでは, マルコフ基底にならない.)

例 : $3 \times 3 \times 3$ 分割表 (# $\mathcal{F}_t = 18$)

$$t : \begin{matrix} \begin{matrix} \square & 2 \\ & 1 \\ & 1 \end{matrix} & \begin{matrix} \square & 1 \\ & 2 \\ & 1 \end{matrix} & \begin{matrix} \square & 1 \\ & 1 \\ & 2 \end{matrix} & \begin{matrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{matrix} & \begin{matrix} 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \end{matrix} \end{matrix}$$

1: $\begin{matrix} 200 & 010 & 001 \\ 010 & 110 & 001 \\ 001 & 001 & 110 \end{matrix}$	2: $\begin{matrix} 200 & 010 & 001 \\ 010 & 101 & 010 \\ 001 & 010 & 101 \end{matrix}$	3: $\begin{matrix} 200 & 010 & 001 \\ 010 & 011 & 100 \\ 001 & 100 & 011 \end{matrix}$
4: $\begin{matrix} 200 & 001 & 010 \\ 010 & 110 & 001 \\ 001 & 010 & 101 \end{matrix}$	5: $\begin{matrix} 200 & 010 & 001 \\ 001 & 110 & 010 \\ 010 & 001 & 101 \end{matrix}$	6: $\begin{matrix} 200 & 001 & 010 \\ 001 & 020 & 100 \\ 010 & 100 & 002 \end{matrix}$
7: $\begin{matrix} 110 & 100 & 001 \\ 100 & 020 & 001 \\ 001 & 001 & 110 \end{matrix}$	8: $\begin{matrix} 110 & 100 & 001 \\ 100 & 011 & 010 \\ 001 & 010 & 101 \end{matrix}$	9: $\begin{matrix} 110 & 001 & 100 \\ 100 & 020 & 001 \\ 001 & 100 & 011 \end{matrix}$
10: $\begin{matrix} 110 & 100 & 001 \\ 001 & 020 & 100 \\ 100 & 001 & 011 \end{matrix}$	11: $\begin{matrix} 110 & 001 & 100 \\ 001 & 110 & 010 \\ 100 & 010 & 002 \end{matrix}$	12: $\begin{matrix} 101 & 100 & 010 \\ 100 & 020 & 001 \\ 010 & 001 & 101 \end{matrix}$
13: $\begin{matrix} 101 & 010 & 100 \\ 100 & 011 & 010 \\ 010 & 100 & 002 \end{matrix}$	14: $\begin{matrix} 101 & 100 & 010 \\ 010 & 011 & 100 \\ 100 & 010 & 002 \end{matrix}$	15: $\begin{matrix} 101 & 010 & 100 \\ 010 & 110 & 001 \\ 100 & 001 & 011 \end{matrix}$
16: $\begin{matrix} 101 & 010 & 100 \\ 010 & 101 & 010 \\ 100 & 010 & 002 \end{matrix}$	17: $\begin{matrix} 011 & 100 & 100 \\ 100 & 020 & 001 \\ 100 & 001 & 011 \end{matrix}$	18: $\begin{matrix} 011 & 100 & 100 \\ 100 & 011 & 010 \\ 100 & 010 & 002 \end{matrix}$

basic move から構成される推移グラフ



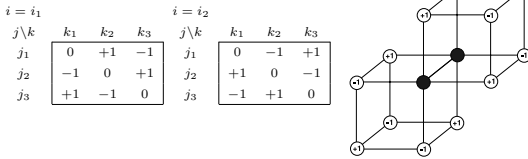
状態 6: $\begin{matrix} 200 & 001 & 010 \\ 001 & 020 & 100 \\ 010 & 100 & 002 \end{matrix}$

は他の状態と相互到達不可能。
 i, j, k のペアをどう選んで

	$i = i_1$		$i = i_2$		
$j \setminus k$	k_1	k_2	$j \setminus k$	k_1	k_2
j_1	+1	-1	j_1	-1	+1
j_2	-1	+1	j_2	+1	-1

を加えても, どこかの頻度が -1 になってしまう。

3 × 3 × 3 表のマルコフ基底には、6 次の move:



が必要。この形の推移を許せば、先ほどの例は連結になることが観察される。実際、{basic move, 6 次の move} は、この問題に対するマルコフ基底になる。

マルコフ基底とイデアル

- 分割表: $x = (x_1, \dots, x_p)$, 十分統計量: $t = (t_1, \dots, t_d)$
 $t = Ax$, $A = (a_{ij})$: $d \times p$ 行列
- $k[u_1, \dots, u_p]$: 体 k 上の多項式環
 $u = \{u_1, \dots, u_p\}$: 不定元
- 分割表 x と単項式の対応

$$x \Leftrightarrow u_1^{x_1} \dots u_p^{x_p} = u^x$$

- move $z = z^+ - z^-$ と二項式の対応

$$z \Leftrightarrow u^{z^+} - u^{z^-}$$

- 例: $u = \{u_{11}, \dots, u_{33}\}$

$$x: \begin{pmatrix} 2 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 2 \end{pmatrix} \Leftrightarrow u^x = u_{11}^2 u_{13} u_{21} u_{22} u_{32} u_{33}^2$$

$$z: \begin{pmatrix} 2 & -2 & 0 \\ -1 & 0 & 1 \\ -1 & 2 & -1 \end{pmatrix} \Leftrightarrow u^{z^+} - u^{z^-} = u_{11}^2 u_{23} u_{32}^2 - u_{12}^2 u_{21} u_{31} u_{33}$$

- $v = \{v_1, \dots, v_d\}$: t_1, \dots, t_d に対応する不定元
- $t = Ax$ は次の準同型写像で表現される

$$\begin{aligned} \phi_A: k[u] &\rightarrow k[v] \\ u_j &\mapsto v_1^{a_{1j}} v_2^{a_{2j}} \dots v_d^{a_{dj}} \end{aligned}$$

- ϕ_A の核

$$I_A = \ker(A) = \{f \in k[u] \mid \phi(k[u]) = 0\}$$

は、 A に対するトーリックイデアルと呼ばれる。

- Point:

$$z \text{ が } A \text{ の move} \Leftrightarrow u^{z^+} - u^{z^-} \in I_A$$

- 例: 2 × 2 分割表

$(v_1, v_2, v_3, v_4) = (r_1, r_2, c_1, c_2)$ と書けば,

$$\begin{aligned} \phi_A(u_{11}) &= r_1 c_1, & \phi_A(u_{12}) &= r_1 c_2, \\ \phi_A(u_{21}) &= r_2 c_1, & \phi_A(u_{22}) &= r_2 c_2, \end{aligned}$$

$$\begin{aligned} \phi_A(u_{11}u_{22} - u_{12}u_{21}) &= \phi_A(u_{11})\phi_A(u_{22}) - \phi_A(u_{12})\phi_A(u_{21}) \\ &= r_1 r_2 c_1 c_2 - r_1 r_2 c_1 c_2 = 0 \end{aligned}$$

i.e., $u_{11}u_{22} - u_{12}u_{21} \in I_A$.

二項式 $u_{11}u_{22} - u_{12}u_{21}$ は move $\begin{pmatrix} +1 & -1 \\ -1 & +1 \end{pmatrix}$ に対応

- [定理](Diaconis and Sturmfels, 1998)

$$\begin{aligned} B = \{z_1, \dots, z_L\}, z_i \in \mathcal{M} \text{ が } A \text{ に対するマルコフ基底} \\ \Leftrightarrow I_A = \langle u^{z_i} - u^{z_i'}, i = 1, \dots, L \rangle \end{aligned}$$

- [補題](Hibi, 2003) I_A は二項式により生成される。

- マルコフ基底の計算 (消去定理)

$$I_A^* = \langle -\phi_A(u_i) + u_i, i = 1, \dots, p \rangle \subset k[u, v]$$

と定義すれば, $I_A = I_A^* \cap k[u]$ が成り立つ。よって, $\{t_1, \dots, t_d\} \succ \{u_1, \dots, u_p\}$ となるような適当な項順序のもとで, I_A^* のグレブナー基底 G^* が求まれば, $G^* \cap k[u]$ が I_A のグレブナー基底になる。

● Risa/Asir での計算

```
[0] load("gr")$
[86] Polys = [-x11+r1*c1,-x12+r1*c2,-x13+r1*c3,
            -x21+r2*c1,-x22+r2*c2,-x23+r2*c3];
[-x11+c1*r1,c2*r1-x12,c3*r1-x13,r2*c1-x21,r2*c2-x22,r2*c3-x23]
[87] Q = hgr(Polys, [r1,r2,c1,c2,c3,x11,x12,x13,x21,x22,x23],
            [[2,5],[2,6]])$
[88] for(J=0; J<length(Q); J++) print([J,Q[J]]);
[0,x23*x12-x22*x13]
[1,x23*x11-x21*x13]
[2,x22*x11-x21*x12]
[3,x23*c2-x22*c3]
[4,-c3*x12+x13*c2]
[5,x23*c1-x21*c3]
[6,x22*c1-x21*c2]
(中略)
[16,c2*r1-x12]
[17,-x11+c1*r1]
[89]
```

● 4ti2 での計算 (演習問題 1(b))

```
$ less 2x3-indep.mat
5 6
1 1 1 0 0 0
0 0 0 1 1 1
1 0 0 1 0 0
0 1 0 0 1 0
0 0 1 0 0 1

$ ./markov 2x3-indep.mat

(計算)

$ less 2x3-indep.mat.mar
3 6
0 -1 1 0 1 -1
1 -1 0 -1 1 0
1 0 -1 -1 0 1
```

演習問題 2(a)(b) も同様に計算できるので確かめよ.

3. 実験計画とマルコフ基底

2 水準実験・組合せ配置

- 本講義では、2 水準・多因子の実験計画データの解析を、マルコフ連鎖・モンテカルロ法で行う方法を紹介する。
 - 実験計画データの例: 噴流式はんだ付け装置の不良品数データ (7 因子, すべて 2 水準)
 - A: prebake condition
 - B: flux density L. W. Condra,
 - C: conveyer speed Reliability Improvement with
 - D: preheat condition Design of Experiments, 1993.
 - E: cooling time
 - F: ultrasonic solder agitator
 - G: solder temperature
- 観測値: 不良品数

- まず、組合せ配置の説明をする。

とりあえず、7 因子のうち A,B,C,D の 4 因子のみに注目し、不良品数の以下のデータが得られたと仮定する。

run	A	B	C	D	x
1	+1	+1	+1	+1	$x_1 = 69$
2	+1	+1	+1	-1	$x_2 = 31$
3	+1	+1	-1	+1	$x_3 = 55$
4	+1	+1	-1	-1	$x_4 = 149$
5	+1	-1	+1	+1	$x_5 = 46$
6	+1	-1	+1	-1	$x_6 = 43$
7	+1	-1	-1	+1	$x_7 = 118$
8	+1	-1	-1	-1	$x_8 = 30$
9	-1	+1	+1	+1	$x_9 = 43$
10	-1	+1	+1	-1	$x_{10} = 45$
11	-1	+1	-1	+1	$x_{11} = 71$
12	-1	+1	-1	-1	$x_{12} = 380$
13	-1	-1	+1	+1	$x_{13} = 37$
14	-1	-1	+1	-1	$x_{14} = 36$
15	-1	-1	-1	+1	$x_{15} = 212$
16	-1	-1	-1	-1	$x_{16} = 52$

- このような、因子のすべての水準の組合せについて観測を行う計画を、完全実施要因計画（または、組合せ配置計画）とよぶ。
- 実験の目的: 不良品数を減らすような水準の選択
- 因子の数を p とすれば、組合せ配置の実験回数 k は $k = 2^p$.
- 計画行列 D : $\{+1, -1\}$ を要素とする $k \times p$ 行列

85

$$D = \begin{pmatrix} +1 & +1 & +1 & +1 \\ +1 & +1 & +1 & -1 \\ +1 & +1 & -1 & +1 \\ +1 & +1 & -1 & -1 \\ +1 & -1 & +1 & +1 \\ +1 & -1 & +1 & -1 \\ +1 & -1 & -1 & +1 \\ +1 & -1 & -1 & -1 \\ -1 & +1 & +1 & +1 \\ -1 & +1 & +1 & -1 \\ -1 & +1 & -1 & +1 \\ -1 & +1 & -1 & -1 \\ -1 & -1 & +1 & +1 \\ -1 & -1 & +1 & -1 \\ -1 & -1 & -1 & +1 \\ -1 & -1 & -1 & -1 \end{pmatrix} = (d_1 \quad d_2 \quad d_3 \quad d_4)$$

86

- p 因子の組合せ配置は、 2^p 型 p 元分割表データに他ならない。
 $p = 4$ 因子の例では、

$$X = (X_{1111}, X_{1112}, X_{1121}, X_{1122}, \dots, X_{2222})'$$

- したがって、前章で説明した分割表に対する条件付検定の方法でモデルの当てはまりが評価できる。
- 確率分布の仮定: 独立なポアソン分布 (2×2 分割表の例 3 と同じ)

$$X_{abcd} \sim \text{Po}(\mu_{abcd}), \quad a, b, c, d = 1, 2, \text{ i.i.d.}$$

87

- 対数線形モデル (飽和モデルの母数の母数変換)

$$\begin{aligned} \log \mu_{abcd} = & \kappa + \alpha_a + \beta_b + \gamma_c + \delta_d + (\alpha\beta)_{ab} + (\alpha\gamma)_{ac} + (\alpha\delta)_{ad} \\ & + (\beta\gamma)_{bc} + (\beta\delta)_{bd} + (\gamma\delta)_{cd} + (\alpha\beta\gamma)_{abc} + (\alpha\beta\delta)_{abd} \\ & + (\alpha\gamma\delta)_{acd} + (\beta\gamma\delta)_{bcd} + (\alpha\beta\gamma\delta)_{abcd} \end{aligned}$$

- 飽和モデルと 1 対 1 対応とするために、通常は、

$$\begin{aligned} \sum_a \alpha_a = \dots = \sum_d \delta_d = 0 \\ \sum_a (\alpha\beta)_{ab} = \sum_b (\alpha\beta)_{ab} = \dots = \sum_c (\gamma\delta)_{cd} = \sum_d (\gamma\delta)_{cd} = 0 \\ \sum_a (\alpha\beta\gamma)_{abc} = \sum_b (\alpha\beta\gamma)_{abc} = \dots = \sum_d (\beta\gamma\delta)_{bcd} = 0 \\ \sum_a (\alpha\beta\gamma\delta)_{abcd} = \dots = \sum_d (\alpha\beta\gamma\delta)_{abcd} = 0 \end{aligned}$$

の制約をつける。しかし、この問題は 2 水準であるので、右辺の各項の母数の自由度はすべて 1 になる。

88

- 例えば、 $(\alpha\beta)_{ab}$ は、因子 A と因子 B の水準組合せ (a, b) の 2 因子交互作用であるが、

$$\sum_a (\alpha\beta)_{ab} = \sum_b (\alpha\beta)_{ab} = 0$$

より、

$$\begin{aligned} (\alpha\beta)_{11} = \psi_{AB} \quad (\alpha\beta)_{12} = -\psi_{AB} \\ (\alpha\beta)_{21} = -\psi_{AB} \quad (\alpha\beta)_{22} = \psi_{AB} \end{aligned}$$

と書くことができる。

主効果, 3 因子交互作用, 4 因子交互作用についてもすべて同様。

89

- 結局、対数線形モデルの飽和モデルは、

- ベクトル $(1, \dots, 1)'$
- 計画行列 $D = (d_1, \dots, d_4)$ の各列ベクトル, d_1, \dots, d_4
- D の 2 列の成分ごとの積, $d_1 \odot d_2, d_1 \odot d_3, \dots, d_3 \odot d_4$
- D の 3 列の成分ごとの積, $d_1 \odot d_2 \odot d_3, \dots, d_2 \odot d_3 \odot d_4$
- D の 4 列の成分ごとの積, $d_1 \odot d_2 \odot d_3 \odot d_4$

をすべて (適当な順番で) 並べた $k \times k$ 行列 M をもちいて、

90

$$\begin{pmatrix} \log \mu_{1111} \\ \log \mu_{1112} \\ \log \mu_{1121} \\ \log \mu_{1122} \\ \vdots \\ \log \mu_{2211} \\ \log \mu_{2212} \\ \log \mu_{2221} \\ \log \mu_{2222} \end{pmatrix} = M \begin{pmatrix} \psi_0 \\ \psi_A \\ \vdots \\ \psi_D \\ \psi_{AB} \\ \vdots \\ \psi_{CD} \\ \psi_{ABC} \\ \vdots \\ \psi_{BCD} \\ \psi_{ABCD} \end{pmatrix}$$

と書ける。行列 M は、共変量行列 (モデル行列) とよばれる。

91

$$M = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & -1 & 1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & -1 & 1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \end{pmatrix}$$

92

- M は、次数 16 のアダマール行列。

- [定理] $M'X$ (の各成分) は、母数

$$\psi = (\psi_0, \psi_A, \dots, \psi_D, \psi_{AB}, \dots, \psi_{CD}, \psi_{ABC}, \dots, \psi_{BCD}, \psi_{ABCD})'$$

(の各成分) の十分統計量。

- 例えば,

- ψ_0 の十分統計量は $(1, \dots, 1)X = \sum_{a,b,c,d} X_{abcd}$

- ψ_A (因子 A の主効果) の十分統計量は,

$$\begin{aligned} d_1'X &= (X_{1111} + \dots + X_{1222}) - (X_{2111} + \dots + X_{2222}) \\ &= \sum (\text{A の水準が 1 の観測値}) - \sum (\text{A の水準が 2 の観測値}) \end{aligned}$$

93

- ψ_{AB} (因子 A, B の 2 因子交互作用) の十分統計量は,

$$\begin{aligned} &(d_1 \odot d_2)'X \\ &= \sum ((A, B) \text{ の水準が } (1, 1), (2, 2) \text{ の観測値}) \\ &\quad - \sum ((A, B) \text{ の水準が } (1, 2), (2, 1) \text{ の観測値}) \end{aligned}$$

- ψ_{ABC} (因子 A, B, C の 3 因子交互作用) の十分統計量は,

$$\begin{aligned} &(d_1 \odot d_2 \odot d_3)'X \\ &= \sum ((A, B, C) \text{ の水準が } (1, 1, 1), (1, 2, 2), (2, 1, 2), (2, 2, 1) \text{ の観測値}) \\ &\quad - \sum ((A, B, C) \text{ の水準が } (1, 1, 2), (1, 2, 1), (2, 1, 1), (2, 2, 2) \text{ の観測値}) \end{aligned}$$

という具合。

94

- 検証したいモデル (H_0) は、対数線形モデルの左辺の母数のいくつかを $= 0$ とおいたもの。

ただし、 $= 0$ とおく母数は、すべての組合せを許すわけではなく、以下のクラスに属するものに限る。

- [定義] (階層モデル) 対数線形モデルにおいて、

「高次の交互作用がモデルに含まれる」

\Rightarrow 「それに含まれるすべての低次の交互作用がモデルに含まれる」が満たされるモデルを、階層モデルとよぶ。

- 例えば、 ψ_{ABC} を含む階層モデルは、それより低次の

$$\psi_A, \psi_B, \psi_C, \psi_{AB}, \psi_{AC}, \psi_{BC}$$

をすべて含む。

- 検証するモデルを階層モデルに限ることは、解釈の面からも自然。

95

- 階層モデルは、モデルに含まれる極大な交互作用の集合 (生成集合) で表記できる。

- 例えば、モデル ABC/ABD の適合度検定は,

$$H_0 : \psi_{CD} = \psi_{ACD} = \psi_{BCD} = \psi_{ABCD} = 0$$

となる。

このモデルの自由度: $(16 - 1) - 4 = 11$

検定統計量の自由度: 4

- R で計算する場合は、一般化線形モデル (generalized linear model) の関数 `glm` を使うとよい。(尤度比検定統計量の値を、デフォルトで計算してくれる。)

96

```

> data4 <- read.table("data4.txt", header=T)
> data4
  A B C D x
1  1 1 1 1 69
2  1 1 1 -1 31
3  1 1 -1 1 55
4  1 1 -1 -1 149
5  1 -1 1 1 46
6  1 -1 1 -1 43
7  1 -1 -1 1 118
8  1 -1 -1 -1 30
9 -1 1 1 1 43
10 -1 1 1 -1 45
11 -1 1 -1 1 71
12 -1 1 -1 -1 380
13 -1 -1 1 1 37
14 -1 -1 1 -1 36
15 -1 -1 -1 1 212
16 -1 -1 -1 -1 52

```

97

```

> data4.glm <- glm(x~A+B+C+D+A*B+A*C+A*D+B*C+B*D+A*B*C+A*B*D,
  data4,family="poisson")
> summary(data4.glm)

Call:
glm(formula = x ~ A + B + C + D + A * B + A * C + A * D + B *
  C + B * D + A * B * C + A * B * D, family = "poisson", data = data4)

Deviance Residuals:
    1     2     3     4     5     6     7     8 
4.015 -4.037 -3.298  2.476 -2.080  2.746  1.503 -2.463 
    9    10    11    12    13    14    15    16 
4.821 -3.131 -2.617  1.279 -2.446  3.450  1.196 -2.133 

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  4.182653   0.034131 122.546 < 2e-16 ***
A             -0.072021   0.034131  -2.110  0.03485 *
B              0.142381   0.034131   4.172 3.03e-05 ***

```

98

```

C             -0.517643   0.031613 -16.374 < 2e-16 ***
D              0.020120   0.030598   0.658  0.51083
A:B            -0.001776   0.034131  -0.052  0.95850
A:C             0.212262   0.031613   6.714 1.89e-11 ***
A:D             0.089063   0.030598   2.911  0.00361 **
B:C            -0.069127   0.031613  -2.187  0.02877 *
B:D            -0.442261   0.030598 -14.454 < 2e-16 ***
A:B:C           0.018033   0.031613   0.570  0.56838
A:B:D           0.146741   0.030598   4.796 1.62e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

Null deviance: 1021.25 on 15 degrees of freedom
Residual deviance: 135.02 on 4 degrees of freedom
AIC: 255.19

Number of Fisher Scoring iterations: 5

```

99

```

> fitted(data4.glm)
      1      2      3      4      5      6
40.78947 59.21053 83.21053 120.78947 61.58650 27.41350
      7      8      9     10     11     12
102.41350 45.58650 18.61224 69.38776 95.38776 355.61224
      13     14     15     16
53.93769 19.06231 195.06231 68.93769
>
> 1-pchisq(135.02, 4)
[1] 0
>
> qchisq(0.95, 4)
[1] 9.487729
>

```

100

- p 値の評価の手法については、分割表の解析と全く同様のことがいえる。すなわち、
 - 漸近分布論(カイ2乗近似)を利用するのがもっとも簡単。ただし、得られたデータによっては、漸近分布論の当てはまりが悪いかもしれない。
 - 可能であれば、正確法により評価したい。ただし、データのサイズが大きければ、計算量の問題が生じる。
 - 正確法が難しい場合は、マルコフ連鎖・モンテカルロ法が有用。

101

- 十分統計量を固定した X の条件付分布:

$$p(\mathbf{x} \mid M'\mathbf{x} = M'\mathbf{x}^o) = C(M'\mathbf{x}^o) \prod_{i=1}^k \frac{1}{x_i!},$$

\mathbf{x}^o : 観測されたデータ,

$C(M'\mathbf{x}^o)$: 規格化定数

$$C(M'\mathbf{x}^o)^{-1} = \sum_{\mathbf{x} \in \mathcal{F}(M'\mathbf{x}^o)} \left(\prod_{i=1}^k \frac{1}{x_i!} \right),$$

$$\mathcal{F}(M'\mathbf{x}^o) = \{\mathbf{x} \mid M'\mathbf{x} = M'\mathbf{x}^o, \mathbf{x} \in \{0, 1, 2, \dots\}^k\}.$$

102

- つまり、共変量行列 M の転置値を配置とみて、十分統計量の値 $M'x^o$ を共有する空間 $\mathcal{F}(M'x^o)$ を連結に繋ぐ基底 (M' に対するマルコフ基底) を求めればよい。
- マルコフ基底の計算には、4ti2 が便利。

2 水準実験・一部実施計画

- 組合せ配置は、因子の数が増えると実験回数が膨大になる。
- 例えば、噴流式はんだ付け装置の不良品数データは、7 因子実験であるので、組合せ配置を行うためには $2^7 = 128$ 回の測定が必要となり、コストがかかる。
- 実験回数を減らすために有効な方法のひとつに、別名関係 (定義関係) を満たす水準組合せの実験だけをおこなう、という方法がある。これは、レギュラーな一部実施計画とよばれる。
- レギュラーな一部実施計画では、実験回数は、組合せ配置の $\frac{1}{2^q}$ 倍となる。ただし、 q は独立な別名関係の数。

- レギュラーな一部実施計画は、実験回数が 2^{p-q} の形に限られるという欠点を除けば、統計理論におけるさまざまなよい性質をもつ、きわめて有効な方法。
- 本講義では、レギュラーな一部実施計画で得られた実験データについて、これまでに述べた方法で相似検定を実行する方法を説明する。

- 噴流式はんだ付け装置の不良品数データ

run	A	B	C	D	E	F	G	\bar{x}
1	+1	+1	+1	+1	+1	+1	+1	$\bar{x}_1 = 69$
2	+1	+1	+1	-1	-1	-1	-1	$\bar{x}_2 = 31$
3	+1	+1	-1	+1	+1	-1	-1	$\bar{x}_3 = 55$
4	+1	+1	-1	-1	-1	+1	+1	$\bar{x}_4 = 149$
5	+1	-1	+1	+1	-1	+1	-1	$\bar{x}_5 = 46$
6	+1	-1	+1	-1	+1	-1	+1	$\bar{x}_6 = 43$
7	+1	-1	-1	+1	-1	-1	+1	$\bar{x}_7 = 118$
8	+1	-1	-1	-1	+1	+1	-1	$\bar{x}_8 = 30$
9	-1	+1	+1	+1	-1	-1	+1	$\bar{x}_9 = 43$
10	-1	+1	+1	-1	+1	+1	-1	$\bar{x}_{10} = 45$
11	-1	+1	-1	+1	-1	+1	-1	$\bar{x}_{11} = 71$
12	-1	+1	-1	-1	+1	-1	+1	$\bar{x}_{12} = 380$
13	-1	-1	+1	+1	+1	-1	-1	$\bar{x}_{13} = 37$
14	-1	-1	+1	-1	-1	+1	+1	$\bar{x}_{14} = 36$
15	-1	-1	-1	+1	+1	+1	+1	$\bar{x}_{15} = 212$
16	-1	-1	-1	-1	-1	-1	-1	$\bar{x}_{16} = 52$

別名関係: $ABDE = ACDF = BCDG = I$

- 別名関係
 $ABDE = ACDF = BCDG = I$
を満たす 16 点でのみ測定を行っている。
 2^{7-3} 一部実施要因計画とよばれる。
- k : 実験回数
 p : 制御因子の数 (すべて 2 水準)
 q : 別名関係の数
- $k = 2^{p-q}$ の関係がある。
噴流式はんだ付け装置の実験では、 $p = 7, q = 3, k = 2^{7-3} = 16$ 。
- 計画行列: $\{+1, -1\}$ を要素とする $k \times p$ 行列

- 計画行列

$$D = \begin{pmatrix} +1 & +1 & +1 & +1 & +1 & +1 & +1 \\ +1 & +1 & +1 & -1 & -1 & -1 & -1 \\ +1 & +1 & -1 & +1 & +1 & -1 & -1 \\ +1 & +1 & -1 & -1 & -1 & +1 & +1 \\ +1 & -1 & +1 & +1 & -1 & +1 & -1 \\ +1 & -1 & +1 & -1 & +1 & -1 & +1 \\ +1 & -1 & -1 & +1 & -1 & -1 & +1 \\ +1 & -1 & -1 & -1 & +1 & +1 & -1 \\ -1 & +1 & +1 & +1 & -1 & -1 & +1 \\ -1 & +1 & +1 & -1 & +1 & +1 & -1 \\ & & & & & & \vdots \\ -1 & -1 & -1 & +1 & +1 & +1 & +1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 \end{pmatrix} = (d_1 \quad \dots \quad d_7)$$

- 考え方: 計画行列 D から, 検証したいモデル (H_0) の共変量行列 (モデル行列) M を作成し, 配置 M' に対するマルコフ基底を求める.
- 条件付分布やファイバーなど, 組合せ配置のときと全く同じ.
- 注意すべき点はひとつだけ:
検証するモデルにおいて, すべての母数が推定可能であるか?
- 母数の同時推定可能性は, 別名関係

$$ABDE = ACDF = BCDG = I$$

から判定できる.

109

- 4 因子交互作用までのすべての別名関係:

$$\begin{aligned} I &= ABDE = ABFG = ACDF = ACEG = BCDG = BCEF = DEFG \\ A &= BDE = BFG = CDF = CEG, & B &= ADE = AFG = CDG = CEF \\ C &= ADF = AEG = BDG = BEF, & D &= ABE = ACF = BCG = EFG \\ E &= ABD = ACG = DFG, & F &= ABG = ACD = BCF = DEG \\ G &= ABF = ACE = BCD = DEF \\ AB &= DE = FG = ABDG = ACDG = ACEF = BCDF = BCEG \\ AC &= DF = EG = ABEF = BCDE = BCFG \\ AD &= BE = CF = ABCG = AEFG = BDFG = CDEG \\ BC &= DG = EF = ABDF = ABEG = ACDE = ACFG \\ BD &= AE = CG = ABCF = ADFG = BEFG = CDEF \\ CD &= AF = BG = ABCE = ADEG = BDEF = CEFG \\ AG &= BF = CE = ABCD = ADEF = BDEG = CDFG \\ ABC &= ADG = AEF = BDF = BEG = CDE = CFG \end{aligned}$$

110

別名関係を考慮して, すべての母数が推定可能になるようにモデルを定めればよい.

- 例えば, 7 つの因子の主効果と, ふたつの交互作用, AC, BD からなるモデル

$$AC/BD/E/F/G$$

は, AC と BD が別名関係にないから, すべての母数は同時に推定可能.

111

この場合の共変量行列 M は,

$$M = \begin{pmatrix} & AC & BD \\ 1 & 1 & 1 \\ 1 & 1 & -1 \\ \vdots & D & \vdots \\ 1 & 1 & -1 \\ 1 & 1 & 1 \end{pmatrix}.$$

112

- 一方で, モデル

$$AB/DE/C/F/G$$

は, AB と DE が別名関係にあるから推定可能でない.

$$AB = DE = FG = ABDG = \dots$$

- このような, 別名関係の判定は, 多項式環イデアルの文脈では「イデアル所属問題」と等価になることが示される. この点については, 本講義では省略. 演習問題を参照.

113

- 飽和モデルの母数は $16(=k)$ 個. このとき共変量行列 M は次数 16 のアダマール行列になる.

- モデルの解釈は一通りでない. 例えば,

$$ABC/AD/BD/CD/AG/EF \text{ と } ABCD$$

はいずれも母数が推定可能な階層モデルで, 飽和モデルの解釈となる.

114

- Hamada and Nelder (1997) は、この噴流式はんだ付け装置の不良品数データに対して、

AC/BD/E/F/G

というモデルの当てはまりを考察している。(演習問題 7)

```
> data7 <- read.table("data7.txt", header=T)
> data7
  A B C D E F G x
1  1 1 1 1 1 1 1 69
2  1 1 1 -1 -1 -1 -1 31
3  1 1 -1 1 1 -1 -1 55
4  1 1 -1 -1 -1 1 1 149
5  1 -1 1 1 -1 1 -1 46
6  1 -1 1 -1 1 -1 1 43
7  1 -1 -1 1 -1 -1 1 118
8  1 -1 -1 -1 1 1 -1 30
9  -1 1 1 1 -1 -1 1 43
10 -1 1 1 -1 1 1 -1 45
11 -1 1 -1 1 -1 1 -1 71
12 -1 1 -1 -1 1 -1 1 380
13 -1 -1 1 1 1 -1 -1 37
14 -1 -1 1 -1 -1 1 1 36
15 -1 -1 -1 1 1 1 1 212
16 -1 -1 -1 -1 -1 -1 -1 52
>
```

```
> data7.glm <- glm(x~A+B+C+D+E+F+G+A*C+B*D,data7,family="poisson")
> summary(data7.glm)
```

Call:
glm(formula = x ~ A + B + C + D + E + F + G + A * C + B * D,
family = "poisson", data = data7)

Deviance Residuals:
Min 1Q Median 3Q Max
-2.52419 -0.47818 0.04629 0.36474 2.62292

Coefficients:
Estimate Std. Error z value Pr(>|z|)
(Intercept) 4.17140 0.03393 122.946 < 2e-16 ***
A -0.10842 0.03250 -3.337 0.000848 ***
B 0.19039 0.03015 6.315 2.70e-10 ***
C -0.42315 0.03352 -12.625 < 2e-16 ***
D 0.01264 0.03086 0.409 0.682193
E 0.10302 0.03206 3.213 0.001312 **
F -0.04236 0.02784 -1.522 0.128070

```
G 0.38144 0.03352 11.380 < 2e-16 ***
A:C 0.18109 0.03318 5.457 4.83e-08 ***
B:D -0.29896 0.03380 -8.845 < 2e-16 ***
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

Null deviance: 1021.255 on 15 degrees of freedom
Residual deviance: 19.093 on 6 degrees of freedom
AIC: 135.26

Number of Fisher Scoring iterations: 4

>
>
>
>
>
>
```

```
> fitted(data7.glm)
  1 2 3 4 5 6
64.52677 47.25345 53.14603 151.07960 30.42595 46.79383
  7 8 9 10 11 12
115.24100 32.53337 49.42430 46.13193 70.90290 360.53502
 13 14 15 16
35.18867 30.25510 232.14438 51.41770
> 1-pchisq(19.093,6)
[1] 0.00400942
> qchisq(0.95,6)
[1] 12.59159
>
```

マルコフ基底と実験計画, 問題集その 1, 2009.09.15

1. 2×3 分割表で行和、列和が固定されているものを考える。つまり

$$A = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix},$$

を考える。

- (a) Markov 基底をトーリックイデアル I_A のグレブナ基底を計算することによりもとめよ。
- (b) 4ti2 を使って Markov 基底を計算せよ。
- (c) $Ax = \beta$, $x \in \mathbb{N}_0^6$ を満たす x を $\beta = (2, 3; 1, 2, 2)$ についてすべて求めよ。
- (d) ファイバー $\mathcal{F}_{(2,3;1,2,2)}$ の連結グラフを書け。

参考文献: [5, chap.3], [3, chap.5].

2. $3 \times 3 \times 3$ 分割表で全ての 2 次元周辺頻度 $\{x_{ij+}\}, \{x_{i+k}\}, \{x_{+jk}\}$ が固定されているものを考える。

- (a) この分割表に対応する行列 A を求めよ。
データ: 3x3x3cont.mat
- (b) 4ti2 を使って Markov 基底を計算せよ。
- (c) $x_{ij+} = 1, x_{i+k} = 1, x_{+jk} = 1$ を満たす分割表を全て列挙せよ。
- (d) (研究課題) 与えられた Markov 基底と分割表の 1 つからファイバーのすべての元を列挙するプログラムをつくれ。またそれから連結グラフを表示するプログラムを書け。
プログラム: enumerate_fiber, connect2 を用いよ。

3. 離散確率変数 x_1, x_2, x_3 、各変数は値として +1 もしくは -1 をとるとする。 (x_1, x_2, x_3) の発生確率を次のようにとる。

$$P(x_1, x_2, x_3) = \frac{\exp(0.2(x_1x_2 + x_2x_3 + x_1x_3))}{Z}$$

ここで Z は正規化定数とよばれるもので

$$Z = \sum_{(x_1, x_2, x_3) \in \{+1, -1\}^3} \exp(0.2(x_1x_2 + x_2x_3 + x_1x_3))$$

である。この分布に従うような (x_1, x_2, x_3) の列をマルコフ連鎖モンテカルロ法を用いてサンプリングし、 $\{+1, -1\}^3$ の 8 点の出現割合がそれぞれ $P(x_1, x_2, x_3)$ に近くなることを確かめよ。

この場合のマルコフ連鎖モンテカルロ法のアルゴリズムは次の通り。

0. $\mathbf{x} = (x_1, x_2, x_3)$ に適当な初期状態を設定。
例えば、 $\mathbf{x} \leftarrow (-1, -1, -1)$
1. x_1, x_2, x_3 のいずれかをランダムに選択。
選んだ変数を x_i とすれば、 x_i の値を $-x_i$ に置き換えたものを \mathbf{x}' とする。
2. $r \leftarrow \frac{P(\mathbf{x}')}{P(\mathbf{x})}$
3. $0 \leq R < 1$ の一様乱数 R をとる
4. $r > R$ であれば、 $\mathbf{x}_{\text{next}} \leftarrow \mathbf{x}'$
それ以外の場合、 $\mathbf{x}_{\text{next}} \leftarrow \mathbf{x}$
5. \mathbf{x}_{next} がサンプルとして得られる
6. $\mathbf{x} \leftarrow \mathbf{x}_{\text{next}}$
step 1 に戻る

適当なプログラミング言語で上のアルゴリズムを実装するか、もしくは、プログラム: metropolis を用いよ。

参考文献: [4, p.8]

4. 2×3 分割表 周辺頻度 $(5, 15; 5, 5, 10)$ のファイバー \mathcal{F} 、すなわち

$$\mathcal{F} = \{(x_{ij}) \mid x_{ij} \in \mathbf{N}, x_{1+} = 5, x_{2+} = 15, x_{+1} = 5, x_{+2} = 5, x_{+3} = 10\}$$

から、分布 $\frac{1}{Z} \frac{1}{x_{11}!x_{12}!x_{13}!x_{21}!x_{22}!x_{23}!}$ に従って元をマルコフ連鎖モンテカルロ法を用いてサンプリングし、 $2x_{11} + x_{12}$ の期待値を推定せよ。 $(Z$ は正規化定数で $\sum_{x \in \mathcal{F}} \frac{1}{x_{11}!x_{12}!x_{13}!x_{21}!x_{22}!x_{23}!}$)

プログラム: 2x3mcmc を用いよ。

5. 2×2 分割表でデータ

疾病/喫煙	あり	なし	
あり	3	1	4
なし	2	4	6
	5	5	10

が与えられているとする。

- (a) ファイバー $\mathcal{F}_{(4,6;5,5)}$ の要素を全列挙し、
 H_0 : 喫煙と疾病は無関係

が真と仮定した時のそれぞれの出現確率 $p(x) = \frac{1}{Z} \frac{1}{x_{11}!x_{12}!x_{21}!x_{22}!}$ を求めよ。ただし、 Z は正規化定数である。

- (b) 「 $X_{11} > c$ ならば H_0 を棄却」という検定方式を考える。与えられたデータの p 値を計算せよ。

6. 5×5 2 元分割表でデータ

幾何/代数	5	4	3	2	1	計
5	2	1	1	0	0	4
4	8	3	3	0	0	14
3	0	2	1	1	1	5
2	0	0	0	1	1	2
1	0	0	0	0	1	1
計	10	6	5	2	3	26

が与えられているとする。([5], [6] より引用)

帰無仮説 (H_0): 代数と幾何の成績は独立。

とする。

- (a) H_0 の下での当てはめ値を計算せよ。
 (b) カイ 2 乗適合度統計量 $\chi^2(x)$ を計算せよ。
 (c) 上の結果から漸近的 p 値を求めよ。
 (d) マルコフ連鎖モンテカルロ法を用いてサンプリングを行い、 p 値を近似的に求めよ。

プログラム : 5x5mcmc を用いよ。

参考文献 : [6], [5, chap.3]

7. 2^{7-3} 一部実施計画 (要因 A, B, C, D, E, F, G , 別名関係 $ABDE = ACDF = BCDG = I$) のデータとして次のようなものが与えられているとする。
 ([1] より引用)

run	A	B	C	D	E	F	G	y
1	1	1	1	1	1	1	1	69
2	1	1	1	-1	-1	-1	-1	31
3	1	1	-1	1	1	-1	-1	55
4	1	1	-1	-1	-1	1	1	149
5	1	-1	1	1	-1	1	-1	46
6	1	-1	1	-1	1	-1	1	43
7	1	-1	-1	1	-1	-1	1	118
8	1	-1	-1	-1	1	1	-1	30
9	-1	1	1	1	-1	-1	1	43
10	-1	1	1	-1	1	1	-1	45
11	-1	1	-1	1	-1	1	-1	71
12	-1	1	-1	-1	1	-1	1	380
13	-1	-1	1	1	1	-1	-1	37
14	-1	-1	1	-1	-1	1	1	36
15	-1	-1	-1	1	1	1	1	212
16	-1	-1	-1	-1	-1	-1	-1	52

階層モデル $H_0 : AC/BD/E/F/G$ のデータへの当てはまりを、以下の手順で検証せよ。

(a) モデル H_0 に対する共変量行列 M を求めよ。

データ : covariate_mat1.mat, 2_7-3.dat

(b) モデル H_0 の下での当てはめ値 $\{m_1, \dots, m_{16}\}$ を求めよ。

(c) 与えられたデータ $\{y_1, \dots, y_{16}\}$ について、尤度比検定統計量

$$G(y) = 2 \sum_{i=1}^{16} y_i \log \frac{y_i}{m_i}$$

を計算せよ。また、この値を、自由度 6 のカイ 2 乗分布の上側 5% 点と比較せよ。

(d) M' の Markov 基底を求めよ。

(e) データに対するモデル H_0 の当てはまりを、尤度比検定統計量にもとづき検証する。マルコフ連鎖モンテカルロ法により、このデータの p 値を計算せよ。

プログラム : cov1_mcmc を用いよ。

さらに、別の階層モデル $AB/AC/BD/E/F/G$ についても、上と同様の検定を行え。

データ : covariate_mat2.mat

参考文献 : [1]

8. (研究課題, 社会階層の移動のマルコフモデル) L (level), G (generation) を自然数とする. i を $0 \leq i < L^G$ をみたす整数とすると、この数の G 桁 L 進数への展開を $i_{G-1} \dots i_1 i_0$ とする. 次のような式で表される affine toric variety を考える.

$$x_{i_{G-1} \dots i_1 i_0} = p_{i_0} p_{i_0 i_1} p_{i_1 i_2} \dots p_{i_{G-2} i_{G-1}}$$

ここで x 変数は L^G 個, 添字 1 つの p 変数が L 個, 添字 2 つの p 変数が L^2 個. x 変数の 2 項関係式をきめる行列 A を求め (参考 : sstatus2.rr), この A に対して 4ti2 で Markov 基底を計算せよ. どの位のサイズの L, G まで計算ができるか?

参考文献

- [1] S.Aoki, A.Takemura, Markov chain Monte Carlo tests for designed experiments, arXiv:math.ST0611463v1

- [2] S.Aoki, A.Takemura, Markov basis for design of experiments with three-level factors, arXiv:0709.4323v2
- [3] B.Sturmfels, Gröbner Bases and Convex Polytopes, 1995, American Mathematical Society, University lecture series 8.
- [4] 伊庭 幸人, 種村 正美, 大森 裕浩, 和合 肇, 佐藤 整尚, 高橋 昭彦, “統計科学のフロンティア 12, 計算統計 II マルコフ連鎖モンテカルロ法とその周辺”, 岩波書店, 2005
- [5] 日比孝之編, グレブナー基底の現在, 数学書房, 2006
- [6] 青木敏, マルコフ連鎖、モンテカルロ法による分割表の解析, 応用統計学会第 27 回シンポジウム発表資料, 2005

1. 2×3 分割表で行和、列和が固定されているものを考える。つまり

$$A = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

を考える。

(a) Markov 基底をトーリックイデアル I_A のグレブナ基底を計算することによりもとめよ。

$$I_A = \langle x_2x_6 - x_3x_5, x_1x_6 - x_3x_4, x_1x_5 - x_2x_4 \rangle$$

トーリックイデアルを計算する Asir のプログラム toric.rr を用いれば、

```
Asir
[1206] load("toric.rr");
[1219] toric_ideal([[1,1,1,0,0,0],[0,0,0,1,1,1],
[1,0,0,1,0,0],[0,1,0,0,1,0],[0,0,1,0,0,1]]);
ideal :
[x5-t1*t4,x4-t1*t3,x3-t1*t2,x2-t0*t4,x1-t0*t3,
x0-t0*t2,t0*t1*t2*t3*t4*t5-1]
gb :
[x1*x5-x2*x4,x0*x5-x2*x3,x0*x4-x1*x3,t3*x5-t4*x4,
t3*x2-t4*x1,t2*x5-t4*x3,t2*x4-t3*x3,t2*x2-t4*x0,
t2*x1-t3*x0,t5*x0*x4*x5-t1,-t5*x0*x1*x5+t0,t4*t5*x0*x4-1,
t4*t5*x0*x1*x5-x2,-t3*t4*t5*x0*x3+t2]
[x1*x5-x2*x4,x0*x5-x2*x3,x0*x4-x1*x3]
[1220]
```

(b) 4ti2 を使って Markov 基底を計算せよ。

上でグレブナ基底を得られているので、それが Markov 基底でもあるが、あえて 4ti2 を使い Markov 基底を計算する。行列 A を表す次のような入力ファイルを用意。

```
cont2x3.mat
5 6
1 1 1 0 0 0
0 0 0 1 1 1
1 0 0 1 0 0
0 1 0 0 1 0
0 0 1 0 0 1
```

4ti2 の Markov 基底を求めるコマンド markov を実行

markov の実行

```
$ markov cont2x3.mat
... 略
File cont2x3.mat.cost not found. Using default cost.
COMPUTING GENERATING SET ... PROJECT-AND-LIFT ...
Phase 1: Saturating 3 variable(s).
  Saturating (3) ... (GM) Size:      2, Time: -0.00 / 0.01 secs. Done.
  Saturated already on 2 variable(s).
Done. Size:      2, Time: -0.00 / 0.01 secs
Phase 2: Lifting 3 variable(s).
  Lifting (1) ... (N) Size:      3, Time: -0.00 / 0.01 secs. Done.
Done. Size:      3, Time: -0.00 / 0.01 secs
COMPUTING MINIMAL GENERATING SET (FAST) ...
Size:      3, Time: -0.00 / 0.01 secs. Done.
Total Time: 0.02 secs.
```

Markov 基底が次のファイルに出力される。

cont2x3.mat.mar

```
3 6
0 -1 1 0 1 -1
1 -1 0 -1 1 0
1 0 -1 -1 0 1
```

(c) $Ax = \beta$, $x \in \mathbf{N}_0^6$ を満たす x を $\beta = (2, 3; 1, 2, 2)'$ についてすべて求めよ。

$Ax = \beta$ を満たす x を 1 つとってくる。たとえば、

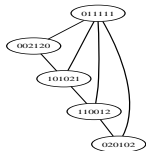
$$x = (0, 1, 1, 1, 1, 1)'$$

これに対して、上で求めた Markov 基底を足し引きして、各成分が非負であるものを列挙していけばよい。
このような x は次の 5 つ。

$$(0, 1, 1, 1, 1, 1)', (0, 2, 0, 1, 0, 2)', (1, 1, 0, 0, 1, 2)', (1, 0, 1, 0, 2, 1)', (0, 0, 2, 1, 2, 0)'$$

(d) ファイバー $\mathcal{F}_{(2,3;1,2,2)}$ の連結グラフを書け。

連結グラフは Markov 基底の加減で移り合うもの同士を結べばよい。



参考文献: [5, chap.3], [3, chap.5].

プログラム: cont2x3.mat, ex-stat-ans.tex

2. $3 \times 3 \times 3$ 分割表で全ての 2 次元周辺頻度 $\{x_{ij+}\}, \{x_{i+k}\}, \{x_{+jk}\}$ が固定されているものを考える。

(a) この分割表に対応する行列 A を求めよ。

行列の各列が

$$(x_{111}, x_{112}, x_{113}, x_{121}, x_{122}, x_{123}, x_{131}, x_{132}, x_{133}, x_{211}, x_{212}, x_{213}, x_{221}, x_{222}, x_{223}, x_{231}, x_{232}, x_{233}, \\ x_{311}, x_{312}, x_{313}, x_{321}, x_{322}, x_{323}, x_{331}, x_{332}, x_{333})'$$

に対応しているものとし、各行は

$$(x_{11+}, x_{12+}, x_{13+}, x_{21+}, x_{22+}, x_{23+}, x_{31+}, x_{32+}, x_{33+}, x_{+11}, x_{+12}, x_{+13}, x_{+21}, x_{+22}, x_{+23}, x_{+31}, x_{+32}, x_{+33}, \\ x_{+11}, x_{+12}, x_{+13}, x_{+21}, x_{+22}, x_{+23}, x_{+31}, x_{+32}, x_{+33})'$$

プログラム enumerate_fiber の実行

```
$ enumerate_fiber start_v4.txt move4.txt
n_move : 1, msize : 6
start_v :
 0 1 1 1 1 1
n_move : 3, msize : 6
move : 3 6
 0 -1 1 0 1 -1
 1 -1 0 -1 1 0
 1 0 -1 -1 0 1
0 0 2 1 2 0
1 0 1 0 2 1
1 1 0 0 1 2
0 2 0 1 0 2
depth : 0, count : 5, queue : 4
depth : 1, count : 5, queue : 3
depth : 2, count : 5, queue : 2
depth : 3, count : 5, queue : 1
depth : 4, count : 5, queue : 0
count : 5
fiber :
0 1 1 1 1 1
0 0 2 1 2 0
1 0 1 0 2 1
1 1 0 0 1 2
0 2 0 1 0 2
```

3. 離散確率変数 x_1, x_2, x_3 、各変数は値として $+1$ もしくは -1 をとるとする。 (x_1, x_2, x_3) の発生確率を次のようにとる。

$$P(x_1, x_2, x_3) = \frac{\exp(0.2(x_1x_2 + x_2x_3 + x_1x_3))}{Z}$$

ここで Z は正規化定数とよばれるもので

$$Z = \sum_{(x_1, x_2, x_3) \in \{+1, -1\}^3} \exp(0.2(x_1x_2 + x_2x_3 + x_1x_3))$$

である。この分布に従うような (x_1, x_2, x_3) の列をマルコフ連鎖モンテカルロ法を用いてサンプリングし、 $\{+1, -1\}^3$ の 8 点の出現割合がそれぞれ $P(x_1, x_2, x_3)$ に近くなることを確かめよ。

プログラム: metropolis を用いよ。

参考文献: [4, p.8]

この場合のマルコフ連鎖モンテカルロ法のアルゴリズムは次の通り。

0. $\mathbf{x} = (x_1, x_2, x_3)$ に適当な初期状態を設定。
例えば、 $\mathbf{x} \leftarrow (-1, -1, -1)$
1. x_1, x_2, x_3 のいずれかをランダムに選択。
選んだ変数を x_i とすれば、 x_i の値を $-x_i$ に置き換えたものを \mathbf{x}' とする。
2. $r \leftarrow \frac{P(\mathbf{x}')}{P(\mathbf{x})}$
3. $0 \leq R < 1$ の一様乱数 R をとる
4. $r > R$ であれば、 $\mathbf{x}_{\text{next}} \leftarrow \mathbf{x}'$
それ以外の場合、 $\mathbf{x}_{\text{next}} \leftarrow \mathbf{x}$
5. \mathbf{x}_{next} がサンプルとして得られる
6. $\mathbf{x} \leftarrow \mathbf{x}_{\text{next}}$
step 1 に戻る

プログラム metropolis の実行

```

$ ./metropolis
-1 -1 -1
-1 1 -1
 1 1 -1
...
 1 1 1
 1 1 1
 1 1 -1
 1 -1 -1
-1 -1 -1
index 0 : 2198
index 1 : 1001
index 2 : 890
index 3 : 926
index 4 : 995
index 5 : 991
index 6 : 932
index 7 : 2067
total : 10000
z : 8.556622
index 0 : 0.219800 0.212948
index 1 : 0.100100 0.095684
index 2 : 0.089000 0.095684
index 3 : 0.092600 0.095684
index 4 : 0.099500 0.095684
index 5 : 0.099100 0.095684
index 6 : 0.093200 0.095684
index 7 : 0.206700 0.212948

```

4. 2×3 分割表 周辺頻度 $(5, 15; 5, 5, 10)$ のファイバー \mathcal{F} 、すなわち

$$\mathcal{F} = \{(x_{ij}) \mid x_{ij} \in \mathbf{N}, x_{1+} = 5, x_{2+} = 15, x_{+1} = 5, x_{+2} = 5, x_{+3} = 10\}$$

から、分布 $\frac{1}{Z} \frac{1}{x_{11}!x_{12}!x_{13}!x_{21}!x_{22}!x_{23}!}$ に従って元をマルコフ連鎖モンテカルロ法を用いてサンプリングし、 $2x_{11} + x_{12}$ の期待値を推定せよ。(Z は正規化定数で $\sum_{x \in \mathcal{F}} \frac{1}{x_{11}!x_{12}!x_{13}!x_{21}!x_{22}!x_{23}!}$)

2×3 分割表のマルコフ基底は

$$\mathcal{B} = \left\{ \begin{array}{|c|c|c|} \hline 1 & -1 & \\ \hline -1 & 1 & \\ \hline & & \\ \hline \end{array}, \begin{array}{|c|c|c|} \hline 1 & & -1 \\ \hline -1 & & 1 \\ \hline & & \\ \hline \end{array}, \begin{array}{|c|c|c|} \hline & 1 & -1 \\ \hline & -1 & 1 \\ \hline & & \\ \hline \end{array} \right\}$$

与えられた確率分布 $f(\mathbf{x}) = \frac{1}{Z} \prod_{i,j} \frac{1}{x_{ij}!}$ に従って、ファイバー \mathcal{F} から元をサンプリングする、マルコフ連鎖モンテカルロ法は以下の手順である。

0. ファイバー \mathcal{F} の元 \mathbf{x} を 1 つ取り出す (なんでもよい)
 $e \leftarrow 0$
1. $\mathcal{B} \cup (-\mathcal{B})$ からランダムに元 \mathbf{z} を選ぶ
2. $\mathbf{x} + \mathbf{z}$ の各要素が全て非負ならば、 $r \leftarrow \frac{f(\mathbf{x} + \mathbf{z})}{f(\mathbf{x})}$
 それ以外の場合、 $r \leftarrow 0$
3. $0 \leq R < 1$ の一様乱数 R をとる
4. $r > R$ であれば、 $\mathbf{x}_{\text{next}} \leftarrow \mathbf{x} + \mathbf{z}$
 それ以外の場合、 $\mathbf{x}_{\text{next}} \leftarrow \mathbf{x}$
5. \mathbf{x}_{next} がサンプルとして得られる
 $e \leftarrow e + (\mathbf{x}_{\text{next}} \text{ の } 2x_{11} + x_{12})$

6. $x \leftarrow x_{\text{next}}$
 step 1 に戻る

7. $e/(\text{サンプル数})$ が期待値の推定値

これを実行するプログラムが 2x3mcmc である。
 2x3mcmc (乱数 seed) (サンプル数) のように実行する。
 正確な期待値は $15/4 = 3.75$ である。

2x3mcmc の実行結果

```
./2x3mcmc 0 100000
2 0 3 3 5 7
2 0 3 3 5 7
2 0 3 3 5 7
2 0 3 3 5 7
1 1 3 4 4 7
0 1 4 5 4 6
0 2 3 5 3 7
1 2 2 4 3 8
2 1 2 3 4 8
1 2 2 4 3 8
...
2 1 2 3 4 8
3 0 2 2 5 8
2 0 3 3 5 7
2 1 2 3 4 8
ave : 3.746200
```

5. 2×2 分割表でデータ

疾病/喫煙	あり	なし	
あり	3	1	4
なし	2	4	6
	5	5	10

が与えられているとする。

(a) ファイバー $\mathcal{F}_{(4,6;5,5)}$ の要素を全列挙し、

H_0 : 喫煙と疾病は無関係

が真と仮定した時のそれぞれの出現確率 $p(x) = \frac{1}{Z} \frac{1}{x_{11}!x_{12}!x_{21}!x_{22}!}$ を求めよ。ただし、 Z は正規化定数である。

ファイバー $\mathcal{F}_{(4,6;5,5)}$ の元は次の 5 つである。

<table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>4</td><td>0</td></tr><tr><td>1</td><td>5</td></tr></table>	4	0	1	5	,	<table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>3</td><td>1</td></tr><tr><td>2</td><td>4</td></tr></table>	3	1	2	4	,	<table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>2</td><td>2</td></tr><tr><td>3</td><td>3</td></tr></table>	2	2	3	3	,	<table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>1</td><td>3</td></tr><tr><td>4</td><td>2</td></tr></table>	1	3	4	2	,	<table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>0</td><td>4</td></tr><tr><td>5</td><td>1</td></tr></table>	0	4	5	1
4	0																											
1	5																											
3	1																											
2	4																											
2	2																											
3	3																											
1	3																											
4	2																											
0	4																											
5	1																											

それぞれの確率は

$$\frac{1}{42}, \frac{10}{42}, \frac{20}{42}, \frac{10}{42}, \frac{1}{42}$$

となる。

(b) 「 $X_{11} > c$ ならば H_0 を棄却」という検定方式を考える。与えられたデータの p 値を計算せよ。

観測値の x_{11} は 3 より $c = 3$ として、

$$p \text{ 値} = \Pr(X_{11} \geq 3) = \frac{10}{42} + \frac{1}{42} = \frac{11}{42}$$

$p \text{ 値} > 0.05$ より、 H_0 は棄却されない。

6. 5×5 2元分割表でデータ

幾何/代数	5	4	3	2	1	計
5	2	1	1	0	0	4
4	8	3	3	0	0	14
3	0	2	1	1	1	5
2	0	0	0	1	1	2
1	0	0	0	0	1	1
計	10	6	5	2	3	26

が与えられているとする。([5], [6] より引用)

帰無仮説 (H_0) : 代数と幾何の成績は独立。

とする。

(a) H_0 の下での当てはめ値を計算せよ。(R or 手計算)

H_0 の下での当てはめ値 (理論度数) の計算。セル (i, j) の当てはめ値は

$$\hat{x}_{ij} = n \frac{x_{i+} x_{+j}}{n^2}$$

で計算できる。

(b) カイ 2 乗適合度統計量 $\chi^2(x)$ を計算せよ。(R or 表)

(c) 上の結果から漸近的 p 値を求めよ。

カイ 2 乗検定統計量の式

$$\chi^2(\mathbf{x}) = \sum_{i,j} \frac{(x_{ij} - \hat{x}_{ij})^2}{\hat{x}_{ij}}$$

にしたがって計算を行えば、 $\chi^2(\mathbf{x}) = 25.3376$ という結果が得られる。

R でこれを実行するには次のように行う。

```
R でのカイ 2 乗検定の実行
m <- matrix(c(2,8,0,0,1,3,2,0,0,1,3,1,0,0,0,0,
              1,1,0,0,0,1,1,1),nrow=5,ncol=5)
chisq.test(m)
```

この実行結果によれば、漸近的 p 値は 0.06409

(d) マルコフ連鎖モンテカルロ法を用いてサンプリングを行い、 p 値を近似的に求めよ。

Q5 と同じようにマルコフ連鎖モンテカルロ法を行う。今回は p 値を計算しなければならないので、得られたサンプル \mathbf{x} について、 $\chi^2(\mathbf{x}) \geq 25.3376$ を判定して真であれば、カウント c を増やす。サンプリング終了時に、 $\frac{c}{(\text{サンプル数})}$ が p 値の近似になる。アルゴリズムを改めて書けば次のようになる。

0. $\mathbf{x} \leftarrow$ (与えられているデータ)
 カウント $c \leftarrow 0$
 基準のカイ 2 乗統計量 $\chi^2 \leftarrow \chi^2(\mathbf{x})$
1. $B \cup (-B)$ からランダムに元 \mathbf{z} を選ぶ
2. $\mathbf{x} + \mathbf{z}$ の各要素が全て非負ならば、 $r \leftarrow \frac{f(\mathbf{x}+\mathbf{z})}{f(\mathbf{x})}$
 それ以外の場合、 $r \leftarrow 0$
3. $0 \leq R < 1$ の一様乱数 R をとる
4. $r > R$ であれば、 $\mathbf{x}_{\text{next}} \leftarrow \mathbf{x} + \mathbf{z}$
 それ以外の場合、 $\mathbf{x}_{\text{next}} \leftarrow \mathbf{x}$
5. \mathbf{x}_{next} がサンプルとして得られる
 $\chi^2(\mathbf{x}_{\text{next}}) \geq \chi^2$ ならば $c \leftarrow c + 1$

6. $\mathbf{x} \leftarrow \mathbf{x}_{\text{next}}$
 step 1 に戻る

これを実行するプログラムが 5x5mcmc である。
 5x5mcmc (乱数 seed) (サンプル数) のように実行する。

```

5x5mcmc の実行結果
./5x5mcmc 0 100000
chisq : 25.337619
0 2 1 1 0 7 4 1 1 1 1 0 3 0 1 1 0 0 0
1 1 0 0 0 0
0 2 1 1 0 7 4 1 1 1 1 0 3 0 1 1 0 0 0
1 1 0 0 0 0
0 2 1 1 0 7 4 1 1 1 1 0 3 0 1 1 0 0 0
1 1 0 0 0 0
....
2 2 0 0 0 7 2 2 0 3 1 0 2 2 0 0 1 1 0
0 0 1 0 0 0
2 2 0 0 0 7 2 2 0 3 1 0 2 2 0 0 1 1 0
0 0 1 0 0 0
2 2 0 0 0 7 2 2 0 3 1 0 2 2 0 0 1 1 0
0 0 1 0 0 0
tchi_count : 6302
approximate p-value : 0.063020
  
```

p 値 > 0.05 より H_0 は棄却されない。

7. 2^{7-3} 一部実施計画 (要因 A, B, C, D, E, F, G , 別名関係 $ABDE = ACDF = BCDG = I$) のデータとして次のようなものが与えられているとする。([1] より引用)

run	A	B	C	D	E	F	G	y
1	1	1	1	1	1	1	1	69
2	1	1	1	-1	-1	-1	-1	31
3	1	1	-1	1	1	-1	-1	55
4	1	1	-1	-1	-1	1	1	149
5	1	-1	1	1	-1	1	-1	46
6	1	-1	1	-1	1	-1	1	43
7	1	-1	-1	1	-1	-1	1	118
8	1	-1	-1	-1	1	1	-1	30
9	-1	1	1	1	-1	-1	1	43
10	-1	1	1	-1	1	1	-1	45
11	-1	1	-1	1	-1	1	-1	71
12	-1	1	-1	-1	1	-1	1	380
13	-1	-1	1	1	1	-1	-1	37
14	-1	-1	1	-1	-1	1	1	36
15	-1	-1	-1	1	1	1	1	212
16	-1	-1	-1	-1	-1	-1	-1	52

階層モデル $H_0: AC/BD/E/F/G$ のデータへの当てはまりを、以下の手順で検証せよ。

- (a) モデル H_0 に対する共変量行列 M を求めよ。

階層モデル $AC/BD/E/F/G$ だから、主効果 A, B, C, D, E, F, G と 2 因子交互作用 AC, BD であるから、共変量行列は

$$M = (1, d_A, d_B, d_C, d_D, d_E, d_F, d_G, d_{AC}, d_{BD})$$

で具体的には、

```

covariate_mat1.mat
10 16
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 -1 -1 -1 -1 -1 -1 -1 -1
1 1 1 1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1 -1
1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1
1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1
1 -1 1 -1 -1 1 -1 1 -1 1 -1 1 1 -1 1 -1
1 -1 -1 1 1 -1 -1 1 -1 1 1 -1 -1 1 1 -1
1 -1 -1 1 -1 1 1 -1 1 -1 -1 1 -1 1 1 -1
1 1 -1 -1 1 1 -1 -1 -1 -1 1 1 -1 -1 1 1
1 -1 1 -1 -1 1 -1 1 1 -1 -1 -1 1 -1 1 1

```

のようになる。(ただし、これは転置行列の方 M')

(b) モデル H_0 の下での当てはめ値 $\{m_1, \dots, m_{16}\}$ を求めよ。

R を用いて当てはめ値を計算するには次のようにする。まず次のようなファイルを用意しておく。

```

2_7-3.dat
A, B, C, D, E, F, G, x
1, 1, 1, 1, 1, 1, 1, 69
1, 1, 1, -1, -1, -1, -1, 31
1, 1, -1, 1, 1, -1, -1, 55
1, 1, -1, -1, -1, 1, 1, 149
1, -1, 1, 1, -1, 1, -1, 46
1, -1, 1, -1, 1, -1, 1, 43
1, -1, -1, 1, -1, -1, 1, 118
1, -1, -1, -1, 1, 1, -1, 30
-1, 1, 1, 1, -1, -1, 1, 43
-1, 1, 1, -1, 1, 1, -1, 45
-1, 1, -1, 1, -1, 1, -1, 71
-1, 1, -1, -1, 1, -1, 1, 380
-1, -1, 1, 1, 1, -1, -1, 37
-1, -1, 1, -1, -1, 1, 1, 36
-1, -1, -1, 1, 1, 1, 1, 212
-1, -1, -1, -1, -1, -1, -1, 52

```

R で次のような計算を行えばいい。

```

R での当てはめ値の計算
> dat<-read.table(file="2_7-3.dat", header=T, sep=",")
> dat.glm<-glm(x~A+B+C+D+E+F+G+A*C+B*D, dat, family=poisson)
> fitted(dat.glm)
      1      2      3      4      5      6      7      8
64.52677 47.25345 53.14603 151.07960 30.42595 46.79383 115.24100 32.53337
      9     10     11     12     13     14     15     16
49.42430 46.13193 70.90290 360.53502 35.18867 30.25510 232.14438 51.41770

```

(c) 与えられたデータ $\{y_1, \dots, y_{16}\}$ について、尤度比検定統計量

$$G(y) = 2 \sum_{i=1}^{16} y_i \log \frac{y_i}{m_i}$$

を計算せよ。また、この値を、自由度 6 のカイ 2 乗分布の上側 5% 点と比較せよ。

$$G(\mathbf{x}) = 19.09271$$

$$\chi_{0.05}^2(6) = 18.5476$$

(d) M' の Markov 基底を求めよ。

4ti2 を用いて Markov 基底を計算すると 23 個の Markov 基底が得られる。

(e) データに対するモデル H_0 の当てはまりを、尤度比検定統計量にもとづき検証する。マルコフ連鎖モンテカルロ法により、このデータの p 値を計算せよ。

Q5 と同じようにマルコフ連鎖モンテカルロ法を行う。今回は p 値を計算しなければならないので、得られたサンプル \mathbf{x} について、 $G(\mathbf{x}) \geq 19.09271$ を判定して真であれば、カウント c を増やす。サンプリング終了時に、 $\frac{c}{(\text{サンプル数})}$ が p 値の近似になる。アルゴリズムを改めて書けば次のようになる。

0. $\mathbf{x} \leftarrow$ (与えられているデータ)
 カウント $c \leftarrow 0$
 基準の尤度比検定統計量 $G \leftarrow G(\mathbf{x})$
1. $B \cup (-B)$ からランダムに元 \mathbf{z} を選ぶ
2. $\mathbf{x} + \mathbf{z}$ の各要素が全て非負ならば、 $r \leftarrow \frac{f(\mathbf{x} + \mathbf{z})}{f(\mathbf{x})}$
 それ以外の場合、 $r \leftarrow 0$
3. $0 \leq R < 1$ の一様乱数 R をとる
4. $r > R$ であれば、 $\mathbf{x}_{\text{next}} \leftarrow \mathbf{x} + \mathbf{z}$
 それ以外の場合、 $\mathbf{x}_{\text{next}} \leftarrow \mathbf{x}$
5. \mathbf{x}_{next} がサンプルとして得られる
 $G(\mathbf{x}_{\text{next}}) \geq G$ ならば $c \leftarrow c + 1$
6. $\mathbf{x} \leftarrow \mathbf{x}_{\text{next}}$
 step 1 に戻る

これを実行するプログラムが cov1.mcmc である。

cov1.mcmc (乱数 seed) (サンプル数) のように実行する。

cov1.mcmc の実行結果

```
./cov1_mcmc 0 100000
likelihood ratio statistic : 19.092715
69 48 45 157 23 49 116 34 51 47 73 353 41 22 233 56
68 48 45 158 23 50 115 34 52 47 73 352 41 21 234 56
68 48 45 158 23 50 115 34 52 47 73 352 41 21 234 56
69 48 44 157 23 49 116 35 52 47 73 353 41 21 233 56
.....
64 48 56 161 25 52 110 25 47 46 71 350 40 28 238 56
65 49 55 160 24 51 111 26 47 46 71 350 40 28 238 56
64 49 55 160 24 52 111 26 47 46 72 350 40 28 238 55
64 49 55 160 24 52 111 26 47 46 72 350 40 28 238 55
tlratio_count : 4025
approximate p-value : 0.004025
```

モデル $AB/AC/BD/E/F/G$ での同様の計算

(a) 共変量行列 M

```
covariate_mat2.mat
11 16
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 -1 -1 -1 -1 -1 -1 -1 -1
1 1 1 1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1 -1
1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1
1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1
1 -1 1 -1 -1 1 -1 1 -1 1 1 -1 1 -1 1 -1
1 -1 -1 1 1 -1 -1 1 -1 1 1 -1 -1 1 1 -1
1 -1 -1 1 -1 1 1 -1 1 -1 -1 1 -1 1 1 -1
1 1 -1 -1 1 1 -1 -1 -1 -1 1 1 -1 -1 1 1
1 -1 1 -1 -1 1 -1 1 1 -1 1 -1 -1 1 -1 1
1 1 1 1 -1 -1 -1 -1 -1 -1 -1 -1 1 1 1 1
```

(b) R で同じ計算を行えば、

```
R での当てはめ値の計算
> dat<-read.table(file="2_7-3.dat", header=T, sep=",")
> dat.glm<-glm(x~A+B+C+D+E+F+G+A*B+A*C+B*D, dat, family=poisson)
> fitted(dat.glm)
      1      2      3      4      5      6      7      8
63.12502 44.15620 49.96722 146.75155 33.08098 48.63780 120.01126 35.26996
      9     10     11     12     13     14     15     16
49.99003 48.73134 74.91772 365.36090 33.03148 29.24715 226.87628 47.84509
>
```

(c)

$$G(y) = 16.8381, \chi_{0.05}^2(5) = 11.0705$$

(d) M' の Markov 基底を 4ti2 を用いて計算すれば 41 個の Markov 基底が得られる。

(e) cov2.mcmc を実行した結果は次の通り。

```
cov2.mcmc の実行結果
./cov2_mcmc 1 100000
likelihood ratio statistic : 16.838141
p : 0.496995 0.135229
9.003644 62 38 49 155 36 53 115 33 45 41 82 371 44 31 218 44
p : 1.240357 0.939720
8.553470 62 39 50 153 36 52 116 33 44 41 82 372 43 33 218 43
p : 2.046756 0.645705
7.119319 62 40 50 152 35 52 117 33 44 42 82 371 42 33 219 43
p : 1.052914 0.914755
7.024061 64 40 48 152 34 51 118 34 44 42 82 371 43 32 218 44
p : 1.267671 0.494924
...
p : 2.105334 0.048496
4.524540 63 46 49 146 37 43 115 42 55 42 71 371 34 30 227 46
p : 0.991831 0.143881
4.548710 64 45 48 147 37 43 115 42 55 42 71 371 35 29 226 47
p : 0.502882 0.824895
4.548710 64 45 48 147 37 43 115 42 55 42 71 371 35 29 226 47
p : 2.006670 0.186261
3.159083 64 45 48 147 36 44 116 41 54 43 72 370 35 29 226 47
tlratio_count : 401
approximate p-value : 0.004010
```

参考文献

- [1] S.Aoki, A.Takemura, Markov chain Monte Carlo tests for designed experiments, arXiv:math.ST0611463v1
- [2] S.Aoki, A.Takemura, Markov basis for design of experiments with three-level factors, arXiv:0709.4323v2
- [3] B.Sturmfels, Gröbner Bases and Convex Polytopes, 1995, American Mathematical Society, University lecture series 8.
- [4] 伊庭 幸人, 種村 正美, 大森 裕浩, 和合 肇, 佐藤 整尚, 高橋 昭彦, “統計科学のフロンティア 12, 計算統計 II マルコフ連鎖モンテカルロ法とその周辺”, 岩波書店, 2005
- [5] 日比孝之編, グレブナー基底の現在, 数学書房, 2006
- [6] 青木敏, マルコフ連鎖、モンテカルロ法による分割表の解析, 応用統計学会第 27 回シンポジウム発表資料, 2005

1 計画 (design) と計画イデアル (design ideal)

k を体とする. ここでは主に $k = \mathbf{Q}$ として考える.

定義 1 k^n の相異なる点の有限集合を計画 (design) とよぶ.

問 1 次の実施計画の計画点を具体的に書け.

1. D_1 : 2^2 -完全実施計画 (水準 $\{-1, 1\}$)
2. D_2 : 2^2 -完全実施計画 (水準 $\{0, 1\}$)
3. D_3 : 3^3 -完全実施計画 (水準 $\{-1, 0, 1\}$) の中で少なくとも 1 つの変数が 0 であるもの.
4. D_4 : 別名関係 $ACD = I$ をもつ 2^{4-1} -一部実施計画 (水準 $\{-1, 1\}$)
5. D_5 : 別名関係 $ABC^2 = BCD = I$ ($a + b + 2c \equiv 0 \pmod{3}, b + c + d \equiv 0 \pmod{3}$) を持つ 3^{4-2} -一部実施計画 (水準 $\{-1, 0, 1\}$)
6. D_6 : 別名関係 $ABDE = ACDF = BCDG = I$ を持つ 2^{7-3} -一部実施計画 (水準 $\{-1, 1\}$)

例 1 (1) $D_1 = \{-1, 1\}^2 = \{(1, 1), (1, -1), (-1, 1), (-1, -1)\}$

参考文献 : [1, chap 3.1]

定義 2 計画 D に対して, D の任意の点を零点にもつ全ての多項式の集合

$$\mathbf{I}(D) = \{f(x) \in \mathbf{Q}[x] := \mathbf{Q}[x_1, \dots, x_n] \mid f(a) = 0 (\forall a \in D)\}$$

を D の計画イデアル (design ideal) という.

1 点 $a = (a_1, \dots, a_n) \in \mathbf{Q}^n$ のみを計画点としてもつ計画の計画イデアルは, 定義より,

$$\langle x_1 - a_1, \dots, x_n - a_n \rangle \subset \mathbf{Q}[x]$$

であるので, m 点 $a_i = (a_{i1}, \dots, a_{in})$ ($1 \leq i \leq m$) からなる計画の計画イデアルは

$$\bigcap_{i=1}^m \langle x_1 - a_{i1}, \dots, x_n - a_{in} \rangle$$

である. 従って, イデアルの交わりを計算すればよい.

例 2 $D_1 = \{-1, 1\}^2 = \{(1, 1), (1, -1), (-1, 1), (-1, -1)\}$ の計画イデアル

$$\begin{aligned} \mathbf{I}(D_1) &= \langle x - 1, y - 1 \rangle \cap \langle x - 1, y + 1 \rangle \cap \langle x + 1, y - 1 \rangle \cap \langle x + 1, y + 1 \rangle \\ &= \langle x^2 - 1, y^2 - 1 \rangle \end{aligned}$$

2つのイデアル $I = \langle f_1, \dots, f_k \rangle, J = \langle g_1, \dots, g_\ell \rangle$ の交わりは

$$\langle tf_1, \dots, tf_k, (1-t)g_1, \dots, (1-t)g_\ell \rangle \cap k[x]$$

と計算できる [2, chap 4.3]. 例 2 の 1 目と 2 目のイデアルの交わりを Risa/Asir と Singular により計算するには以下のようにする.

Risa/Asir

```
[127] F=[t*(x-1),t*(y-1),(1-t)*(x-1),(1-t)*(y+1)];
[t*x-t,t*y-t,(-t+1)*x+t-1,(-t+1)*y-t+1]
[128] V=[t,x,y];
[t,x,y]
[129] G=nd_gr(F,V,0,2);
[-y^2+1,-x+1,-y+2*t-1]
```

Risa/Asir では、関数 `nd_gr()` によりグレブナ基底を計算できる. 計算に使用する単項式順序は第 4 引数で指定する. それぞれ, 0: 全次数逆辞書式順序 (grevlex), 1: 全次数辞書式順序 (grlex), 2: 辞書式順序 (lex) を意味する. 変数順序は第 2 引数の降順となる. 今の場合変数順序 $t \succ x \succ y$ の lex 順序に関するグレブナ基底を出力する. 従って, 最後の出力から t を含まない多項式を取り出したものが求める交わりの生成元である.

Singular

```
> ring r=0,(t,x,y),lp;
> ideal I=x-1,y-1;
> ideal J=x-1,y+1;
> ideal F=t*I,(1-t)*J;
> groebner(F);
_[1]=y2-1
_[2]=x-1
_[3]=2t-y-1
> eliminate(F,t);
_[1]=y2-1
_[2]=x-1
> intersect(I,J);
_[1]=x-1
_[2]=y2-1
_[3]=xy-x-y+1
> option(redSB);
> groebner(_);
_[1]=y2-1
_[2]=x-1
```

Singular ではまず基礎環を宣言する. 1 行目で $\mathbb{Q}[t, x, y]$, 項順序を lp (辞書式順序) で宣言している. グレブナ基底の計算には関数 `groebner()` もしくは `std()` を用いる. Risa/Asir

の時と同じくこの出力から t を含まない多項式を取り出せばよい. また, Singular には消去イデアルを計算する関数 `eliminate()` があり, それを用いることもできる. 今の場合第 2 引数で t を消去するように指定している. さらに, 直接交わりを計算する関数 `intersect()` も実装されている. ただし, この出力は簡約されていないので, `option(redSB)` で簡約基底を返すように指定してからグレブナ基底を計算している. 記号 "_" は直前の出力を表す.

2 つのイデアルの交わりが計算できたので, これを (D_1 の場合は 3 回) 繰り返せば計画イデアルは計算できる. しかし, 複数のイデアルの交わりをまとめて計算する方法 [3, chap 6.2] もある. N 個のイデアル $I_i = \langle f_{i1}, \dots, f_{ik_i} \rangle \subset k[x]$ の交わりは

$$I = \langle t_1 f_{i1}, \dots, t_i f_{ik_i} (i = 1, \dots, N), t_1 + \dots + t_N - 1 \rangle \cap k[x]$$

である. D_1 を例にすると,

Risa/Asir

```
[130] F=[t1*(x-1),t1*(y-1),t2*(x-1),t2*(y+1),
t3*(x+1),t3*(y-1),t4*(x+1),t4*(y+1),t1+t2+t3+t4-1];
[t1*x-t1,t1*y-t1,t2*x-t2,t2*y+t2,t3*x+t3,t3*y-t3,t4*x+t4,t4*y+t4,
t1+t2+t3+t4-1]
[131] V=[t1,t2,t3,t4,x,y];
[t1,t2,t3,t4,x,y]
[132] G=nd_gr(F,V,0,[[0,4],[0,2]]);
[-y^2+1,-x^2+1,(-y+1)*x+y+4*t4-1,(y+1)*x-y+4*t3-1,
(y-1)*x+y+4*t2-1,(-y-1)*x-y+4*t1-1]
```

最後の出力の多項式のうち, t_1, \dots, t_4 を含まないものが求める交わりの生成元である. ここで, 項順序の指定法が以前とは少し異なっている. (2 のままでも構わない.) `[[0,4],[0,2]]` は, まず変数リスト V の前から 4 つ目までの変数に関する 0 : grevlex 順序で比較を行い, 引き分けた場合は次の 2 つの変数に関する 0 : grevlex 順序で比較を行い, という意味である (block order). つまり, $t_1 \succ t_2 \succ t_3 \succ t_4 \gg x \succ y$ という変数順序が指定されており, 各ブロック内では全次数逆辞書式順序が用いられる.

Singular

```
> ring r=0,(t1,t2,t3,t4,x,y),(dp(4),dp(2));
> ideal I=x-1,y-1;
> ideal J=x-1,y+1;
> ideal K=x+1,y-1;
> ideal L=x+1,y+1;
> ideal F=t1*I,t2*J,t3*K,t4*L,t1+t2+t3+t4-1;
> groebner(F);
_[1]=y^2-1
_[2]=x^2-1
_[3]=4*t4-x*y+x+y-1
_[4]=2*t3-t4*x+t4+x-1
_[5]=2*t2-t3*y+t3-t4*y+t4+y-1
```

```

_[6]=t1+t2+t3+t4-1
> eliminate(F,t1*t2*t3*t4);
_[1]=y^2-1
_[2]=x^2-1
> intersect(I,J,K,L);
_[1]=y^2-1
_[2]=x^2-1
_[3]=x^2*y+x^2-y-1
> option(redSB);
> groebner(_);
_[1]=y^2-1
_[2]=x^2-1

```

同じ事を Singular で行うにはこのように入力する. block order の指定は (dp(4), dp(2)) のように項順序 (dp は grevlex 順を表す) の後にカッコで各ブロックの変数の数を入力する. また, eliminate() で複数の変数を消去したいときは変数の積を第 2 引数にとる. 更に, intersect() は 3 つ以上のイデアルの交わりも計算できる.

問 2 次の実施計画の計画イデアル (のグレブナ基底) を計算せよ.

参考文献 : [1, chap 3.2]

1. 問 1. の D_1
2. 問 1. の $D_2 \sim D_6$
3. $D_7 = \{(0, 0), (1, 0), (2, 0), (3, 0), (0, 1), (1, 1), (2, 1), (0, 2)\} \subset \mathbf{Q}^2$ (Echelon design)

問 3 (計算機に詳しい人向け) Singular の eliminate() や intersect() に相当する関数を Risa/Asir で実装せよ.

問 4 (計算機に詳しい人向け) 計画 D (例えば点のリスト) を入力とし, その計画イデアル $I(D)$ を返す関数を Risa/Asir, Singular 等で実装せよ.

注 1 CoCoA には 問 4 に相当する関数 IdealOfPoints() が実装されており, 次のようにして利用する.

```

CoCoA
Use R ::= Q[x,y];
P := [[1,1],[1,-1],[-1,1],[-1,-1]];
I := IdealOfPoints(P);
I;
Ideal(y^2 - 1, x^2 - 1)
-----
I.GBasis;

```

```
[y^2 - 1, x^2 - 1]
```

2 別名関係とイデアル所属問題

計画 D における 2 つの効果 (主効果 or 交互作用) F, G が別名関係を持つことと $F - G \in I(D)$ であることは同値である. ここでは計算機により, イデアル所属を判定してみる. それには, グレブナ基底の基本的な性質のひとつである次の命題を用いる.

命題 3 ([2] chap 2.6) G をイデアル $I \subset k[x]$ のグレブナ基底とし, $f \in k[x]$ とする. このとき, $f \in I$ であることと, f を G で割った余りが 0 であることは同値.

Risa/Asir と Singular で余り (normal form) を計算するには次のようにする.

例 3 $f = x^2y + xy^2 + y^2$, $g = x^2y + xy^2 + y^2 - x^2 - 2y$ がイデアル $I = \langle y^2 - 1, xy - 1 \rangle$ に属するかどうか.

Risa/Asir

```
[133] B=[y^2-1,x*y-1];
[y^2-1,y*x-1]
[134] V=[x,y];
[x,y]
[135] GB=nd_gr(B,V,0,0);
[x-y,y^2-1]
[136] F=x^2*y+x*y^2+y^2;
y*x^2+y^2*x+y^2
[137] p_nf(F,GB,V,0);
2*y+1
[138] G=y*x^2+y^2*x+y^2-x^2-2*y;
(y-1)*x^2+y^2*x+y^2-2*y
[139] p_nf(G,GB,V,0);
0
```

まずグレブナ基底を計算し, $p_nf()$ によって余り (の定数倍) を計算する. $p_nf()$ の第 4 引数は割り算に使用する項順序の指定であるのでグレブナ基底計算に用いた項順序と同じものを指定しないとイデアル所属が正しく判定されない. この計算によって, $f \notin I$, $g \in I$ であることが分かった. Singular では次のようにする.

Singular

```
> ring r=0,(x,y),dp;
> ideal I=y^2-1,x*y-1;
> ideal GB=groebner(I);
```

```

> poly f=x^2*y+x*y^2+y^2;
> reduce(f,GB);
2y+1
> poly g=y*x^2+y^2*x+y^2-x^2-2*y;
> reduce(g,GB);
0

```

問 5 D_4 において, 共変量行列を考える.

1. このとき, 主効果 C と交互作用 AD は同じ列を生成することを確認せよ.
2. 主効果 C と交互作用 AD が別名関係にあることをイデアル所属を判定することで確認せよ.
3. $\{A^{e_1}B^{e_2}C^{e_3}D^{e_4} \mid e_i = 0, 1\}$ の間の別名関係を全て列挙せよ.

問 6 D_6 について,

1. AB と AC , AB と DE , AC と $ABEF$ がそれぞれ別名関係にあるかどうか判定せよ.
2. 別名関係を全て列挙するプログラムを書け.

問 7 D_4 について, 主効果のみからなるモデル $M : A/B/C/D$ を考える.

1. 問 5 で列挙した別名関係に注意して, M に 2 因子交互作用を 1 つ追加した識別可能な階層モデルを全て挙げよ.
2. M に 3 因子交互作用を追加して識別可能な階層モデルを作ることができるか.
3. M に交互作用を追加して得られる識別可能な階層モデルのグラフを作れ. (1. の続きを飽和モデルになるまで行え.)

参考文献

- [1] Pistone, G. Riccomagno, E. Wynn, H. P. (2000). *Algebraic Statistics, Computational Commutative Algebra in Statistics*. Chapman & Hall.
- [2] D. Cox, J. Little and D. O'Shea. (1992). *Ideals, Varieties, and Algorithms*, Springer-Verlag, New York.
- [3] T. Becker, V. Weispfenning. (1993). *Gröbner Bases*, Springer-Verlag, New York-Berlin-Heidelberg.
- [4] Asir User's Manual,
<http://www.math.kobe-u.ac.jp/OpenXM/Current/doc/asir2000/html-jp/man.toc.html>
- [5] Singular Online Manual,
<http://www.singular.uni-kl.de/Manual/latest/index.htm>

1 計画 (design) と計画イデアル (design ideal)

問 1 次の実施計画の計画点を具体的に書け.

1. D_1 : 2^2 -完全実施計画 (水準 $\{-1, 1\}$)
2. D_2 : 2^2 -完全実施計画 (水準 $\{0, 1\}$)
3. D_3 : 3^3 -完全実施計画 (水準 $\{-1, 0, 1\}$) の中で少なくとも1つの変数が0であるもの.
4. D_4 : 別名関係 $ACD = I$ をもつ 2^{4-1} -一部実施計画 (水準 $\{-1, 1\}$)
5. D_5 : 別名関係 $ABC^2 = BCD = I$ ($a + b + 2c \equiv 0 \pmod{3}, b + c + d \equiv 0 \pmod{3}$) を持つ 3^{4-2} -一部実施計画 (水準 $\{-1, 0, 1\}$)
6. D_6 : 別名関係 $ABDE = ACDF = BCDG = I$ を持つ 2^{7-3} -一部実施計画 (水準 $\{-1, 1\}$)

解答 ファイル `design_data.rr` に点のデータを用意してある.

問 2 次の実施計画の計画イデアル (のグレブナ基底) を計算せよ.

1. 問 1. の D_1
2. 問 1. の $D_2 \sim D_6$
3. $D_7 = \{(0, 0), (1, 0), (2, 0), (3, 0), (0, 1), (1, 1), (2, 1), (0, 2)\} \subset \mathbf{Q}^2$ (Echelon design)

解答 $\mathbf{I}(D_4), \mathbf{I}(D_5), \mathbf{I}(D_6)$ は lex 順, それ以外は grevlex 順 に関するグレブナ基底.

- $\mathbf{I}(D_1) = \langle x^2 - 1, y^2 - 1 \rangle$
- $\mathbf{I}(D_2) = \langle x^2 - x, y^2 - y \rangle$
- $\mathbf{I}(D_3) = \langle x^3 - x, y^3 - y, z^3 - z, xyz \rangle$
- $\mathbf{I}(D_4) = \langle a - cd, b^2 - 1, c^2 - 1, d^2 - 1 \rangle$
- $\mathbf{I}(D_5) = \langle 2a - (2d^2 - d)c^2 - (3d^2 - 2)c + 2d^2, 2b - 3dc^2 - (3d^2 - 2)c + 2d, c^3 - c, d^3 - d \rangle$
- $\mathbf{I}(D_6) = \langle a - ceg, b - cef, c^2 - 1, d - efg, e^2 - 1, f^2 - 1, g^2 - 1 \rangle$
- $\mathbf{I}(D_7) = \langle x^4 - 6x^3 + 11x^2 - 6x, x^3y - 3x^2y + 2xy, xy^2 - xy, y^3 - 3y^2 + 2y \rangle$

問 3 (計算機に詳しい人向け) Singular の `eliminate()` や `intersect()` に相当する関数を Risa/Asir で実装せよ.

解答 `design.rr` の関数 `eliminate()` や `intersects()` 等が一つの解答例.

問 4 (計算機に詳しい人向け) 計画 D (例えば点のリスト) を入力とし, その計画イデアル $I(D)$ を返す関数を Risa/Asir, Singular 等で実装せよ.

解答 `design.rr` の関数 `design_ideal()` が一つの解答例.

2 別名関係とイデアル所属問題

問 5 D_4 において, 共変量行列を考える.

- このとき, 主効果 C と交互作用 AD は同じ列を生成することを確認せよ. 主効果の共変量行列 M の転置行列 M' は

$$M' = \begin{matrix} A \\ B \\ C \\ D \end{matrix} \begin{pmatrix} 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \end{pmatrix}$$

であるので, AD に対する列は $(1 \ -1 \ 1 \ -1 \ 1 \ -1 \ 1 \ -1)$ となり, C と AD は同じ列を生成することが分かる.

- 主効果 C と交互作用 AD が別名関係にあることをイデアル所属を判定することで確認せよ.

```
Risa/Asir
[189] I4=[a-c*d,b^2-1,c^2-1,d^2-1];
[a-d*c,b^2-1,c^2-1,d^2-1]
[190] p_nf(c-a*d,I4,[a,b,c,d],2);
0
```

- $\{A^{e_1}B^{e_2}C^{e_3}D^{e_4} \mid e_i = 0, 1\}$ の間の別名関係を全て列挙せよ. $I = ACD, A = CD, B = ABCD, C = AD, D = AC, AB = BCD, BC = ABD, BD = ABC$ の 8 つの別名関係があることが分かる. プログラム `alias-1.rr` を参照.

問 6 D_6 について,

- AB と AC , AB と DE , AC と $ABEF$ がそれぞれ別名関係にあるかどうか判定せよ.

Risa/Asir

```
[211] I6=[a-c*e*g,b-c*e*f,c^2-1,d-e*f*g,e^2-1,f^2-1,g^2-1];  
[a-g*e*c,b-f*e*c,c^2-1,d-g*f*e,e^2-1,f^2-1,g^2-1]  
[212] p_nf(a*b-a*c,I6,[a,b,c,d,e,f,g],2);  
-g*e+g*f  
[213] p_nf(a*b-d*e,I6,[a,b,c,d,e,f,g],2);  
0  
[214] p_nf(a*c-a*b*e*f,I6,[a,b,c,d,e,f,g],2);  
0
```

従って, AB と AC は別名関係にないが, AB と DE , AC と $ABEF$ は別名関係にある.

2. 別名関係を全て列挙するプログラムを書け.
各単項式に対して, 計画イデアルで割った余りが同じという同値関係で同値類を取ればよい. プログラム `alias-2.rr` を参照.

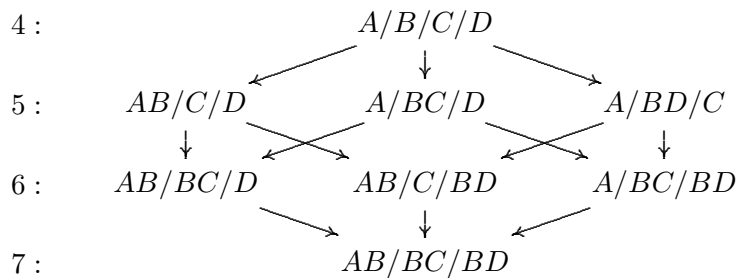
問 7 D_4 について, 主効果のみからなるモデル $M : A/B/C/D$ を考える.

1. 問 5 で列挙した別名関係に注意して, M に 2 因子交互作用を 1 つ追加した識別可能な階層モデルを全て挙げよ.
主効果と交絡していない 2 因子交互作用は AB , BC , BD の 3 つなので,

$$AB/C/D, \quad A/BC/D, \quad A/BD/C$$

の 3 つが求める識別可能な階層モデルである.

2. M に 3 因子交互作用を追加して識別可能な階層モデルを作ることができるか.
3 因子交互作用を追加すると, それに含まれる 2 因子交互作用が 3 つ追加されることになる. 主効果の 4 つと定数項とを合わせると全部で 9 つになるが, 計画は 8 点であるので必ず少なくとも 1 組が交絡する. したがって, M に 3 因子交互作用を追加した形の識別可能な階層モデルは作れない.
3. M に交互作用を追加して得られる識別可能な階層モデルのグラフを作れ. (1. の続きを飽和モデルになるまで行え.)



9月16日(水曜日)

(講義2) 10:00~12:00

「D-加群とグレブナー基底」

高山 信毅(神戸大学大学院理学研究科・JST CREST)

[概要:パラメータ付き積分の満たす微分方程式系を計算する問題を中心として、D-加群のアルゴリズムを紹介する。特に holonomic rank の計算、Pfaffian への変換、ODE section の計算、b-関数の計算、積分アルゴリズムについて解説する。付録として、A-超幾何方程式の b-関数を計算すると多面体の計算がいろいろできることも紹介する。]

(演習2) 14:00~

「D-加群とグレブナー基底」

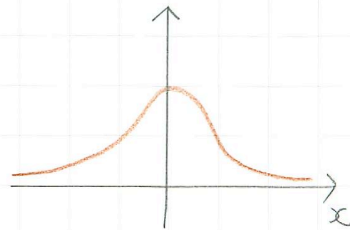
中山 洋将(神戸大学大学院理学研究科・JST CREST)

西山 絢太(神戸大学大学院理学研究科・JST CREST)

[概要:Risa/Asir、Macaulay2、kan/sm1 などを利用し、holonomic rank の計算、Pfaffian への変換、ODE section の計算、b-関数の計算、積分アルゴリズムなどを実習する。可能であれば、Maple を利用して常微分方程式の解の級数展開の方法も実習する。]

やりた い こと の 例 に よ り 説 明

$$N(x, \frac{1}{\beta}) = \frac{1}{\sqrt{2\pi}} \sqrt{\beta} e^{-\frac{\beta}{2}x^2}, \quad \beta > 0.$$



問. $m(\beta) = \int_{-\infty}^{\infty} x^2 N(x, \frac{1}{\beta}) dx$ は β の 関 数 と して ど ん な 関 数 だ? (running example)

Step 1. (微分作用素環 D の ア イ ン ヴ ィ ズ ム で.) $m(\beta)$ の みた す 微 分 方 程 式 を 求 め る.

Step 2. その方程式を D の ア イ ン ヴ ィ ズ ム や 数 値 解 析 で 解 析 し $m(\beta)$ の 性 質 を い っ ぽ う.

部分積分を $\int_{-\infty}^{\infty} e^{-x^2} dx = \sqrt{\pi}$ で. $m(\beta) = \frac{1}{\beta}$

$$K(x, \beta) = x^2 e^{-\frac{\beta}{2}x^2} \quad \text{とか.}$$

$$\tilde{m}(\beta) = \int_{-\infty}^{\infty} K(x, \beta) dx = \sqrt{2\pi} \beta^{-\frac{3}{2}}$$

Step 1.

① $K(x, \beta)$ の みた す 線 型 微 分 方 程 式 系 を 求 め る.

② 上 っ ち. $\int_{-\infty}^{\infty} K(x, \beta) dx$ が β に つ い て みた す 線 型 微 分 方 程 式 を 求 め る.

$$\frac{\partial K}{\partial \beta} = -\frac{x^2}{2} K \quad \leadsto \quad \left(\frac{\partial}{\partial \beta} + \frac{x^2}{2}\right) \cdot K = 0 \quad \text{--- ①}$$

$$\begin{aligned} \frac{\partial K}{\partial x} &= 2x e^{-\frac{\beta}{2}x^2} + x^2(-\beta x) e^{-\frac{\beta}{2}x^2} \\ &= \frac{2}{x} K - \beta x K \quad \leadsto \quad \left(x \frac{\partial}{\partial x} - 2 + \beta x^2\right) \cdot K = 0 \quad \text{--- ②} \end{aligned}$$

Prop. [Oaku, p.166] [SST, p.227] 積分法.

$$\left[l(\beta, \frac{\partial}{\partial \beta}) + \frac{\partial}{\partial x} l_1(\beta, x, \frac{\partial}{\partial \beta}, \frac{\partial}{\partial x}) \right] \cdot K = 0 \quad \text{--- ③}$$

$$\Rightarrow l \cdot \int_a^b K(x, \beta) dx + [l_1 \cdot K]_a^b = 0$$

$$\text{☹} \int_a^b [l + \frac{\partial}{\partial x} l_1] \cdot K dx = 0 \quad \therefore \quad l \int_a^b K dx + \int_a^b \frac{\partial}{\partial x} (l_1 \cdot K) dx = 0 //$$

$$\qquad \qquad \qquad [l_1 \cdot K]_a^b$$

① ② より ③ の 形 の 微 分 作 用 素 を っ け ば だ り.

$$\frac{\partial}{\partial \beta} + \frac{x^2}{2} \quad \text{--- ①}$$

$$x \frac{\partial}{\partial x} - 2 + \beta x^2 = \frac{\partial}{\partial x} x - 3 + \beta x^2 \quad \text{--- ②}$$

βx^2 の項をけすには、 $\beta \times ① - \frac{1}{2} \times ②$

$$\beta \frac{\partial}{\partial \beta} + \frac{\partial}{\partial x} \left(-\frac{1}{2} x \right) + \frac{3}{2} \quad \text{--- } l$$

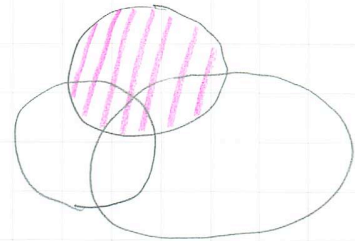
$$\text{--- } \frac{\partial}{\partial x} l_1$$

Prop 5.1. $\left(\beta \frac{\partial}{\partial \beta} + \frac{3}{2} \right) \cdot \tilde{m}(\beta) = 0$

Step 2.

$$\left(\beta \frac{\partial}{\partial \beta} + \frac{3}{2} \right) \cdot \tilde{m}(\beta) = 0 \text{ を解くと、 } \tilde{m}(\beta) = C \cdot \beta^{-\frac{3}{2}} \quad C \text{ は任意定数}$$

より複雑な10次元関数の積分を1次元にも使える。



Step 1 を D の行"ピル"の概念で整理.

$$D = \mathbb{Q} \langle x, y, \partial_x, \partial_y \rangle \quad \text{微分作用素環} \quad (\text{Oaku, 3章})$$

$$\partial_x x = x \partial_x + 1 \quad \partial_x y = y \partial_x$$

$$\partial_y y = y \partial_y + 1 \quad \partial_y x = x \partial_y$$

D の元は x, y の関数に作用する.

$$\partial_x \cdot f(x, y) = \frac{\partial f}{\partial x}, \quad \partial_y \cdot f(x, y) = \frac{\partial f}{\partial y}$$

$$x \cdot f(x, y) = xf, \quad y \cdot f(x, y) = yf$$

展望

$$I = \text{Ann}_D f \stackrel{\text{def}}{=} \{ l \in D \mid l \cdot f = 0 \}$$

$$\text{Ann}_D^{(n)} f \stackrel{\text{def}}{=} \{ l \in D, \text{ l は } n \text{ 階以下} \mid l \cdot f = 0 \}$$

I は D の左イデ"ピル" (もちろん有限生成)

積分消去で $l(y, \partial_y)$ をみつけることは、

$$\boxed{(I + \partial_x D) \cap \mathbb{Q} \langle y, \partial_y \rangle} \text{ の生成元をみつける}$$

↑ ↑
左イデ"ピル" 右イデ"ピル"

ことに他ならない
これを系統的に説明...

比較

$$J \subset \mathbb{Q} \langle x, y, \partial_x, \partial_y \rangle$$

$$J \cap \mathbb{Q} \langle y, \partial_y \rangle \quad \text{消去法}$$

$$[\text{CLO}, 2 \text{章}, 3章]$$

Def. $u=(u_1, u_2), v=(v_1, v_2) \quad u_i+v_i \geq 0$ [SST, §1.1]

$(u, v) \in \mathbb{R}^2 \times \mathbb{R}^2$ weight $\gamma \succ \mu \cup \nu$

$$\bar{in}_{(u,v)} \left(\sum_{\alpha, \beta, \gamma, \delta} c_{\alpha\beta\gamma\delta} x^\alpha y^\beta \partial_x^\gamma \partial_y^\delta \right)$$

= $\alpha u_1 + \beta u_2 + \gamma v_1 + \delta v_2$ が最大となる ℓ の項の和

($\gamma \succ \mu$ と (u, v) の内積)

initial term と呼ぶ

ただし

$u_1+v_1 > 0$ なら $\partial_x \in \mathcal{S} =$

$u_1+v_1 = 0$ なら $\partial_x \notin \mathcal{S}$

$u_2+v_2 > 0$ なら $\partial_y \in \mathcal{S} =$

$u_2+v_2 = 0$ なら $\partial_y \notin \mathcal{S}$

$\mathbb{Q}[x, y, \xi, \eta]$ は多項式環

例. $\bar{in}_{(0,0,1,1)} \left(y \partial_y + \frac{1}{2} y x^2 \right) = y \eta$

$\bar{in}_{(0,1,0,1)} \left(\quad \quad \quad \right) = y \partial_y$

◎ 順序 $\prec_{(u,v)}$ について. 多項式環のグレブナ基底とほぼ同様の議論が可能 (左イデアル)
[Oaku,] \downarrow def.

Th. [SST, Th 1.1.6]. G を D の左イデアル I の $\prec_{(u,v)}$ についての Gröbner basis とすると,
 $\{ \bar{in}_{(u,v)}(g) \mid g \in G \}$ が $\bar{in}_{(u,v)}(I)$ の生成元.

Def. D の左イデアル I を holonomic イデアル [SST, p.29] [Oaku,]

$\Leftrightarrow \bar{in}_{(0,0,1,1)}(I) \subset \mathbb{Q}[x, y, \xi, \eta]$ の (knull) 次元が ≤ 2

Th (Bernstein 不等式) [Björk, 一章] [堀田. 代数入門 5章]

$D \supseteq I \neq 0 \Rightarrow \dim \bar{in}_{(0,0,1,1)}(I) \geq 2$

Continho

$\mathcal{F}_k = \left\{ \sum_{s+t \leq k} c_{\alpha\beta\gamma\delta} x^\alpha y^\beta \partial_x^s \partial_y^t \right\}$ ← k 階以上下の作用素

$\dim_{\mathbb{Q}} \mathcal{F}_k / \mathcal{F}_k \cap I$ は k の多項式 $H(k)$. (k が増える)

さて

Th. [SST, Th 1.4.1]

I が holonomic イデアル $\Leftrightarrow H(k)$ の次数が ≤ 2
 $I \neq 0$,

例. $\bar{in}_{(0,0,1,1)} \left(y \partial_y + \frac{1}{2} y x^2 \right) = y \eta$

$\bar{in}_{(0,0,1,1)} \left(x \partial_x - 2 + y x^2 \right) = x \xi$

$\dim \langle x \xi, y \eta \rangle = 2$

Th. [Björk, 一章]

I が holonomic 行"列 $\Rightarrow (I + \partial_x D) \cap \mathbb{Q}\langle y, \partial_y \rangle$ は $\mathbb{Q}\langle y, \partial_y \rangle$ の holonomic 行"列 or trivial 行"列 (全体)

Fourier 変換 (Laplace 変換) [Oaku, p.88]

$F: x \mapsto -\partial_x, \partial_x \mapsto x$ D の環同型

$(I + \partial_x D) \cap \mathbb{Q}\langle y, \partial_y \rangle = (F(I) + xD) \cap \mathbb{Q}\langle y, \partial_y \rangle$ 制限消去

Th-Alg. [Oaku, p.145-159] [SST, p.19-211]

制限消去は以下の手続で計算可能。

- ① I の $(-1, 0, 1, 0)$ -Gröbner 基底を計算 $\partial_x \in H$ 生成元を g_1, \dots, g_p とする。

- ② $0 \leq i \leq m_j$ に対して

$\partial_x^i g_j = \sum_s l_{js}^i(y, \partial_y) \partial_x^s + \chi(\dots)$ と書く。

$\sum_s l_{js}^i(y, \partial_y) \partial_x^s$ 達から $\partial_x^s, s \geq 1$ を消去

ただし、 χ は $\mathbb{Q}\langle y, \partial_y \rangle$ 自由加群の元とみなして ∂_x^s 達を消去 (∂_x^0 を加けてはいいない)

m_j の逆数の方

$\partial_x = x \partial_x$

$[\cap \mathbb{Q}\langle \partial_x \rangle]$ の生成元を $h(\partial_x)$ とする。

$h(s) = 0$ かつ非負整数根をたない \Rightarrow 終了

r_0 が $h(s) = 0$ の最大整数根

$m_j + \text{ord}_{(-1, 0, 1, 0)}(g_j) = r_0$

とすると m_j をえらぶ。

[CL02, 5章]

例 $L_1 = \partial_y + \frac{1}{2} \partial_x^2$

$L_2 = -\partial_x \partial_x - 2 + y \partial_x^2 = -x \partial_x - 3 + y \partial_x^2$

$\text{sp}(L_1, L_2) = y L_1 - \frac{1}{2} L_2$
 $= y \partial_y + \frac{1}{2} x \partial_x + \frac{3}{2}$
 $= y \partial_y + \frac{3}{2} + \chi \left(\frac{1}{2} \partial_x \right)$

$L_1, x \partial_x + 3 + 2y \partial_y, y \partial_x \partial_y - x \partial_y + 2 \partial_x, 2y^2 \partial_y^2 + x^2 \partial_y + 9y \partial_y + 6$

$\partial_x(\partial_x - 1) + 2x^2 \partial_y$

$\partial_x^2 + 2 \partial_y, \partial_x^3 + 2 \partial_y \partial_x, \dots$

$\partial_x^k \partial_x(\partial_x - 1) + \partial_x^k 2x^2 \partial_y$
 $= x^{-k} \partial_x(\partial_x - 1) x^k \partial_x^k + \partial_x^k 2x^2 \partial_y$
 $= (\partial_x + k)(\partial_x + k - 1) \partial_x^k + \partial_x^k 2x^2 \partial_y$
 $= k(k-1) \partial_x^k + \chi(\dots) + \partial_x^k 2x^2 \partial_y$

このとき、 x はいい...

① $\text{Ann} \partial_y, \text{Ann}^{(i)}$

② tropical implicitization を D でできるか?

Step 2. にて. (まだ"また"...)

$(F(I) + xD) \cap \mathbb{Q}\langle x, y, \partial_y \rangle$ O.D.E. (常微分方程式) 113113 研究あり.

$D = \mathbb{Q}\langle x, y, t, \partial_x, \partial_y, \partial_t \rangle$

holonomic.

$(I + tD) \cap \mathbb{Q}\langle x, y, \partial_x, \partial_y \rangle = J$ Jの解空間をどのようににしろるか? → Gröbner 基底の方法.

例. $\langle \partial_x + 3\partial_y + 1, \partial_x^3 - \partial_y \rangle$ $A = (1, 0)$ $I_A = \langle \partial_x^3 - \partial_y \rangle$, $\int_C e^{zx + yt^3} dt$

Prop Jが holonomic ならば $\Rightarrow J \cap \mathbb{Q}\langle x, y, \partial_x \rangle \neq \{0\}$

② 言解法は 単純消法. (Oaku, P.97) [CLO, 3章 §1]

① $\mathbb{F}_R = \left\{ \sum_{s+t \leq R} C_{s,t} x^s y^t \partial_x^s \partial_y^t \right\}$

$\dim_{\mathbb{Q}} \mathbb{F}_R / \mathbb{F}_R J = O(R^2)$ $R \gg 0$

②-線型写像 $g: \mathbb{F}_R \cap \mathbb{Q}\langle x, y, \partial_y \rangle \rightarrow \mathbb{F}_R / \mathbb{F}_R J$ を考える.

\downarrow 次元 $\binom{R+3}{3} = O(R^3)$

$\dim_{\mathbb{Q}} \ker g + \dim_{\mathbb{Q}} \text{Im } g = O(R^3)$ $\therefore \ker g \neq \{0\}$ //
 $\leq O(R^2)$

$R = \mathbb{Q}\langle x, y \rangle \langle \partial_x, \partial_y \rangle$

Rでグリーナー基底の理論 OK.

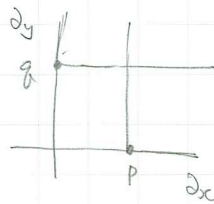
↑ 有理式 $\frac{\partial a}{\partial y}$ 係数とみる. $\partial_x a(x, y) = a(x, y) \partial_x + \frac{\partial a}{\partial x}$ かつ. はとて可換.

Prop Jが holonomic $\Rightarrow \dim_{\mathbb{Q}\langle x, y \rangle} R/RJ < +\infty$

① $J \cap \mathbb{Q}\langle x, y, \partial_x \rangle \ni a(x, y) \partial_x^p + \dots$

$J \cap \mathbb{Q}\langle x, y, \partial_y \rangle \ni b(x, y) \partial_y^q + \dots$

$\dim_{\mathbb{Q}\langle x, y \rangle} R/RJ \leq p \cdot q$ //



Th-Alg R/RJ の standard monomial の個数が Jの解空間の次元.

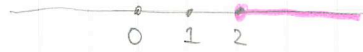
std monomial \Rightarrow Pfaffian \in 得子.

(例 2nd 講義 11)

例 11. $\langle \partial_x^2 + a\partial_x + b \rangle \subset \mathbb{Q}(x) \langle \partial_x \rangle$.

$1, \partial_x$ 是 std monomial.

$$\frac{\partial}{\partial x} \begin{pmatrix} f \\ \partial_x f \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -b & -a \end{pmatrix} \begin{pmatrix} f \\ \partial_x f \end{pmatrix} \rightarrow \frac{\partial}{\partial x} \begin{pmatrix} f_1 \\ f_2 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -b & -a \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \end{pmatrix}$$



例 12. RJ の std monomial 是 $1, \partial_x, \partial_y$ 是得子。

$$\frac{\partial}{\partial x} \begin{pmatrix} f \\ \partial_x f \\ \partial_y f \end{pmatrix} = A \begin{pmatrix} f \\ \partial_x f \\ \partial_y f \end{pmatrix} \quad \begin{array}{l} \leftarrow \partial_x^2 \text{ の normal form} \\ \leftarrow \partial_x \partial_y \text{ の normal form} \end{array}$$

$$\frac{\partial}{\partial y} \begin{pmatrix} f \\ \partial_x f \\ \partial_y f \end{pmatrix} = B \begin{pmatrix} f \\ \partial_x f \\ \partial_y f \end{pmatrix}$$

(Knoppix/Math での) マニュアルの位置

MathDoc-Search およびインターネットでのマニュアル検索に使えるキーワード.

- ① (Macaulay2) Dmodules deRham 検索 : Macaulay2
パッケージ Dmodules のマニュアル.
/usr/share/doc/Macaulay2/Dmodules/html/index.html
- ② asir yang 検索 : yang パッケージの説明書 (日本語).
/usr/local/icms2006/projects/openxm/doc/asir-contrib/ja
- ③ OpenXM 検索 : asir, kan/sm1 等の文書. (日本語, 文書を選択)
/usr/local/OpenXM/doc/index/asir-ja.html
- ④ nk_restriction 検索 : asir の制限イデアルの計算関数.
/usr/local/OpenXM/doc/index/asir-ja.html から 実験的関数を選ぶ.

⑦ ワイル代数 (微分作用素環), かけ算

Macaulay 2 (emacs イタズラース, ホモロジ-代数, グリーン

~~D-加群については~~
少し大きい計算には使えない)

load Package "Dmodules" (load("D-modules.m2"))

QQ[x, y, dx, dy, WeylAlgebra => {x => dx, y => dy}]

dx * x

x * dx

L1 = x * dx - 2 + y * x^2

L2 = dy + x^2 / 2

L1/2 - y * L2

kan/sm1

豊富な機能, 歴史あり.

asir

高速, 機能少ない.

↓
演習で

$\beta \in \mathbb{C}$ に.

x^2 の項が消失する.

$$N(x, \frac{1}{\beta}) = \frac{1}{\sqrt{2\pi}} \sqrt{\beta} e^{-\frac{\beta}{2} x^2}$$

$$m(\beta) = \int_{-\infty}^{\infty} x^2 N(x, \frac{1}{\beta}) dx = \frac{1}{\beta}$$

$$k(x, \beta) = x^2 e^{-\frac{\beta}{2} x^2}$$

$\beta \in \mathbb{C}$ と仮定.

$$L1 \cdot k = 0, L2 \cdot k = 0.$$

$$\left[\partial_x \left(\beta \frac{\partial}{\partial \beta} \right) + \frac{\partial}{\partial x} \left(L1 \left(\beta, x, \frac{\partial}{\partial \beta}, \frac{\partial}{\partial x} \right) \right) \right] \cdot k = 0$$

$$L \cdot \int_{-\infty}^{\infty} k(x, \beta) dx = 0.$$

ワイル代数

```
Macaulay 2, version 1.1.99
with packages: Classic, Core, Elimination, IntegralClosure,
               PrimaryDecomposition, SchurRings, SimpleDoc

i1 : loadPackage "Dmodules"

o1 = Dmodules

i2 : QQ[x,y,dx,dy,WeylAlgebra=>{x=>dx,y=>dy}]

o2 = QQ[x, y, dx, dy]

o2 : PolynomialRing
```

高山信毅 M2による計算例

ワイル代数

```
i3 : dx*x

o3 = x*dx + 1

o3 : QQ[x, y, dx, dy]

i4 : x*dx

o4 = x*dx

o4 : QQ[x, y, dx, dy]
```

高山信毅 M2による計算例

ワイル代数

$$i5 : L1 = x \cdot dx - 2 + y \cdot x^2$$

$$o5 = x^2 y + x^2 dx - 2$$

$$o5 : QQ[x, y, dx, dy]$$

$$i6 : L2 = dy + x^2/2$$

$$o6 = -\frac{x^2}{2} + dy$$

$$o6 : QQ[x, y, dx, dy]$$

ワイル代数

$$i7 : L1/2 - y \cdot L2$$

$$o7 = -\frac{x \cdot dx}{2} - y \cdot dy - 1$$

① 特性多様体とその次元
(characteristic variety) (Knull dimension)

$$I = \text{ideal}(L1, L2) \text{ (e)}$$

$$J = \text{inw}(I, \{0, 0, 1, 1\}) \text{ (e)}$$

$\text{in}_{(0,0,1,1)}(I)$ を計算

$$\dim(J) \text{ (e)}$$

J の Knull 次元を計算

Th holonomic \Rightarrow
 $(I + \partial_x D) \cap \mathbb{C}\langle y, \partial_y \rangle \neq 0$ である
 holonomic.

Characteristic variety, 次元

i8 : $I = \text{ideal}(L1, L2)$

$$o8 = \text{ideal}(x^2 y + x^2 dx - 2, -x^2 + dy)$$

o8 : Ideal of $\mathbb{C}\langle x, y, dx, dy \rangle$

i9 : $J = \text{inw}(I, \{0, 0, 1, 1\})$

$$o9 = \text{ideal}(2dy, x^2 dx)$$

o9 : Ideal of $\mathbb{C}\langle x, y, dx, dy \rangle$

i10 : $\dim J$

$$o10 = 2$$

◎ $(1, 0, -1, 0)$ -グレブナ基底の計算

$J = \text{gbw}(I, \{1, 0, -1, 0\})$ ☺

積分消去
制限消去 のアルゴリズム

補.

$JJ = \text{gens}(J)$ ☺

$JJ_{(0,1)}$ ☺

Jの要素をとり出すには?
matrix型で

$T = \text{matrix} \{ \{x, x+y\}, \{dx, dy\} \}$
matrixの成分をとり出すには
 $T_{(0,1)}$ などと入力.

$\text{Dintegration}(I, \{1, 0\})$ ☺

$\frac{D}{dx} \otimes \frac{D}{dy}$ (積分) の計算

補.

$\text{toString}(I)$ ☺

入力形式で出力させるには便利

(1, 0, -1, 0)-グレブナ基底の計算

```
i11 : J=gbw(I,{1,0,-1,0});
```

```
i12 : JJ=gens(J)
```

```
o12 = | x^2+2dy xdx-2ydy-2 xydy+dx dy+2x 2y^2dy^2+dx^2dy+9ydy+6 |
```

```
o12 : Matrix (QQ[x, y, dx, dy]) 1 <--- (QQ[x, y, dx, dy]) 4
```

```
i13 : JJ_(0,1)
```

```
o13 = x*dx - 2y*dy - 2
```

```
o13 : QQ[x, y, dx, dy]
```

```
i14 : Dintegration(I,{1,0})
```

```
o14 = HashTable{0 => cokernel | -2ydy-3 0 | }
      | 0 -ydy-2 |
```

```
1 => 0
```

② 消去、常微分方程式の計算

$$I = \text{ideal}(x*dx + 3*y*dy + 1, dx^3 - dy) \text{ (Ⓢ)}$$

$$\dim_{(x,y)} R/R.I < +\infty$$

$$J = \text{gbw}(I, \{0, 0, 0, 1\}) \text{ (Ⓢ)}$$

$$JJ = \text{gens}(J) \text{ (Ⓢ)}$$

$$\text{toString}(JJ_{(0,2)}) \text{ (Ⓢ)}$$

$I \cap \mathbb{Q}\langle x, y, dx \rangle$ の元は 2×2

Maple (級数解の計算など未対応)

$$? \text{DEtools}[\text{formal_sol}] ; \text{ (Ⓢ)}$$

$$\text{with}(\text{DEtools}) ; \text{ (Ⓢ)}$$

$$L := 3*y*dx^3 + x*dx + 1 ; \text{ (Ⓢ)}$$

$$Lb := \text{subs}(y=1, L) \text{ (Ⓢ)}$$

$$\text{formal_sol}(Lb, [dx, x], T, x = \text{infinity}) ; \text{ (Ⓢ)}$$

1.1.7 の見方

Step 2.
(解の性質)

$x = \infty$ の
級数解を計算

消去、常微分方程式の計算

```
i16 : I:=ideal(x*dx+3*y*dy+1,dx^3-dy)
```

```
o16 = ideal (x*dx + 3y*dy + 1, dx3 - dy);
```

```
i17 : J:=gbw(I,{0,0,0,1});
```

```
i18 : JJ:=gens J
```

```
o18 = | xdx+3ydy+1 -dx^3+dy 3ydx^3+xdx+1 |
```

```
o18 : Matrix (QQ[x, y, dx, dy]) <--- (QQ[x, y, dx, dy])3
```

```
i19 : toString JJ_(0,2)
```

```
o19 = 3*y*dx^3+x*dx+1
```

Maple による級数解の計算

```
taka@orange2(1)=> maple
|\^/| Maple 7 (IBM INTEL LINUX)
> with(DEtools);
> L:=3*y*dx^3+x*dx+1;
                                3
                                L := 3 y dx  + x dx + 1

> LL:=subs(y=1,L);
                                3
                                LL := 3 dx  + x dx + 1

> formal_sol(LL,[dx,x],T,x=infinity);
                                3      6
[[T (1 - 6 T  + O(T )), T = 1/x],
                                1/2      1      3      6      2
[T  exp(2 ----) (1 + 5/144 T  + O(T )), - 1/3 ---- = 1/x]]
                                3      T
                                T      1
```

◎ その他

Syz matrix $\{(2-y*x^2, -x^3/2, x)\}$ ⊙

$$\begin{array}{cc}
 0 & x \\
 2 & 0 \\
 x^2 & x^2y-2
 \end{array}$$

\downarrow \downarrow
 $0 \cdot \frac{\partial}{\partial x} + 2 \frac{\partial}{\partial y} + x$ $x \frac{\partial}{\partial x} + 0 \cdot \frac{\partial}{\partial y} + (x^2y-2)$

$$R = QQ[xx, p]$$

$$D = QQ[x, dx]$$

$$g = \text{map}(R, D, \{x \Rightarrow xx, dx \Rightarrow p\}) \quad \leftarrow \text{同じ変数名不可}$$

use D

$$g(x+dx+1)$$

R での計算. Pfaffian の計算. yang パッケージ

例題: $x\partial_x + 3y\partial_y + 1, \partial^3 - \partial_y$. S_x は $\theta_x = x\partial_x$. S_y は $\theta_y = y\partial_y$.

```
import("yang.rr");
yang.define_ring([x,y]);
Sx=yang.operator(x);
Sy=yang.operator(y);
L1=Sx+3*Sy+1;
L2=y*Sx*(Sx-1)*(Sx-2)-x^3*Sy;
G=yang.buchberger([L1,L2]);
yang.stdmon(G);
S1=yang.constant(1);
Base=[S1, Sy, Sy*Sy];
Pf=yang.pfaffian(Base,G);
/* Pf[0], Pf[1] */
```

$$\frac{\partial}{\partial x} \left(\begin{matrix} \\ \\ \end{matrix} \right) = \text{Pf}[0] \left(\begin{matrix} \\ \\ \end{matrix} \right)$$

$$\frac{\partial}{\partial y} \left(\begin{matrix} \\ \\ \end{matrix} \right) = \text{Pf}[1] \left(\begin{matrix} \\ \\ \end{matrix} \right)$$

↑
これは θ_x, θ_y で $\frac{\partial}{\partial x}, \frac{\partial}{\partial y}$.

余談: 美しい出力で見たい (インスピレーションが湧くように...)

TeXmacs の利用.

```
No name [1]
ファイル 編集 挿入 セッション OpenXM 書式 文書 みえ方 移る
[Icons]
I B S
OpenXM/Risa/Asir-Contrib(20090222), Copyright 2000-2
help(); [html help], ox_help(0); ox_help("keyword")
for help messages (unix version only).
openxm] quotetotex_env("conv_rule",7);
7
openxm] dx^3+x;

$$\partial_x^3 + x(\phantom{0})$$

openxm] newmat(2,2,[[1,1],[x^2,y^2]]);

$$\begin{pmatrix} 1 & 1 \\ x^2 & y^2 \end{pmatrix}$$

```

Asir の場合.

```
print_xdvi_form(式);
print_xdvi_form(Pf[0]);
```


1. 準備体操.

(a) $x\partial_x(x\partial_x - 1)$ を微分作用素環で展開せよ (手計算および計算機) $\partial_x x = x\partial_x + 1$ であるから,

$$x\partial_x(x\partial_x - 1) = x(\partial_x x)\partial - x\partial_x = x(x\partial_x + 1)\partial_x - x\partial_x = x^2\partial_x^2 + x\partial_x - x\partial_x = x^2\partial_x^2.$$

これを様々なシステムで計算させてみる.

Risa/Asir

```
[1206] V=[x,dx];
[x,dx]
[1207] P=x*dx;
x*dx
[1208] Q=x*dx-1;
x*dx-1
[1209] DP=dp_ptod(P,V);
(1)*<<1,1>>
[1210] DQ=dp_ptod(Q,V);
(1)*<<1,1>>+(-1)*<<0,0>>
[1211] DPQ=dp_weyl_mul(DP,DQ);
(1)*<<2,2>>
[1212] dp_dtop(DPQ,V);
x^2*dx^2
```

Kan/sml

```
sm1>[(x) ring_of_differential_operators 0] define_ring ;
sm1>(x*Dx). (x*Dx-1). mul ;
sm1>dehomogenize ::
x^2*Dx^2
```

Singular

```
> LIB "nctools.lib";
> ring r=0,(x,dx),dp;
> def D=Weyl();
> setring D;
> poly f = x*dx*(x*dx-1);
> f;
x^2*dx^2
```

Macaulay 2

```
i1 : R = QQ[x,dx,WeylAlgebra => {x=>dx}];
```

```
i2 : x*dx*(x*dx-1)
```

```
      2 2  
o2 = x dx
```

```
o2 : R
```

(b) $x\partial_x(x\partial_x - 1)\cdots(x\partial_x - k)$ を微分作用素環で展開せよ. $k = 2, 3, 4, 5$.

一般に, $x\partial_x(x\partial_x - 1)\cdots(x\partial_x - k) = x^{k+1}\partial_x^{k+1}$ である. (証明は帰納法)

(c) a, b, c を数字パラメータ, $\theta_x = x\partial_x, \theta_y = y\partial_y$ とするとき, $\theta_x(\theta_x + \theta_y + c - 1) - x(\theta_x + \theta_y + a)(\theta_x + b)$ を微分作用素環で展開せよ.

$$\begin{aligned} & \theta_x(\theta_x + \theta_y + c - 1) - x(\theta_x + \theta_y + a)(\theta_x + b) \\ &= \theta_x(\theta_x + c - 1) - x(\theta_x + a)(\theta_x + b) + \theta_x\theta_y - x\theta_y(\theta_x + b) \\ &= x\{x(1-x)\partial_x^2 + (c - x(a+b+1))\partial_x - ab + ((1-x)\partial_x - b)y\partial_y\} \end{aligned}$$

(d) $x \mapsto -\partial_x, \partial_x \mapsto x$ を微分作用素の Fourier 変換と呼ぶ. a をパラメータとするとき, $L = \partial_x + 2ax$ の Fourier 変換 $F(L)$ を求めよ. $F(F(L))$ を計算せよ.

$$F(L) = x + 2a(-\partial_x) = x - 2a\partial_x,$$

$$F(F(L)) = F(x - 2a\partial_x) = -\partial_x - 2ax = -L.$$

Risa/Asir

```
[1390] load("nk_restriction.rr");  
[1525] L=dx+2*a*x;  
dx+2*a*x  
[1526] FL=nk_restriction.fourier_trans(L,[x],[dx]);  
-2*a*dx+x  
[1527] FFL=nk_restriction.fourier_trans(FL,[x],[dx]);  
-dx-2*a*x
```

Kan/sm1

```
sm1>[(x,a) ring_of_differential_operators 0] define_ring ;  
sm1>(Dx+2*a*x). [(x) (Dx)] laplace0 /FL set ;  
sm1>FL ::  
-2*a*Dx+x  
sm1>FL [(x) (Dx)] laplace0 ::  
-2*x*a-Dx
```

Macaulay 2

```
i1 : loadPackage "Dmodules";  
  
i2 : R=QQ[a][x,dx,WeylAlgebra => {x=>dx}];
```

```

i3 : L=dx+2*a*x;

i4 : FL=Fourier L

o4 = x - 2a*dx

o4 : R

i5 : FFL = Fourier FL

o5 = - 2a*x - dx

o5 : R

```

2. (a) $I \cdot e^{-t-xt^3} = 0$ となる holonomic なイデアル I を求めよ. holonomic であることをどのように確かめればよいか?

$\partial_t \bullet e^{-t-xt^3} = (-1-3xt^2)e^{-t-xt^3}$, $\partial_x \bullet e^{-t-xt^3} = -t^3 e^{-t-xt^3}$ であるので, $I = \langle \partial_t + 1 + 3xt^2, \partial_x + t^3 \rangle$ は $I \cdot e^{-t-xt^3} = 0$ を満たす. n 変数の Weyl algebra D の左イデアル I が holonomic であることの定義は, I の特性イデアル $\text{ch}(I) := \text{in}_{(0,e)}(I) \subset K[x, \xi]$ の Krull 次元が n であることだったので, それを確かめればよい. I の重み $(0, 0, 1, 1)$ に関するグレブナ基底は $\{\partial_t + 1 + 3xt^2, \partial_x + t^3\}$ である (下線部が先頭項) ので, $\text{ch}(I) = \langle \xi_x, \xi_t \rangle \subset K[x, t, \xi_x, \xi_t]$. これの Krull 次元は 2 であるので, I は holonomic である.

Macaulay 2

```

i1 : loadPackage "Dmodules";

i2 : R=QQ[x,t,dx,dt,WeylAlgebra => {x=>dx,t=>dt}];

i3 : I=ideal(dt+1+3*x*t^2,dx+t^3);

o3 : Ideal of R

i4 : charIdeal(I)

o4 = ideal (dt, dx)

o4 : Ideal of QQ [x, t, dx, dt]

i5 : dim(o4)

o5 = 2

```

- (b) D/I の t についての D -加群としての積分 $D/(I + \partial_t D)$ を計算せよ. (一行コマンドでの実行 および Gröbner basis の計算へ分解しての実行.)

Risa/Asir

```
[1384] load("nk_restriction.rr");
[1519] I=[dt+1+3*x*t^2,dx+t^3];
[dt+3*t^2*x+1,dx+t^3]
[1520] nk_restriction.integration_ideal(I,[t,x],[dt,dx],[1,0]);
-- generic_bfct_and_gr :0sec(0.001976sec)
generic bfct : [[1,1],[s,1]]
S0 : 0
B_{S0} length : 1
-- fctr(BF) + base :0sec(0.0009sec)
-- integration_ideal_internal :0.004001sec(0.001181sec)
[27*x^3*dx^2+54*x^2*dx+6*x+1]
```

Macaulay 2

```
i1 : loadPackage "Dmodules";

i2 : R=QQ[t,x,dt,dx,WeylAlgebra => {t=>dt,x=>dx}];

i3 : I=ideal(dt+1+3*x*t^2,dx+t^3);

o3 : Ideal of R

i4 : Dintegration(I,{1,0})

o4 = HashTable{0 => cokernel | -27x3dx^2-54x2dx-6x-1 |}
      1 => 0
```

Kan/sml

```
sm1>(cohom.sm1) run ;
sm1>[[ (Dt+1+3 x t^2) (Dx+t^3) ] [(t)] [[(x) (t)] []]] integration ::
[ Dt+1+3 x t^2 , Dx+t^3 ] ==> [ 3*x*Dt^2+t+1 , -Dt^3+Dx ]
[ [ 3*x*Dt^2+t+1 , -Dt^3+Dx ] , [ t ] ] bfm
sm1>sm1>b-function is -s
[ [ 3*x*Dt^2+t+1 , -Dt^3+Dx ] , [ t ] , 0 , 1 ] restall1_s
Computing a free resolution ...
Resolution procedure stoped because counter == 0.
A free resolution obtained.
0-th cohomology: [ 1 , [ 27*x^3*Dx^2+54*x^2*Dx+6*x+1 ] ]
sm1>-1-th cohomology: [ 0 , [ ] ]
sm1>[ [ 1 , [ 27*x^3*Dx^2+54*x^2*Dx+6*x+1 ] ] , [ 0 , [ ] ] ]
```

• 1-step ずつ実行

I の生成元に変数 t に関する Fourier 変換を施したイデアル $F(I) = \langle t+1+3x\partial_t^2, \partial_x - \partial_t^3 \rangle$ の制限イデアルを計算する.

まず, $(t, x, \partial_t, \partial_x)$ に関する重みが $w = (-1, 0, 1, 0)$ であるようなグレブナ基底を計算する. 負の重みを含むので斉次化経由での計算が必要である.

```
Risa/Asir
[1206] FI=[t+1+3*x*dt^2,dx-dt^3];
[3*x*dt^2+t+1,dx-dt^3]
[1207] HFI=map(homogenize,FI,[t,x,dt,dx],h);
[3*x*dt^2+h^2*t+h^3,h^2*dx-dt^3]
[1208] dp_weyl_set_weight(ltov([1,0]));
[ 1 0 ]
[1209] HG=nd_weyl_gr(HFI,[t,x,dt,dx,h],0,11);
[3*x*dt^2+h^2*t+h^3,h^2*dx-dt^3,-3*h^2*x*dx+(-h^2*t-h^3)*dt-h^4,(9*h^2*x^2*dt+
(18*h^2*t-18*h^3)*x^2)*dx+6*h^2*t^2*x*dt+(6*h^4*t-6*h^5)*x-h^4*t^2-2*h^5*t-h^6,
27*h^2*x^3*dx^2+54*h^4*x^2*dx+6*h^6*x+h^4*t^3+3*h^5*t^2+3*h^6*t+h^7]
[1210] G=map(subst,HG,h,1);
[3*x*dt^2+t+1,dx-dt^3,-3*x*dx+(-t-1)*dt-1,(9*x^2*dt+(18*t-18)*x^2)*dx+6*t^2*x*
dt+(6*t-6)*x-t^2-2*t-1,27*x^3*dx^2+54*x^2*dx+6*x+t^3+3*t^2+3*t+1]
```

次いで, イデアル $F(I)$ の重み w に関する generic b -関数を計算する.

```
Risa/Asir
[1213] generic_bfct(FI,[t,x],[dt,dx],[1,0]);
s
```

generic b -関数の最大整数根は 0 である. 出力 G の生成元の w に関する階数 (ord) は順に 2, 3, 1, 1, 0 であるから最後の生成元 $F(P) := 27x^3\partial_x^2 + 54x^2\partial_x + 6x + t^3 + 3t^2 + 3t + 1$ において $t = 0$ とした, $27x^3\partial_x^2 + 54x^2\partial_x + 6x + 1$ が I の積分イデアルの生成元である.

(c) $\int_0^{+\infty} e^{-t-xt^3} dt$ (Airy 関数もどき) のみたす微分方程式を計算せよ.

$A(x) = \int_0^{+\infty} e^{-t-xt^3} dt$ とおく. (b) で計算した $F(P) = 27x^3\partial_x^2 + 54x^2\partial_x + 6x + t^3 + 3t^2 + 3t + 1 \in F(I)$ の逆 Fourier 変換は

$$\begin{aligned} P &= 27x^3\partial_x^2 + 54x^2\partial_x + 6x + \partial_t^3 + 3\partial_t^2 + 3\partial_t + 1 \\ &= (27x^3\partial_x^2 + 54x^2\partial_x + 6x + 1) + \partial_t(\partial_t^2 + 3\partial_t + 3) \in I \end{aligned}$$

である. 従って,

$$\begin{aligned}
 (27x^3\partial_x^2 + 54x^2\partial_x + 6x + 1) \bullet A(x) &= -\partial_t(\partial_t^2 + 3\partial_t + 3) \bullet A(x) \\
 &= -\int_0^{+\infty} \partial_t(\partial_t^2 + 3\partial_t + 3) \bullet e^{-t-xt^3} dt \\
 &= -\int_0^{+\infty} \frac{\partial((\partial_t^2 + 3\partial_t + 3) \bullet e^{-t-xt^3})}{\partial t} dt \\
 &= -\left[(\partial_t^2 + 3\partial_t + 3) \bullet e^{-t-xt^3}\right]_0^{+\infty} \\
 &= -\left[(9x^2t^4 - 3xt^2 - 6xt + 1)e^{-t-xt^3}\right]_0^{+\infty} \\
 &= 1
 \end{aligned}$$

つまり, $A(x)$ は微分方程式

$$(27x^3\partial_x^2 + 54x^2\partial_x + 6x + 1) \bullet A(x) = 1$$

を満たす.

- (d) (Maple が使えたら) 上の微分方程式の解の $t = 0$ での漸近展開を求めよ. この結果と数値計算で上の積分の漸近展開の (近似を) 決定せよ.

3. $f(t) = t(1-t)$ とおく.

- (a) $\text{Ann } f^s$ を求めよ.

```
Risa/Asir
[1208] ann(t*(1-t));
[(-t^2+t)*dt+(2*t-1)*s]
```

$$\text{Ann } f^s = \langle (-t^2 + t)\partial_t + (2t - 1)s \rangle$$

- (b) $(E_s - f) \bullet f^s = 0$ である. ここで $E_s \bullet F(s) = F(s + 1)$ となる差分作用素. $E_s \leftrightarrow p$, $s + 1 \leftrightarrow -p\partial_p$ なる Mellin 変換で, $E_s - f$, $\text{Ann } f^s$ を変換したイデアル J は p, t 空間での holonomic なイデアルであることを示せ.

$$J = \langle p - t(1-t), (-t^2 + t)\partial_t + (2t - 1)(-p\partial_p - 1) \rangle.$$

J が holonomic であるかどうか調べるために $\text{ch}(J)$ の Hilbert 多項式を計算してみる.

```
Risa/Asir
[1209] J=[p-t*(1-t), (-t^2+t)*dt+(2*t-1)*(-p*dp-1)];
[p+t^2-t, (-2*t+1)*p*dp+(-t^2+t)*dt-2*t+1]
[1210] V=[p, t, dp, dt];
[p, t, dp, dt]
[1211] M=newmat(5, 4, [[0, 0, 1, 1], [1, 1, 1, 1], [0, 0, 0, -1], [0, 0, -1], [0, -1]]);
[ 0 0 1 1 ]
[ 1 1 1 1 ]
[ 0 0 0 -1 ]
[ 0 0 -1 0 ]
```

```

[ 0 -1 0 0 ]
[1212] G=nd_weyl_gr(J,V,0,M);
[p+t^2-t, (-2*t+1)*p*dp+p*dt, (4*p^2-p)*dp+(2*t-1)*p*dt+4*p]
[1213] IN=[p+t^2-t, (-2*t+1)*p*dp+p*dt, (4*p^2-p)*dp+(2*t-1)*p*dt];
[p+t^2-t, (-2*t+1)*p*dp+p*dt, (4*p^2-p)*dp+(2*t-1)*p*dt]
[1214] DIN=map(dp_ptod,IN,V);
[(1)*<<0,2,0,0>>+(1)*<<1,0,0,0>>+(-1)*<<0,1,0,0>>,
(-2)*<<1,1,1,0>>+(1)*<<1,0,1,0>>+(1)*<<1,0,0,1>>,
(4)*<<2,0,1,0>>+(2)*<<1,1,0,1>>+(-1)*<<1,0,1,0>>+(-1)*<<1,0,0,1>>]
[1215] HDIN=map(dp_ht,DIN);
[(1)*<<0,2,0,0>>, (1)*<<1,1,1,0>>, (1)*<<2,0,1,0>>]
[1216] HIN=map(dp_dtop,HDIN,V);
[t^2,t*p*dp,p^2*dp]
[1214] sm1.hilbert([HIN,V]);
5/2*h^2+3/2*h+1

```

Hilbert 多項式の次数が 2 であるので J が holonomic であることが分かった。

- (c) D/J の変数 t についての D -加群としての積分を計算し, $B(s) = \int_0^1 f^s dt$ のみならず s についての漸化式 (contiguity relation) を求めよ.

Risa/Asir

```

[1384] load("nk_restriction.rr");
[1519] J=[p-t*(1-t), (-t^2+t)*dt+(2*t-1)*(-p*dp-1)];
[p+t^2-t, (-2*t+1)*p*dp+(-t^2+t)*dt-2*t+1]
[1520] FJ=nk_restriction.integration_ideal(J, [t,p], [dt,dp], [1,0]);
-- generic_bfct_and_gr :0.004sec(0.007393sec)
generic bfct : [[1,1],[s,1],[s-1,1]]
S0 : 1
B_{S0} length : 2
-- fctr(BF) + base :0sec(0.004466sec)
-- integration_ideal_internal :0.008001sec(0.005727sec)
[(4*p^2-p)*dp+2*p]

```

この積分計算より, $4p^2\partial_p - p\partial_p + 2p$ に逆 Mellin 変換を施した作用素

$$4E_s(-s-1) + (s+1) + 2E_s = -4(s+2)E_s + (s+1) + 2E_s = -2(2s+3)E_s + (s+1)$$

は $B(s) = \int_0^1 f^s dt$ を annihilate することが分かる. つまり, $-(2(2s+3)E_s - (s+1)) \bullet B(s) = 0$ が成り立つ. 従って, 求める漸化式は

$$2(2s+3)B(s+1) = (s+1)B(s).$$

- (d) この漸化式を用いて $B(10000)$ の値を求めよ.

(c) で得られた漸化式から, s が自然数なら

$$B(s) = \frac{s}{2(2s+1)} B(s-1) = \cdots = \frac{s!}{2^s(2s+1)!!} B(0)$$

が成り立つ. ここで, $B(0) = \int_0^1 dt = 1$ なので

$$B(s) = \frac{s!}{2^s(2s+1)!!} = \frac{(s!)^2}{(2s+1)!}$$

$B(10000)$ の逆数を安直に計算させてみるには以下のようにする.

```
Risa/Asir
[1217] for (B=1, S=1; S<=10000; S++) B*=2*(2*S+1)/S;
[1218] B;
結果は省略 (10進 6023桁の整数)
```

4. $f(t) = t_1 t_2 (1 - t_1 - t_2)$ について上と同様な問題.

$B_2(s) = \int_{D_2} f^s dt$, ($D_2 = \{(t_1, t_2) \mid 0 \leq t_1 \leq 1, 0 \leq t_2 \leq 1 - t_1\}$) について考える (特に, D_2 はサイクル).

```
Risa/Asir
[1384] load("nk_restriction.rr");
[1519] F=t1*t2*(1-t1-t2);
-t1*t2^2+(-t1^2+t1)*t2
[1520] A=ann(F);
[-2*t1*t2*dt2+(3*t1*t2+t1^2-t1)*dt1-3*s*t2+s, t1*t2*dt2+(t1^2-t1)*dt1-3*s*t1+s,
(t2^2+(2*t1-1)*t2)*dt2+(-2*t1*t2-t1^2+t1)*dt1]
[1521] MA=map(subst,A,s,-p*dp-1); /* s は 1 次なので単なる代入で処理 */
[-2*t1*t2*dt2+(3*t1*t2+t1^2-t1)*dt1+(3*p*t2-p)*dp+3*t2-1, t1*t2*dt2+(t1^2-t1)*
dt1+(3*p*t1-p)*dp+3*t1-1, (t2^2+(2*t1-1)*t2)*dt2+(-2*t1*t2-t1^2+t1)*dt1]
[1522] MA=cons(p-F,MA);
[t1*t2^2+(t1^2-t1)*t2+p, -2*t1*t2*dt2+(3*t1*t2+t1^2-t1)*dt1+(3*p*t2-p)*dp+3*t2-
1, t1*t2*dt2+(t1^2-t1)*dt1+(3*p*t1-p)*dp+3*t1-1, (t2^2+(2*t1-1)*t2)*dt2+(-2*t1*t
2-t1^2+t1)*dt1]
[1523] nk_restriction.integration_ideal(MA, [t1,t2,p], [dt1,dt2,dp], [1,1,0]);
-- generic_bfct_and_gr :0.016sec + gc : 0.004sec(0.02054sec)
generic bfct : [[1,1],[s,1],[s-1,2]]
S0 : 1
B_{S0} length : 3
-- fctr(BF) + base :0.004sec(0.004sec)
-- integration_ideal_internal :0.004sec(0.002161sec)
[(27*p^3-p^2)*dp^2+(54*p^2-p)*dp+6*p]
```


$(27p^3 - p^2)\partial_p^2 + (54p^2 - p)\partial_p + 6p$ の逆 Mellin 変換は

$$\begin{aligned} (27p - 1)p^2\partial_p^2 + (54p - 1)p\partial_p + 6p &= (27p - 1)p\partial_p(p\partial_p - 1) + (54p - 1)p\partial_p + 6p \\ &= (27p - 1)(p\partial_p)^2 + 27p\partial_p + 6p \\ &\rightarrow (27E_s - 1)(-s - 1)^2 + 27E_s(-s - 1) + 6E_s \\ &= (27(s + 2)^2 - 27(s + 2) + 6)E_s - (s + 1)^2 \\ &= 3(3s + 4)(3s + 5)E_s - (s + 1)^2 \end{aligned}$$

であるから, $B_2(s)$ の満たす漸化式は

$$3(3s + 4)(3s + 5)B_2(s + 1) = (s + 1)^2 B_2(s).$$

従って, $B(0) = 1/2$ であるので s が自然数なら,

$$\begin{aligned} B_2(s) &= \frac{s^2}{3(3s + 1)(3s + 2)} \cdots \frac{2^2}{3 \cdot 7 \cdot 8} \cdot \frac{1^2}{3 \cdot 4 \cdot 5} \cdot B_2(0) \\ &= \frac{s^2}{3(3s + 1)(3s + 2)} \cdots \frac{2^2}{3 \cdot 7 \cdot 8} \cdot \frac{1^2}{3 \cdot 4 \cdot 5} \cdot \frac{1}{1 \cdot 2} \\ &= \frac{(s!)^2}{3^s(3s + 1)!!(3s + 2)!!} \\ &= \frac{(s!)^3}{(3s + 2)!}. \end{aligned}$$

ちなみに, $B_2(10000)$ の逆数は 10 進 14,319 桁の整数.

(推測) $f(t) = (\prod_{i=1}^n t_i) \cdot (1 - \sum_{i=1}^n t_i)$ の領域 $D_n = \{(t_1, \dots, t_n) \mid 0 \leq t_i \leq 1 - \sum_{j=1}^{i-1} t_j (1 \leq i \leq n)\}$ に対する積分 $B_n(s) = \int_{D_n} f^s dt$ は s が自然数の時,

$$B_n(s) = \frac{(s!)^{n+1}}{((n + 1)s + n)!}.$$

5. $A = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & -1 \\ 0 & 0 & 1 & 1 & -1 \end{pmatrix}$ とおく.

(a) A に付随する A -超幾何系の生成系 J を求めよ.

Risa/Asir

```
[1236] A=[[1,1,1,1,1],[0,1,1,0,-1],[0,0,1,1,-1]];
[[1,1,1,1,1],[0,1,1,0,-1],[0,0,1,1,-1]]
[1237] B=[b1,b2,b3];
[b1,b2,b3]
[1238] G=sm1.gkz([A,B]);
[[x5*dx5+x4*dx4+x3*dx3+x2*dx2+x1*dx1-b1,-x5*dx5+x3*dx3+x2*dx2-b2,
-x5*dx5+x4*dx4+x3*dx3-b3,dx2*dx4-dx1*dx3,dx3*dx5-dx1^2,dx3^2*dx5-
dx1*dx2*dx4,dx3^3*dx5-dx2^2*dx4^2],[x1,x2,x3,x4,x5]]
```

(b) β_i をパラメータとするとき J と ∂_i で生成されるイデアルの生成元を求めよ (β_i の多項式となる). A の決める多面体との関係は?

Risa/Asir

```
[1239] J=G[0]$
[1240] VDV=[x1,x2,x3,x4,x5,b1,b2,b3,dx1,dx2,dx3,dx4,dx5,db1,db2,db3];
[x1,x2,x3,x4,x5,b1,b2,b3,dx1,dx2,dx3,dx4,dx5,db1,db2,db3]
[1241] M=newmat(17,16,[[1,1,1,1,1,0,0,0,1,1,1,1,1,1,1,1]])$
[1242] for (I=0;I<16;I++) M[1][I]=1;
[1243] for (I=0;I<15;I++) M[2+I][15-I]=-1;
[1244] M;
(略)
[1245] J1=nd_weyl_gr(cons(dx1,J),VDV,0,M);
[b1^4+(-3*b2^2+4*b3*b2-3*b3^2)*b1^2+(2*b2^3-2*b3*b2^2-2*b3^2*b2+2*b3^3)
*b1-2*b3*b2^3+5*b3^2*b2^2-2*b3^3*b2,dx1,...(略)
..., -2*x5*dx2*dx5+(b1-b2)*dx2]
[1246] P1=J1[0];
b1^4+(-3*b2^2+4*b3*b2-3*b3^2)*b1^2+(2*b2^3-2*b3*b2^2-2*b3^2*b2+2
*b3^3)*b1-2*b3*b2^3+5*b3^2*b2^2-2*b3^3*b2
[1247] fctr(P1);
[[1,1],[b1-b3,1],[b1-b2,1],[b1-b2+2*b3,1],[b1+2*b2-b3,1]]
```

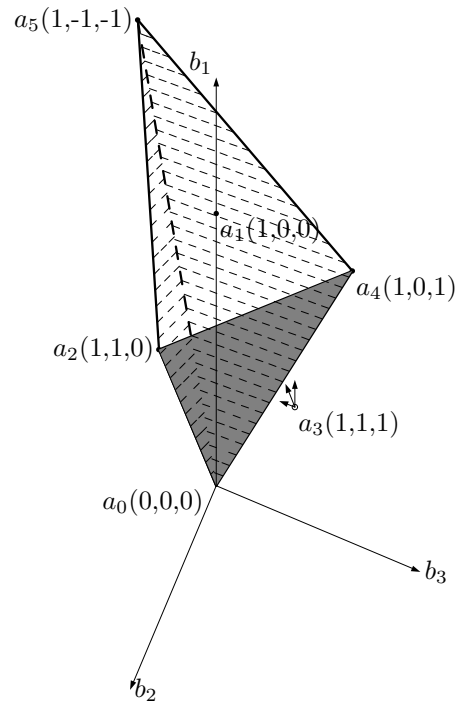
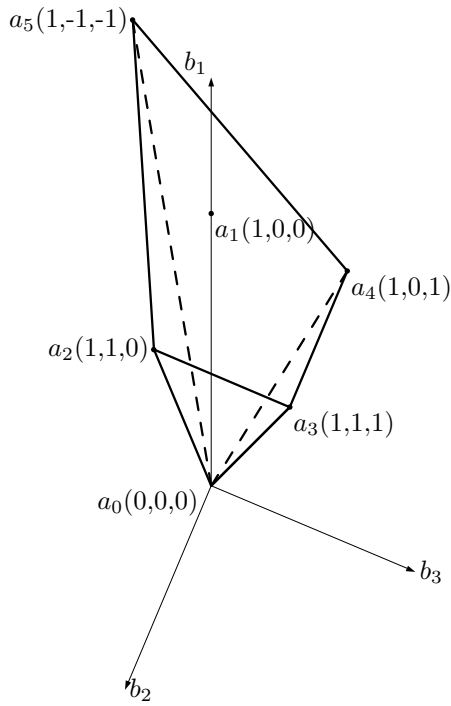
$$(J + \langle \partial_1 \rangle) \cap \mathbf{C}[b_1, b_2, b_3] = \langle (b_1 - b_2)(b_1 - b_3)(b_1 - b_2 + 2b_3)(b_1 + 2b_2 - b_3) \rangle$$

他も同様にすれば,

$$\begin{aligned} (J + \langle \partial_2 \rangle) \cap \mathbf{C}[b_1, b_2, b_3] &= \langle (b_1 - b_3)(b_1 + 2b_2 - b_3)(b_1 + 2b_2 - b_3 - 1)(b_1 + 2b_2 - b_3 - 2) \rangle \\ (J + \langle \partial_3 \rangle) \cap \mathbf{C}[b_1, b_2, b_3] &= \langle (b_1 - b_2 + 2b_3)(b_1 - b_2 + 2b_3 - 1)(b_1 + 2b_2 - b_3)(b_1 + 2b_2 - b_3 - 1) \rangle \\ (J + \langle \partial_4 \rangle) \cap \mathbf{C}[b_1, b_2, b_3] &= \langle (b_1 - b_2)(b_1 - b_2 + 2b_3)(b_1 - b_2 + 2b_3 - 1)(b_1 - b_2 + 2b_3 - 2) \rangle \\ (J + \langle \partial_5 \rangle) \cap \mathbf{C}[b_1, b_2, b_3] &= \langle (b_1 - b_2)(b_1 - b_2 - 1)(b_1 - b_3)(b_1 - b_3 - 1) \rangle. \end{aligned}$$

● 幾何的な意味

行列 $A = (a_1, a_2, a_3, a_4, a_5)$ の各列を (b_1, b_2, b_3) -空間の点とみなし, 原点 a_0 を頂点とする下図左のような cone を考える. 例として, ∂_3 を追加した場合を考える. 点 a_3 を除いてできる下図右のような cone を作り, a_3 の位置に光源を置く. このとき, 光が当たらない facet は $a_0a_2a_5$ と $a_0a_4a_5$ の2つである (図の斜線部). この2つの facet を通る平面の方程式はそれぞれ, $b_1 - b_2 + 2b_3 = 0$, $b_1 + 2b_2 - b_3 = 0$ であり, これらは上で求めたイデアルの生成元の因子として表れており, 残りの因子はこれらの整数差である. 整数差の因子の表れ方等の詳細は [SST2] を参照.



6. Appell の方程式 F_1 を Pfaffian 形式に変換せよ.

$$\begin{aligned} &\theta_x(\theta_x + \theta_y + c - 1) - x(\theta_x + \theta_y + a)(\theta_x + b) \\ &\theta_y(\theta_x + \theta_y + c - 1) - y(\theta_x + \theta_y + a)(\theta_y + b') \end{aligned}$$

Risa/Asir

```
[1221] load("yang.rr");
[1625] yang.define_ring([x,y]);
{[euler,[x,y]],[x,y],[0,0],[0,0],[dx,dy]}
[1626] S=dx+dy;
dx+dy
[1627] L1=yang.mul(dx,S+c-1)-x*yang.mul(S+a,dx+b1);
(-x+1)*dx^2+((-x+1)*dy+(-a-b1)*x+c-1)*dx-b1*x*dy-b1*a*x
[1628] L2=yang.mul(dy,S+c-1)-y*yang.mul(S+a,dy+b2);
((-y+1)*dy-b2*y)*dx+(-y+1)*dy^2+((-a-b2)*y+c-1)*dy-b2*a*y
[1629] G=yang.gr([L1,L2]);
[((b2*y*x-b2*y)*dx+((y-1)*x-y^2+y)*dy^2+(((a-b1+b2)*y-c+b1+1)
*x+(-a-b2)*y^2+(c-1)*y)*dy+b2*a*y*x-b2*a*y^2)/((y-1)*x-y^2+y),
(((x+y)*dy+b2*y)*dx-b1*x*dy)/(-x+y),((-x^2+(y+1)*x-y)*dx^2+(
(-a-b1)*x^2+((a+b1-b2)*y+c-1)*x+(-c+b2+1)*y)*dx+(b1*y-b1)*x*dy
-b1*a*x^2+b1*a*y*x)/(-x^2+(y+1)*x-y)]
[1630] yang.stdmon(G);
[dx,dy,1]
[1631] Base=[1,dx,dy];
[1,dx,dy]
```

```

[1632] Pf=yang.pf(Base,G)$
[1633] Pf[0];
[ 0 (1)/(x) 0 ]
[ (-b1*a)/(x-1) ((-a-b1)*x^2+((a+b1-b2)*y+c-1)*x+(-c+b2+1)*y)/(x^3+(-y-1)*x^2+y*x) (b
[ 0 (b2*y)/(x^2-y*x) (-b1)/(x-y) ]
[1634] Pf[1];
[ 0 0 (1)/(y) ]
[ 0 (b2)/(x-y) (-b1*x)/(y*x-y^2) ]
[ (-b2*a)/(y-1) (-b2*x+b2)/((y-1)*x-y^2+y) (((-a+b1-b2)*y+c-b1-1)*x+(a+b2)*y)^2+(-c+1)

```

基底を $F = (f, \theta_x f, \theta_y f)^T$ とした時,

$$\partial_x F = \text{Pf}[0]F, \quad \partial_y F = \text{Pf}[1]F$$

を満たすことを意味する.

7. (研究課題, (modified) marginal likelihood integral)

$$f_2 = (px_1x_2 + (1-p)y_1y_2)(px_1(1-x_2) + (1-p)y_1(1-y_2))$$

とおくとき, $\int_C f_2^s dp dx_1 dx_2 dy_1 dy_2$ が s についてみたす漸化式を積分アルゴリズムで求めよ. 参考: ml1.rr

参考文献

- [CLO] D.Cox, J.Little, D.O'shea, **グレブナ基底と代数多様体入門**, *Ideals, Varieties, and Algorithms*, (1992, 1997), Springer.
- [CLO2] D.Cox, J.Little, D.O'shea, **グレブナー基底**, *Using Algebraic Geometry*, (1998), Springer.
- [Oaku] 大阿久俊則, **D 加群と計算数学**, 2002, 朝倉書店.
- [SST] M.Saito, B.Sturmfels, N.Takayama, *Gröbner Deformations of Hypergeometric Differential Equations*, 2000, Springer.
- [SST2] M.Saito, B.Sturmfels, N.Takayama, Hypergeometric polynomials and Integer Programming, *Compositio Mathematica*, **115**, (1999) 185–204.

1. 準備体操.

- (a) $x\partial_x(x\partial_x - 1)$ を微分作用素環で展開せよ (手計算および計算機)
- (b) $x\partial_x(x\partial_x - 1) \cdots (x\partial_x - k)$ を微分作用素環で展開せよ. $k = 2, 3, 4, 5$.
- (c) a, b, c を数字パラメータ, $\theta_x = x\partial_x, \theta_y = y\partial_y$ とするとき, $\theta_x(\theta_x + \theta_y + c - 1) - x(\theta_x + \theta_y + a)(\theta_x + b)$ を微分作用素環で展開せよ.
- (d) $x \mapsto -\partial_x, \partial_x \mapsto x$ を微分作用素の Fourier 変換と呼ぶ. a をパラメータとすると, $L = \partial_x + 2ax$ の Fourier 変換 $F(L)$ を求めよ. $F(F(L))$ を計算せよ.
2. (a) $I \cdot e^{-t-xt^3} = 0$ となる holonomic なイデアル I を求めよ. holonomic であることをどのように確かめればよいか?
- (b) D/I の t についての D -加群としての積分 $D/(I + \partial_t D)$ を計算せよ. (一行コマンドでの実行 および Gröbner basis の計算へ分解しての実行.)
- (c) $\int_0^{+\infty} e^{-t-xt^3} dt$ (Airy 関数もどき) のみならず微分方程式を計算せよ.
- (d) (Maple が使えたら) 上の微分方程式の解の $x = 0, x = \infty$ での漸近展開を求めよ. この結果と数値計算で上の積分の漸近展開の (近似を) 決定せよ.
3. $f(t) = t(1-t)$ とおく.
- (a) $\text{Ann } f^s$ を求めよ.
- (b) $(E_s - f) \bullet f^s = 0$ である. ここで $E_s \bullet F(s) = F(s+1)$ となる差分作用素. $E_s \leftrightarrow p, s+1 \leftrightarrow -p\partial_p$ なる Mellin 変換で, $E_s - f, \text{Ann } f^s$ を変換したイデアル J は p, t 空間での holonomic なイデアルであることを示せ.
- (c) D/J の変数 t についての D -加群としての積分を計算し, $B(s) = \int_0^1 f^s dt$ のみならず s についての漸化式 (contiguity relation) を求めよ.
- (d) この漸化式を用いて $B(10000)$ の値を求めよ.
4. $f(t) = t_1 t_2 (1 - t_1 - t_2)$ について上と同様な問題.
5. $A = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & -1 \\ 0 & 0 & 1 & 1 & -1 \end{pmatrix}$ とおく.

- (a) A に付随する A -超幾何系の生成系 J を求めよ。
 (b) β_i をパラメータとするとき J と ∂_i で生成されるイデアルの生成元を求めよ (β_i の多項式となる). A の決める多面体との関係は?

6. Appell の方程式 F_1 を Pfaffian 形式に変換せよ.

$$\begin{aligned} &\theta_x(\theta_x + \theta_y + c - 1) - x(\theta_x + \theta_y + a)(\theta_x + b) \\ &\theta_y(\theta_x + \theta_y + c - 1) - y(\theta_x + \theta_y + a)(\theta_y + b') \end{aligned}$$

7. (研究課題, (modified) marginal likelihood integral)

$$f_2 = (px_1x_2 + (1-p)y_1y_2)(px_1(1-x_2) + (1-p)y_1(1-y_2))$$

とおくとき, $\int_C f_2^s dp dx_1 dx_2 dy_1 dy_2$ が s についてみたす漸化式を積分アルゴリズムで求めよ. 参考: ml1.rr

参考文献

- [CLO] D.Cox, J.Little, D.O'shea, グレブナ基底と代数多様体入門, *Ideals, Varieties, and Algorithms*, (1992, 1997), Springer.
 グレブナ基底, Buchberger アルゴリズムの入門. 消去法. なお効率的な計算な計算法については, 野呂, 横山, グレブナ基底の計算 基礎編 – 計算代数入門, 東大出版会, 2003. どちらの本も Weyl 代数でそのまま通用することが沢山書いてある.
- [CLO2] D.Cox, J.Little, D.O'shea, グレブナー基底, *Using Algebraic Geometry*, (1998), Springer
 続編. 自由加群の部分加群のグレブナ基底についてはこちらを参照.
- [Oaku] 大阿久俊則, D 加群と計算数学, 2002, 朝倉書店.
 D 加群のアルゴリズムについての入門書. とはいいつつ, 本質的なアイデアはすべて書いてあると思う.
- [SST] M.Saito, B.Sturmfels, N.Takayama, *Gröbner Deformations of Hypergeometric Differential Equations*, 2000, Springer.
 積分のみたす微分方程式系のグレブナ基底を活用した, 積分の漸近展開の理論とアルゴリズム. とくに, A -超幾何系について詳しい. 不確定特異点を持つ場合, 積分領域がサイクルでない場合は未開拓. この本を読みきるには, 常微分方程式 (高野, 常微分方程式, 朝倉), 多面体の幾何 (G.Ziegler, *Lectures on Polytopes*, Springer), グレブナ基底, 可換環論などに, ある程度の経験が必要.

- [1] C.M.Bishop, *パターン認識と機械学習 — ベイズ理論による統計的予測 (Pattern Recognition and Machine Learning)*, 2006, Springer
第2章 確率分布 には, 応用上重要な marginal likelihood 積分の数多くの計算例が掲載されている.
- [2] 渡辺澄夫, *代数幾何と学習理論*, 2006, 森北出版.
特異点解消を用いた分配関数 (marginal likelihood 積分) の漸近展開法の解説が含まれている. この方法を利用して, 学習理論の汎化誤差の評価を行なっている.
- [3] 伊庭幸人, *マルコフ連鎖モンテカルロ法の基礎, 統計科学のフロンティア* 12, 計算統計 II, 2005, 岩波書店, 1–106.
Markov chain monte carlo (MCMC) 法の入門, MCMC を利用した分配関数の数値計算法など興味深い解説. 微分方程式で step by step に値を計算できる範囲を広げていく方法に似ている.

9月17日(木曜日)

(講義3) 10:00~12:00

「凸多面体とグレブナー基底」

大杉 英史(立教大学理学部・JST CREST)

日比 孝之(大阪大学大学院情報科学研究科・JST CREST)

[概要:凸多面体に付随するトーリックイデアルのグレブナー基底から構成される、多面体の三角形分割(正則三角形分割)、与えられたトーリックイデアルが持ち得るグレブナー基底を特徴付けるステイト多面体およびグレブナーファンについて解説する。]

(演習3) 14:00~

「凸多面体とグレブナー基底」

中山 洋将(神戸大学大学院理学研究科・JST CREST)

西山 絢太(神戸大学大学院理学研究科・JST CREST)

[概要:凸多面体の種々の基本的な構成を polymake を使って実習し, Macaulay2 や Risa/Asir を使って、正則三角形分割をグレブナー基底から構成する。また, TiGERS などを利用し、正則三角形分割の数え上げ計算を行うとともに, gfan や kan/sm1 を利用し Groebner fan / tropical variety の計算を行う。]

重み順序

$$S = K[x_1, \dots, x_n]$$

$$w = (w_1, \dots, w_n) \in \mathbb{Z}_{\geq 0}^n$$

$$(0 \neq) f = \sum_{\alpha} c_{\alpha} x^{\alpha} \in S \text{ に対して、}$$

係数

$$in_w(f) := \text{内積 } w \cdot \alpha \text{ が最大となる } \alpha \text{ に対する } c_{\alpha} x^{\alpha} \text{ の和}$$

$$I \subset S : \text{イデアル}$$

$$in_w(I) := \{ in_w(f) \mid 0 \neq f \in I \}$$

命題

$w \in \mathbb{Z}_{\geq 0}^n$, $< : S$ の単項式順序 に対して、

$$x^a <_w x^b \iff \begin{matrix} \text{det} \\ w \cdot a < w \cdot b \end{matrix}$$

or

$$w \cdot a = w \cdot b \text{ かつ } x^a < x^b$$

で定義される順序 $<_w$ は S の単項式順序

命題

任意の単項式順序 $<$ と S の任意のイデアル I に対して、

$w \in \mathbb{Z}_{\geq 0}^n$ が存在して、

$$in_{<}(I) = in_w(I)$$

を満たす。

定理

任意のイデアル I に対して

$$\{ in_{<}(I) \mid \text{単項式順序} \}$$

は有限集合。つまり、 I のイデアルは有限種類しかない。

① 各 $in_{<}(I)$ に対応するグロブナー基底 $g_{<}$ とすると、有限集合 $\cup g_{<}$ が得られる。

この集合は任意の単項式順序に対して、 I の GB

$\rightsquigarrow I$ の universal GB といふ。

$$\text{(例)} \quad I = (x_1^6 + x_1^5 x_2 + 3x_1^2 x_2^2 + x_1 x_2^3 + x_1 + x_2^2)$$

$$\{ in_{<}(I) \mid < : \text{単項式順序} \}$$

||

$$\{ (in_{<}(f)) \mid < : \text{ " } \}$$

例えば、任意の $<$ に関して

$$\begin{matrix} x_1 x_2^3 > x_1 \\ x_1 x_2^3 > x_2^2 \end{matrix} \iff in_{<}(f) \text{ にはならない}$$

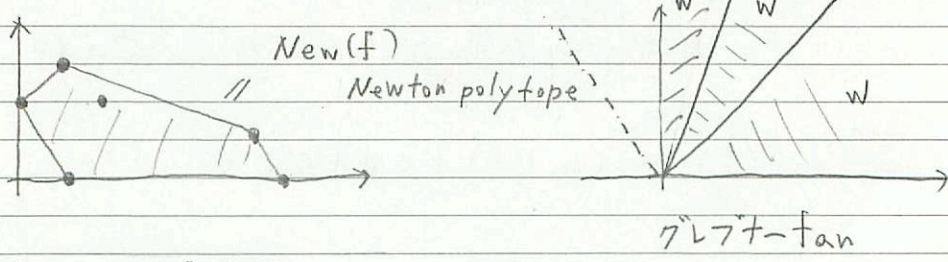
$$w = (1, 0) \Rightarrow \text{in}_w(f) = x_1^6$$

$$w' = (0, 1) \Rightarrow \text{in}_{w'}(f) = x_1 x_2^3$$

$$w'' = (2, 3) \Rightarrow \text{in}_{w''}(f) = x_1^5 x_2$$

$$f = x_1^6 + x_1^5 x_2 + 3x_1^2 x_2^2 + x_1 x_2^3 + x_1 + x_2^2$$

(6,0) (5,1) (2,2) (1,3) (1,0) (0,2)

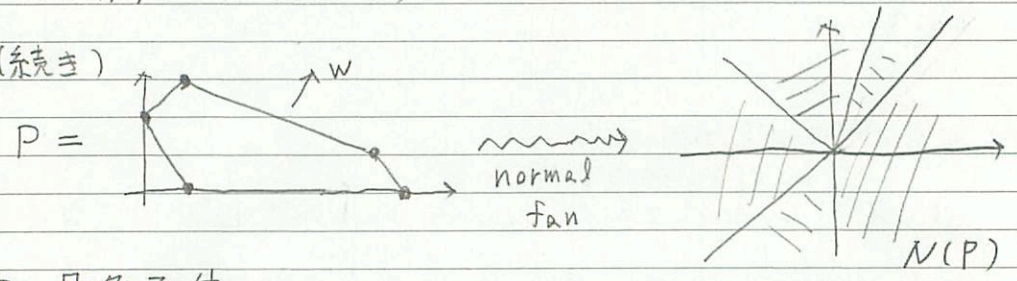


I のイニシャルイデアルは
 $(x_1^6), (x_1 x_2^3), (x_1^3 x_2)$
 の 3 通り。

一般には、 $w \in \mathbb{R}_{\geq 0}^n$, I : イデアル
 $C_I[w] := \{ w' \in \mathbb{R}_{\geq 0}^n \mid \text{in}_{w'}(I) = \text{in}_w(I) \}$
 w で決まる I のグロブナー cone と呼ぶ

$G_I(I) := \{ C_I[w] \mid w \in \mathbb{R}_{\geq 0}^n \}$
 I のグロブナー Fan とする

(例) (続き)



P : 凸多面体
 $\text{face}_w(P) := \{ u \in P \mid w \cdot u \geq w \cdot v \text{ for } \forall v \in P \}$

P の面 F に対して、
 $N_P(F) := \{ w \in \mathbb{R}^n \mid \text{face}_w(P) = F \}$
 F の normal cone
 $N(P) := P$ の normal cone の全体

命題

→ 斉次な多項式で生成される
 $I \subset S$ が斉次イデアルならば、任意の $w \in \mathbb{R}^n$ に対して、ある $w' \in \mathbb{R}_{\geq 0}^n$ が存在して
 $\text{in}_w(I) = \text{in}_{w'}(I)$
 が成り立つ。

定理 (state polytope)

任意の斉次イデアル $I \subset S$ に対して、多面体 $\text{State}(I) \subset \mathbb{R}^n$ が存在して

$$N(\text{State}(I)) = \text{GF}(I)$$

を満たすものが構成できる。

命題

U : 斉次イデアル I の universal GB

のとき、

$$\sum_{f \in U} \text{New}(f) \quad (= \text{State}(I) + \dots)$$

||

$$\text{New}(\prod_{f \in U} f)$$

$$\text{シンゴフスキ-和} : P + Q = \{p+q \mid p \in P, q \in Q\}$$

$$\diamond + 1 = \text{pentagon}$$

イデアルイデアルは一語

ト-リックイデアル

$A = (a_1, \dots, a_n)$ $d \times n$ 行列

$$\left(\begin{array}{l} \text{rank } A = d \text{ i.e. full rank} \\ \exists \alpha \in \mathbb{R}^d \text{ s.t. } \alpha \cdot a_i = 1 \quad \forall i \end{array} \right) \quad \text{± 仮定}$$

$$\text{(例)} \quad A = \begin{pmatrix} 1 & 1 & \dots & 1 \\ \boxed{} \end{pmatrix}$$

$S = k[x_1, \dots, x_n]$

$$I_A = (x^u - x^v \in S \mid u, v \in \mathbb{Z}_{\geq 0}^n, Au = Av)$$

A の ト-リックイデアル

(例)

$$A = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & -1 \end{pmatrix}$$

$$I_A = (x_1 x_2 - x_3 x_5, x_1 x_4 - x_2 x_3, x_2^2 - x_4 x_5)$$

$\overset{g_1}{}, \quad \overset{g_2}{}, \quad \overset{g_3}{}$

$\langle \text{逆辞書式 } (x_1, \dots, x_5) \rangle$

$\{g_1, g_2, g_3, x_1^2 x_4 - x_3^2 x_5\}$ は I_A の $\langle \cdot \rangle$ に関する GB

$$\text{inc}_1(I) = (x_1 x_2, x_2 x_3, x_2^2, x_1^2 x_4)$$

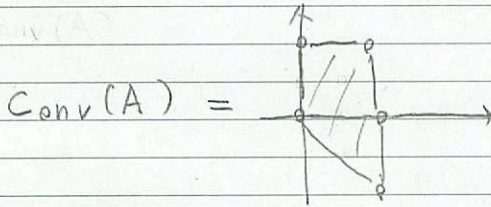
$$\sqrt{\text{inc}_1(I)} = (x_1 x_4, x_2)$$

\langle_2 : 逆辞書式 ($x_5 \succ \dots \succ x_1$)

$\{g_1, g_2, g_3\}$ は $IA \cap \langle_2$ に關する GB

$$in_{\langle_2}(IA) = (x_3 x_5, x_2 x_3, x_4 x_5)$$

$$\parallel \sqrt{in_{\langle_2}(IA)}$$



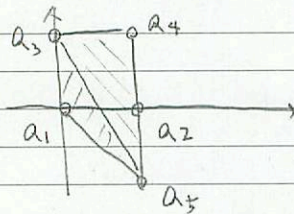
◦ \langle : 単項式順序

$$\Delta(in_{\langle}(IA)) = \{ \text{conv}(B) \mid B \subset A, \prod_{a_i \in B} x_i \notin \sqrt{in_{\langle}(IA)} \}$$

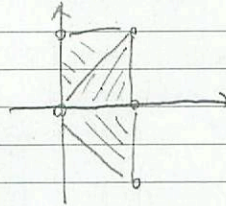
= シェル複体

(例) (続き)

$\sqrt{in_{\langle_1}(IA)} = (x_1, x_4, x_2)$
 に属さないスグエ了711- → $x_1 x_5, x_3 x_4, x_3 x_5, x_4 x_5$
 単項式付 $x_1 x_3 x_5, x_3 x_4 x_5$



$\Delta(in_{\langle_1}(IA))$
not unimodular



$\Delta(in_{\langle_2}(IA))$
unimodular

定理

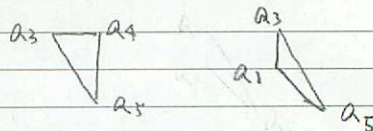
$\Delta(in_{\langle}(IA))$ は Conv(A) の 三角形分割 である。
 (単体分割)

命題

$$\sqrt{in_{\langle}(IA)} = \left(\prod_{a_i \in B} x_i \mid B \subset A, \text{Conv}(B) \notin \Delta(in_{\langle}(IA)) \right)$$

minimal な B を考えれば OK

$\xrightarrow{B \subset A}$
 $\text{Conv}(B) \in \Delta(in_{\langle}(IA))$
 maximal な
 ものを考えれば OK



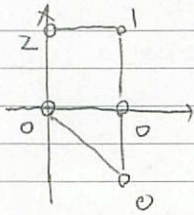
(例) (続き)

$$\sqrt{in_{\langle_1}(IA)} = (x_1, x_4, x_2) = (x_1, x_2) \cap (x_4, x_2)$$

定理

$$\Delta(\text{in}_<(IA)) \text{ が unimodular} \iff \sqrt{\text{in}_<(IA)} = \text{in}_<(IA)$$

例 (続き)

 $w = (0, 0, 2, 1, 0)$ に対応する正則三角形分割 Δ_w


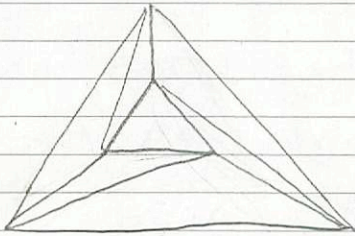
重みの方だけ
引く



$$\text{in}_w(IA) = \text{in}_<(IA)$$

$$\Rightarrow \Delta_w = \Delta(\text{in}_<(IA))$$

例 非正則三角形分割



◦ IA の universal GB

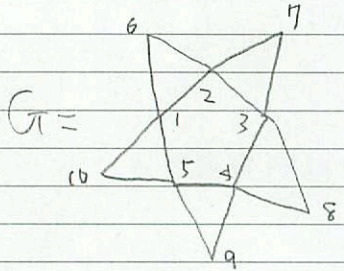
A に対して,

$$\Delta(A) = \begin{pmatrix} A & 0 \\ E_n & E_n \end{pmatrix} \quad (d+n) \times 2n \text{ 行列}$$

A の Lawrence lifting

$$IA(A) \text{ の生成系} \xrightarrow{y_i=1} IA \text{ の universal GB} \\ \text{(Giroirek basis)}$$

大杉多面体



有限グラフ:
頂点 10個
辺 15個

$P = \{ \sum_{j=1}^m \alpha_j e_j \mid \alpha_j \geq 0 \}$ $e_i + e_j \in \mathbb{R}^{10}$ を考える

(13) $\begin{matrix} 2 \\ \swarrow \searrow \\ 0 \quad 3 \end{matrix} \sim (0, 1, 1, 0, 0, 0, 0, 0, 0, 0)$

$P_G = \{ P(e) \mid e \text{ は辺} \}$ の凸閉包: 大杉多面体, 呼ぶ

単体の個数が最大

P_G は ① unimodular triangulation は存在し、これは全て nonregular

450
111頁

② 単体の個数が最小のものも、すべて nonregular

↑ PANTOS というソフトを使って分かった

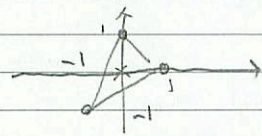
FANO 多面体

頂点が全て整数点

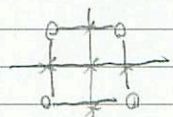
$P \subset \mathbb{R}^d$: 整凸多面体

① Fano 原点

② terminal



③ canonical (terminalでない場合)



④ Gorenstein

P^* が整凸多面体

↑ dual polytope

$P^* = \{ x \in \mathbb{R}^d \mid \langle x, y \rangle \leq 1 \ \forall y \in P \}$

⑤ \mathbb{Q} -factorial

Fano polytope P が単体的

⑥ Smooth



maximal face
(facet)

facet n 頂点が \mathbb{Z}^d の \mathbb{Z} -basis

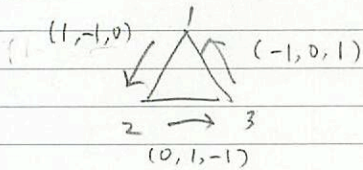
問題の訂正

$$P(\mathbb{G}; w) \subset \mathbb{H} \cong \mathbb{R}^{d-1}$$

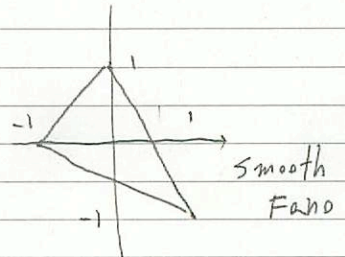
$$\hookrightarrow x_1 + \dots + x_d = 0$$

(a) d 次元 \rightarrow $d-1$ 次元
 \times \circ

(例) $d=3$



$$\begin{pmatrix} (1, -1, 0) \\ (0, 1, -1) \\ (-1, 0, 0) \end{pmatrix} \in \mathbb{R}^3$$



NO

DATE



FANO 凸多面体

整凸多面体 (integral convex polytope) とは、頂点の座標が整数点 (すべての座標が整数) であるものを言う。

- Fano 凸多面体というのは、 \mathbf{R}^d の d 次元整凸多面体であり、その内部に含まれる整数点が原点のもののみのものである。
- terminal Fano 凸多面体とは、境界 (boundary) に含まれる整数点がすべて頂点である Fano 凸多面体のことである。
- canonical Fano 凸多面体とは、境界に頂点以外の整数点を持つ Fano 凸多面体のことである。
- Fano 凸多面体 $\mathcal{P} \subset \mathbf{R}^d$ が Gorenstein Fano 凸多面体であるとは、その双対凸多面体 (dual polytope)

$$\mathcal{P}^* = \{x \in \mathbf{R}^d : \langle x, y \rangle \leq 1, y \in \mathcal{P}\}$$

が整凸多面体となる Fano 凸多面体のことである。但し、 $\langle x, y \rangle$ は \mathbf{R}^d の通常の内積である。

- Q -factorial Fano 凸多面体とは単体的な (すなわち、すべての面が単体である) Fano 凸多面体のことである。
- smooth (非特異) Fano 凸多面体とは、それぞれの facet の頂点が \mathbf{Z}^d の \mathbf{Z} 基底となっている Fano 凸多面体のことである。(すると、smooth ならば、 Q -factorial & Gorenstein である。)

問題 頂点集合 $[d] = \{1, 2, \dots, d\}$ 上の有限グラフ G がある。但し、 G はループも重複辺も含まないものとする。有限グラフ G の orientation とは、 G のそれぞれの辺に矢印を付けることである。矢印が付いた辺を directed edge と呼ぶ。有限グラフ G とその orientation ω の組 $(G; \omega)$ を directed graph と呼び、その directed edge の全体を $E(G; \omega)$ と表す。いま、directed graph $(G; \omega)$ の directed edge $i \rightarrow j$ に \mathbf{R}^d の点 $\mathbf{e}_i - \mathbf{e}_j$ を対応させ、 $\mathcal{P}_{(G; \omega)}$ を $\{\mathbf{e}_i - \mathbf{e}_j : i \rightarrow j \in E(G; \omega)\}$ の凸閉包とする。但し、 $\mathbf{e}_1, \dots, \mathbf{e}_d$ は \mathbf{R}^d の標準的な単位座標ベクトルである。整凸多面体 $\mathcal{P}_{(G; \omega)}$ は方程式 $x_1 + \dots + x_n = 1$ で定義される \mathbf{R}^d の超平面 \mathcal{H} に含まれるから、 \mathcal{H} と \mathbf{R}^{d-1} を同一視することによって、 $\mathcal{P}_{(G; \omega)} \subset \mathbf{R}^{d-1}$ と思う。

たとえば、 G を頂点集合 $[3] = \{1, 2, 3\}$ 上の有限グラフで、辺 $\{1, 2\}, \{2, 3\}, \{1, 3\}$ を持つとし、orientation ω を $1 \rightarrow 2, 2 \rightarrow 3, 3 \rightarrow 1$ とすると、 $\mathcal{P}_{(G; \omega)}$ の頂点は $(1, -1, 0), (0, 1, -1), (-1, 0, 1)$ となる。すると、 $\mathcal{P}_{(G; \omega)} \subset \mathbf{R}^2$ と思うと、その頂点は $(1, -1), (0, 1), (-1, 0)$ となるから、 $\mathcal{P}_{(G; \omega)}$ は 2 次元の Fano 凸多面体である。

- $\mathcal{P}_{(G; \omega)}$ が $d-1$ 次元となるための必要十分条件を求めよ
- $\mathcal{P}_{(G; \omega)}$ が Fano 凸多面体となるための必要十分条件を求めよ。
- $\mathcal{P}_{(G; \omega)}$ が Fano 凸多面体ならば、 $\mathcal{P}_{(G; \omega)}$ は Gorenstein であることを示せ。
- $\mathcal{P}_{(G; \omega)}$ が Q -Fano 凸多面体となるための必要十分条件を求めよ。
- $\mathcal{P}_{(G; \omega)}$ が smooth Fano 凸多面体となるための必要十分条件を求めよ。

解答 (a) と (b) は既知。(c) は totally unimodular matrix の理論。(d) と (e) は未解決問題。一般の G では難しいから、(d) と (e) は完全グラフのときにやっても十分に面白い。

参考文献 arXiv:0704.0049 arXiv:0805.4533 arXiv:0806.2604

凸多面体とグレブナ基底, 問題集. 2009.09.17

1.

$$A = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 2 \end{pmatrix}$$

について、

- (a) トーリックイデアル I_A を求めよ。
- (b) $w = (2, 1, 2)$ について $\text{in}_w(I_A)$ を求めよ。
- (c) $w = (2, 1, 2)$ について regular triangulation を求めよ。
- (d) $w = (1, 2, 1)$ について regular triangulation を求めよ。
- (e) 上の regular triangulation を $\sqrt{\text{in}_w(I_A)}$ の準素分解から計算せよ。

2.

$$A = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{pmatrix}$$

について、

- (a) トーリックイデアル I_A を求めよ。
- (b) $w = (1, 0, 0, 2)$ について $\text{in}_w(I_A)$ を求めよ。
- (c) 上の w について regular triangulation を求めよ。

3.

$$A = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 3 & 4 \end{pmatrix}$$

について、

- (a) トーリックイデアル I_A を求めよ。
- (b) $w = (1, 0, 0, 2)$ について regular triangulation を求めよ。

4. $P = \{(x, y) \in \mathbf{R}^2 \mid x \geq 0, y \geq 0\}$ とする。

- (a) 次の $w_i \in \mathbf{R}^2$ について, $F_i := \text{face}_{w_i}(P)$ を計算せよ。

$$w_1 = (1, 1), w_2 = (-1, -1), w_3 = (-1, 0), w_4 = (0, -1), w_5 = (0, 0)$$

- (b) F_i の normal cone $N_P(F_i)$ を求めよ。

5. $I = \langle x^2 + y^2 - 4, xy - 1 \rangle$ とする。

- (a) $y \prec x$ なる lexicographic order \prec に対して, $C_{I, \prec}$ (\prec できる I の Gröbner cone) を求めよ。

- (b) $C_I[(3,1)] = \{w \in \mathbf{R}_{\geq 0}^2 \mid \text{in}_w(I) = \text{in}_{(3,1)}(I)\}$ を計算せよ。
 (c) I の $\mathbf{R}_{\geq 0}^2$ における Gröbner fan 計算し, 図示せよ。
 (d) 各 cone におけるグレブナ基底を計算し, $\text{in}_w(I)$ を求めよ。

6. $J = \langle x + y + z, y + 2z \rangle$ とする。

- (a) J の Gröbner fan を計算せよ。
 (b) (a) において z の重みを 0 に制限して得られる fan を考える。この fan を normal fan として持つような polytope を与えよ。
 (c) J の Gröbner fan を normal fan として持つような polytope (i.e. state polytope) を計算せよ。

7. $\Delta_2 \times \Delta_2$ つまり

$$A = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

を考える。

- (a) A の triangulation をひとつ, グレブナ基底を求めることにより求めよ。
 (b) I_A の Gröbner fan を求めよ。
8. (a) 次の凸多面体について原点を内点として含むかどうかを polymake などを使って判定せよ。

$$P_1 = \text{conv}(\{(1, 0, 0), (0, 1, 0), (0, 0, 1), (-1, -1, -1)\})$$

$$P_2 = \text{conv}(\{(-1, 0, 0), (0, 1, 0), (0, 0, 1), (1, -1, 1)\})$$

- (b) 次の凸多面体について双対凸多面体を polymake などを用いて計算せよ。

$$P_1 = \text{conv}(\{(1, 0), (0, 1), (-1, -1)\})$$

$$P_2 = \text{conv}(\{(1, 0, 0), (0, 1, 0), (0, 0, 1), (1, 1, 1)\})$$

$$P_3 = \text{conv}(\{(1, -1, -1), (-1, -1, 0), (0, 1, -1), (0, 0, 1)\})$$

$$P_4 = \text{conv}(\{(1, 0, 0), (0, 1, 0), (0, 0, 1), (-1, 0, 0), (0, -1, 0), (0, 0, -1)\})$$

9. (研究課題) 与えられた \mathbb{Z}^d の点集合で定まる凸多面体 P に対してその双対多面体の頂点を計算するプログラムを作成せよ. P が smooth fano 多面体かどうか判定するプログラムを作成せよ. (P が原点を内点にもつか?, P の facet の頂点が \mathbb{Z}^d の基底か判定する.)

参考文献

1.

$$A = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 2 \end{pmatrix}$$

について、

(a) トーリックイデアル I_A を求めよ。

$$I_A = \langle x_0x_2 - x_1^2 \rangle$$

Asir のプログラム `toric.rr` を使って、トーリックイデアルの計算を行うには次のようにする。

```
Asir
[1237] load("toric.rr");
[1250] toric_ideal([[1,1,1],[0,1,2]]);
ideal :
[x2-t0*t1^2,x1-t0*t1,x0-t0,t0*t1*t2-1]
gb :
[x0*x2-x1^2,-t2*x1+1,t2*x0*x2-x1,-t2*x2+t1,x0-t0]
[x0*x2-x1^2]
```

参考:

Asir のプログラム `toric.rr`, `4ti2`, Macaulay2 のプログラム `toric.m2`

(b) $w = (2, 1, 2)$ について $\text{in}_w(I_A)$ を求めよ。

$$\text{in}_w(I_A) = \langle x_0x_2 \rangle$$

(c) $w = (2, 1, 2)$ について regular triangulation を求めよ。

$$I_{\Delta_w} = \sqrt{\text{in}_w(I_A)} = \langle x_0x_2 \rangle = \langle x_0 \rangle \cap \langle x_2 \rangle$$

regular triangulation Δ_w の maximal face は $12, 01$ 。

$$\Delta_w = \{01, 12, 0, 1, 2\}$$

もしくは、重みについて持ち上げた点たちの下側凸包から求める方法

(d) $w = (1, 2, 1)$ について regular triangulation を求めよ。

$$\text{in}_w(I_A) = \langle x_1^2 \rangle$$

$$I_{\Delta_w} = \sqrt{\text{in}_w(I_A)} = \langle x_1 \rangle$$

regular triangulation Δ_w の maximal face は 02 。

$$\Delta_w = \{02, 0, 2\}$$

もしくは、重みについて持ち上げた点たちの下側凸包から求める方法

2.

$$A = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{pmatrix}$$

について、

(a) トーリックイデアル I_A を求めよ。

$$I_A = \langle x_0x_3 - x_1x_2 \rangle$$

(b) $w = (1, 0, 0, 2)$ について $\text{in}_w(I_A)$ を求めよ。

$$\text{in}_w(I_A) = \langle x_0x_3 \rangle$$

(c) 上の w について regular triangulation を求めよ。

$$I_{\Delta_w} = \sqrt{\text{in}_w(I_A)} = \langle x_0x_3 \rangle = \langle x_0 \rangle \cap \langle x_3 \rangle$$

regular triangulation Δ_w の maximal face 123, 012

$$\Delta_w = \{123, 012, 12, 13, 31, 01, 02, 0, 1, 2, 3\}$$

もしくは、重みについて持ち上げた点たちの下側凸包から求める方法

3.

$$A = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 3 & 4 \end{pmatrix}$$

について、

(a) トーリックイデアル I_A を求めよ。

$$I_A = \langle x_0x_3 - x_1x_2, -x_1x_3^2 + x_2^3, -x_1^2x_3 + x_0x_2^2, -x_0^2x_2 + x_1^3 \rangle$$

(b) $w = (1, 0, 0, 2)$ について regular triangulation を求めよ。

$$\text{in}_w(I_A) = \langle -x_1^2x_3, x_0x_3, x_1x_3^2, x_0^2x_2 \rangle$$

Asir のプログラム toric.rr 中の gr_w, in_w を使って、上の計算を行うには次のようにする。

```
Asir
[1220] load("toric.rr");
[1233] Id=toric_ideal([[1,1,1,1],[0,1,3,4]]);
ideal :
[x3-t0*t1^4,x2-t0*t1^3,x1-t0*t1,x0-t0,t0*t1*t2-1]
gb :
[x0*x3-x1*x2,-x1*x3^2+x2^3,-x1^2*x3+x0*x2^2,-x0^2*x2+x1^3,
-t2*x1+1,t2*x0*x3-x2,-t2*x0^2*x2+x1^2,t1*x3-t2*x2^2,-x3+t1*x2,
t2*x0*x2-t1*x1,-x1+t1*x0,x0-t0,t2^2*x0*x2-t1,-t2*x2+t1^2]
[x0*x3-x1*x2,-x1*x3^2+x2^3,-x1^2*x3+x0*x2^2,-x0^2*x2+x1^3]
[1234] G=gr_w(Id,[x0,x1,x2,x3],[1,0,0,2]);
[-x1^2*x3+x0*x2^2,-x0^2*x2+x1^3,x0*x3-x1*x2,-x1*x3^2+x2^3]
[1235] InW=map(in_w,G,[x0,x1,x2,x3],[1,0,0,2]);
[-x1^2*x3,-x0^2*x2,x0*x3,-x1*x3^2]
[1236] primedec(InW,[x0,x1,x2,x3]);
[[x1,x0],[x3,x0],[x3,x2]]
```

$$I_{\Delta_w} = \sqrt{\text{in}_w(I_A)} = \langle x_1x_3, x_0x_3, x_0x_2 \rangle = \langle x_0, x_1 \rangle \cap \langle x_0, x_3 \rangle \cap \langle x_2, x_3 \rangle$$

regular triangulation Δ_w の maximal face が 23, 12, 01。

$$\Delta_w = \{23, 12, 01, 0, 1, 2\}$$

もしくは、重みについて持ち上げた点たちの下側凸包から求める方法

4. $P = \{(x, y) \in \mathbb{R}^2 \mid x \geq 0, y \geq 0\}$ とする。

(a) 次の $w_i \in \mathbb{R}^2$ について、 $F_i := \text{face}_{w_i}(P)$ を計算せよ。

$$w_1 = (1, 1), w_2 = (-1, -1), w_3 = (-1, 0), w_4 = (0, -1), w_5 = (0, 0)$$

- $F_1 := \text{face}_{w_1}(P) = \emptyset$
- $F_2 := \text{face}_{w_2}(P) = \{(0, 0)\}$
- $F_3 := \text{face}_{w_3}(P) = (0, 1) \times \mathbf{R}_{\geq 0}$
- $F_4 := \text{face}_{w_4}(P) = (1, 0) \times \mathbf{R}_{\geq 0}$
- $F_5 := \text{face}_{w_5}(P) = P$

(b) F_i の normal cone $N_P(F_i)$ を求めよ.

- $N_P(F_1) = \{(w_1, w_2) \mid w_1 > 0 \text{ or } w_2 > 0\}$
- $N_P(F_2) = \{(w_1, w_2) \mid w_1 < 0, w_2 < 0\}$
- $N_P(F_3) = \{(w_1, 0) \mid w_1 < 0\}$
- $N_P(F_4) = \{(0, w_2) \mid w_2 < 0\}$
- $N_P(F_5) = \{(0, 0)\}$

5. $I = \langle x^2 + y^2 - 4, xy - 1 \rangle$ とする.

(a) $y < x$ なる lexicographic order $<$ に対して, $C_{I, <}$ ($<$ できまる I の Gröbner cone) を求めよ.

I の $<$ に関する簡約グレブナ基底は $\{y^4 - 4y^2 + 1, x + y^3 - 4y\}$ であるので, $w = (w_1, w_2)$ に関する次の不等式系の解を計算する.

$$\begin{aligned} (0, 2) \cdot w > 0, & \quad (0, 4) \cdot w > 0, \\ (1, -3) \cdot w > 0, & \quad (1, -1) \cdot w > 0, \\ (1, 0) \cdot w > 0, & \quad (0, 1) \cdot w > 0 \end{aligned}$$

故に, $C_{I, <} = \{(w_1, w_2) \mid w_2 > 0, w_2 < 1/3w_1\}$.

polymake で計算するには以下のファイル

```
cone1.txt
INEQUALITIES
0 0 2
0 0 4
0 1 -3
0 1 -1
0 1 0
0 0 1
```

を用意して, シェルから次のコマンドを実行する.

```
$ polymake cone1.txt FACETS
FACETS
0 0 2
0 1 -3
1 0 0
```

(b) $C_I[3, 1] = \{w \in \mathbf{R}_{\geq 0}^2 \mid \text{in}_w(I) = \text{in}_{(3,1)}(I)\}$ を計算せよ.

I の $<_{(3,1)}$ に関する簡約グレブナ基底は $\{y^4 - 4y^2 + 1, x + y^3 - 4y\}$ であるので, $w = (w_1, w_2)$ に関する次の方程式系の解を計算する.

$$\begin{aligned} (1, -3) \cdot w = 0, & \quad (1, -1) \cdot w > 0, \\ (0, 2) \cdot w > 0, & \quad (0, 4) \cdot w > 0, \\ (1, 0) \cdot w > 0, & \quad (0, 1) \cdot w > 0 \end{aligned}$$

故に, $C_I[(3, 1)] = \{(w_1, w_2) \mid w_2 > 0, w_2 = 1/3w_1\}$.

polymake で計算するには以下のファイル

```

cone2.txt
INEQUALITIES
0 1 -3
0 -1 3
0 1 -1
0 0 2
0 0 4
0 1 0
0 0 1

```

を用意して, シェルから次のコマンドを実行する.

```

$ polymake cone2.txt FACETS AFFINE_HULL
FACETS
0 1 -1
1 0 0

AFFINE_HULL
0 1 -3

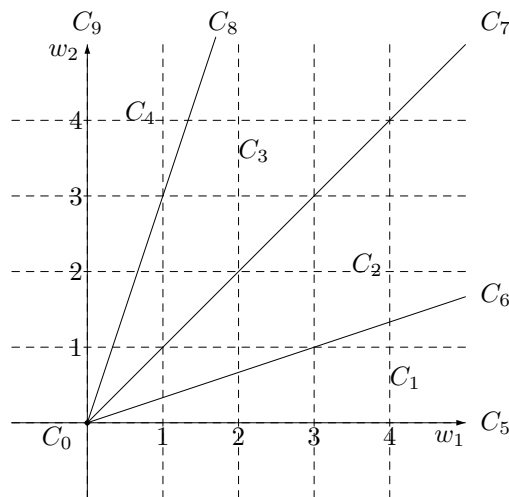
```

(c) I の $\mathbb{R}_{\geq 0}^2$ における Gröbner fan 計算し, 図示せよ.

$\overline{C_{I, \prec}}$ に属さないが十分近い w を一つとる. 今, (b) より, $(3, 1)$ が境界であることが分かっているので, $w = (3, 2)$ を取ってみる. このとき, C_{I, \prec_w} を計算すると,

$$C_{I, \prec_w} = \{(w_1, w_2) \mid w_2 < w_1, w_2 > 1/3w_1\}$$

であり, $C_{I, \prec}$ と境界を共有している. (境界を共有しなかった場合はもっと近くの w を取り直す.) これを $\mathbb{R}_{\geq 0}^2$ を尽くすまで繰り返す. 今は x と y の対称性から残りの部分はすぐに分かる. 図示すると以下のようなになる.



(d) 各 cone におけるグレブナ基底を計算し, $\text{in}_w(I)$ を求めよ.

各 cone に属する w を用いて \prec_w に関するグレブナ基底を計算した. 下線部が先頭項である.

- $C_1 : \{\underline{x} + y^3 - 4y, \underline{y^4} - 4y^2 + 1\}$
- $C_2 : \{\underline{x^2} + y^2 - 4, \underline{xy} - 1, \underline{y^3} + x - 4y\}$

- $C_3 : \{\underline{y^2 + x^2 - 4}, \underline{xy - 1}, \underline{x^3 - 4x + y}\}$
- $C_4 : \{\underline{x^4 - 4x^2 + 1}, \underline{y + x^3 - 4x}\}$
- $C_5 : \{\underline{x + y^3 - 4y}, \underline{y^4 - 4y^2 + 1}\}$
- $C_6 : \{\underline{x + y^3 - 4y}, \underline{y^4 - 4y^2 + 1}\}$
- $C_7 : \{\underline{x^2 + y^2 - 4}, \underline{xy - 1}, \underline{y^3 + x - 4y}\}$
- $C_8 : \{\underline{y^2 + x^2 - 4}, \underline{xy - 1}, \underline{x^3 + y - 4x}\}$
- $C_9 : \{\underline{x^4 - 4x^2 + 1}, \underline{y + x^3 - 4x}\}$
- $C_0 : \{\underline{x + y^3 - 4y}, \underline{y^4 - 4y^2 + 1}\}$

6. $J = \langle x + y + z, y + 2z \rangle$ とする.

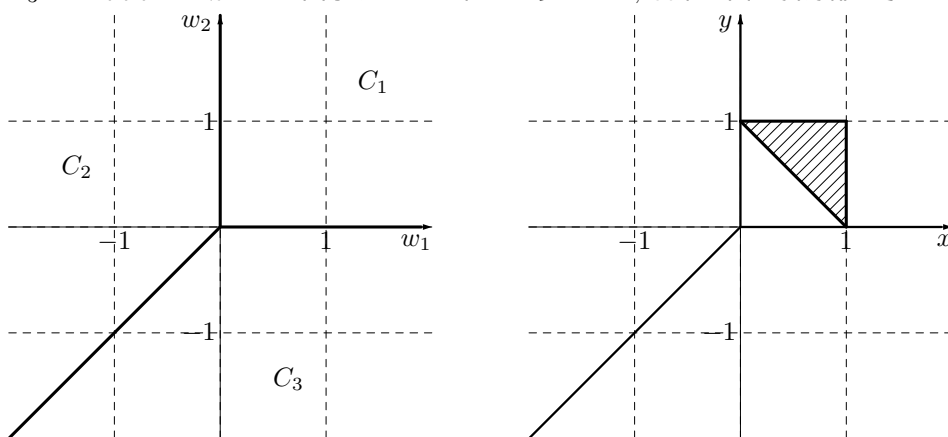
(a) J の Gröbner fan を計算せよ.

$$\begin{aligned} C_1 &= \{(w_1, w_2, w_3) \mid w_1 > w_3, w_2 > w_3\} \\ C_2 &= \{(w_1, w_2, w_3) \mid w_1 < w_2, w_1 < w_3\} \\ C_3 &= \{(w_1, w_2, w_3) \mid w_1 > w_2, w_2 < w_3\} \end{aligned}$$

の3つを最大次元の cone とする fan が J の Gröbner fan である.

(b) (a) において z の重みを 0 に制限して得られる fan を考える. この fan を normal fan として持つような polytope を与えよ.

$w_3 = 0$ 平面での切り口を図示すると左下のようになり, 右下の図の斜線部が求める polytope である.

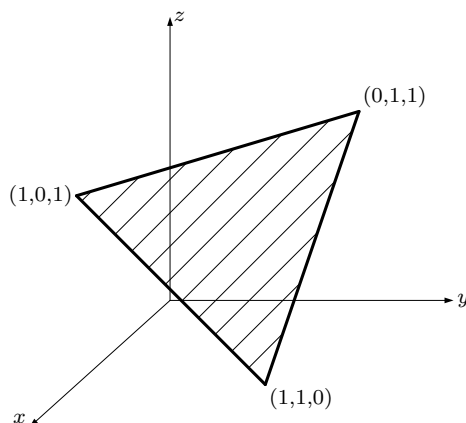


(c) J の Gröbner fan を normal fan として持つような polytope (i.e. state polytope) を計算せよ.

今, J は線型多項式によって生成されているので,

$$\begin{aligned} \text{State}(J) &= \text{Mat}(J) \\ &= \text{conv}\{(1, 1, 0), (0, 1, 1), (1, 0, 1)\} \end{aligned}$$

である. 図示すると以下ようになる.



7. $\Delta_2 \times \Delta_2$ つまり

$$A = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

を考える.

- (a) A の triangulation をひとつ, グレブナ基底を求めることにより求めよ.

I_A の生成元は

$[x_4*x_8-x_5*x_7, x_1*x_8-x_2*x_7, x_3*x_8-x_5*x_6, x_3*x_7-x_4*x_6, x_0*x_8-x_2*x_6, x_0*x_7-x_1*x_6, x_1*x_5-x_2*x_4, x_0*x_5-x_2*x_3, x_0*x_4-x_1*x_3]$

たとえば, $w = (1, 0, 0, 0, 1, 0, 0, 3, 1)$ として, $\text{in}_w(I_A)$ を計算すると,

$[x_3*x_8, x_0*x_5, -x_2*x_4, x_0*x_8, x_0*x_4, -x_5*x_7, x_3*x_7, -x_2*x_7, x_0*x_7]$

となり, これの radical を 準素分解した結果は

$\{\text{ideal}(x_8, x_7, x_5, x_4), \text{ideal}(x_5, x_3, x_2, x_0), \text{ideal}(x_7, x_3, x_2, x_0), \text{ideal}(x_8, x_7, x_2, x_0), \text{ideal}(x_8, x_7, x_4, x_0), \text{ideal}(x_7, x_4, x_3, x_0)\}$

となって, regular triangulation Δ_w の maximal face は

01236, 14678, 14568, 13456, 12356, 12568

となる.

- (b) I_A の Gröbner fan を求めよ.

gfan を用いて Gröbner fan を計算させる。上で計算したイデアルから次のような入力ファイルを作る。(x0, ..., x8 を a, b, ..., i に置き換える)

```
gfan_d2xd2.txt
{
  i*e-h*f, i*b-h*c, i*d-g*f, h*d-g*e, i*a-g*c,
  h*a-g*b, f*b-e*c, f*a-d*c, e*a-d*b
}
```

gfan コマンドを実行し, 出力を gfan_d2xd2.out に出す。

```
$ gfan < gfan_d2xd2.txt > gfan_d2xd2.out
```

全ての reduced Gröbner basis を出力してくれる。その reduced Gröbner basis から Gröbner cone を計算するには, gfan_groebnercone を用いる。例として, gfan_d2xd2.out から reduced Gröbner basis $\{e*i-f*h, d*i-f*g, d*h-e*g, b*i-c*h, b*f-c*e, a*i-c*g, a*h-b*g, a*f-c*d, a*e-b*d\}$ を取り出し, これについての Gröbner cone を計算する。まず, 入力ファイル d2xd2_1.gb を用意。

```
d2xd2_1.gb
{
  e*i-f*h, d*i-f*g, d*h-e*g, b*i-c*h, b*f-c*e, a*i-c*g,
  a*h-b*g, a*f-c*d, a*e-b*d
}
```

gfan_groebnercone を次のように実行する。

```

$ gfan_groebnercone < d2xd2_1.gb
LP algorithm being used: "cddgmp".
_application PolyhedralCone
_version 2.2
_type PolyhedralCone

AMBIENT_DIM
9

DIM
9

IMPLIED_EQUATIONS

LINEALITY_DIM
5

LINEALITY_SPACE
1 0 0 0 -1 -1 0 -1 -1
0 1 0 0 1 0 0 1 0
0 0 1 0 0 1 0 0 1
0 0 0 1 1 1 0 0 0
0 0 0 0 0 0 1 1 1

FACETS
0 0 0 0 1 -1 0 -1 1
0 0 0 1 -1 0 -1 1 0
0 1 -1 0 -1 1 0 0 0
1 -1 0 -1 1 0 0 0 0

RELATIVE_INTERIOR_POINT
4 2 0 2 1 0 0 0 0

```

最後に出力される 4 つのベクトルが、cone を定義する各半空間の法線ベクトルを表している。すなわち、この cone を定義する線形不等式系は

$$w_5 - w_6 - w_8 + w_9 > 0$$

$$w_4 - w_5 - w_7 + w_8 > 0$$

$$w_2 - w_3 - w_5 + w_6 > 0$$

$$w_1 - w_2 - w_4 + w_5 > 0$$

9. (a) 次の凸多面体について原点を内点として含むかどうかを polymake などを使って判定せよ。

$$P_1 = \text{conv}(\{(1, 0, 0), (0, 1, 0), (0, 0, 1), (-1, -1, -1)\})$$

$$P_2 = \text{conv}(\{(-1, 0, 0), (0, 1, 0), (0, 0, 1), (1, -1, 1)\})$$

P_1 を入力とするため次のようなファイルを用意。

```

ip-p1.txt
POINTS
1 1 0 0
1 0 1 0
1 0 0 1
1 -1 -1 -1

```

P_1 の facet を計算するため、次のようなコマンドを入力

```

$ polymake ip_p1.txt FACETS
FACETS
1 -1 -1 -1
1 -1 -1 3
1 -1 3 -1
1 3 -1 -1

```

だから P_1 を定義する線形不等式系は

$$\begin{aligned}
1 - x_1 - x_2 - x_3 &\geq 0 \\
1 - x_1 - x_2 + 3x_3 &\geq 0 \\
1 - x_1 + 3x_2 - x_3 &\geq 0 \\
1 + 3x_1 - x_2 - x_3 &\geq 0
\end{aligned}$$

$(x_1, x_2, x_3) = (0, 0, 0)$ を代入すれば、いずれも $1 > 0$ となり、原点が内点とわかる。結局、facet の計算結果の第 1 成分の部分が正であれば、原点が内点であることがわかる。

P_2 についても同様の計算を行えば、

```

$ polymake ip_p2.txt FACETS
FACETS
1 -1 -1 -1
-1 -1 1 3
1 1 1 -1
1 1 -1 -1

```

となり、原点は内点でないことがわかる。

(b) 次の凸多面体について双対凸多面体を polymake などを用いて計算せよ。

$$\begin{aligned}
P_1 &= \text{conv}(\{(1, 0), (0, 1), (-1, -1)\}) \\
P_2 &= \text{conv}(\{(1, 0, 0), (0, 1, 0), (0, 0, 1), (1, 1, 1)\}) \\
P_3 &= \text{conv}(\{(1, -1, -1), (-1, -1, 0), (0, 1, -1), (0, 0, 1)\}) \\
P_4 &= \text{conv}(\{(1, 0, 0), (0, 1, 0), (0, 0, 1), (-1, 0, 0), (0, -1, 0), (0, 0, -1)\})
\end{aligned}$$

凸多面体 $P = \text{conv}(\{u_1, \dots, u_m\})$ について、その双対凸多面体は

$$P^* = \{v \mid \langle v, u \rangle \geq -1 \ (\forall u \in P)\}$$

である。任意の $u \in P$ について、ある非負実数 t_i で $\sum_{i=1}^m t_i = 1$ を満たすものが存在して、

$$u = \sum_{i=1}^m t_i u_i$$

とかけるから、

$$\langle v, u \rangle \geq -1 \ (\forall u \in P)$$

の条件は、

$$\langle v, u_i \rangle \geq -1 \ (1 \leq i \leq m) \quad \dots\dots\dots \textcircled{1}$$

と同値になる。この線形不等式系により定まる凸多面体が双対凸多面体である。

P_1 の場合、上の線形不等式系にあたるものは

$$\begin{aligned}1 + x_1 &\geq 0 \\1 + x_2 &\geq 0 \\1 - x_1 - x_2 &\geq 0\end{aligned}$$

であるから、次のような入力を用意する。

```
dual_p1.txt
INEQUALITIES
1 1 0
1 0 1
1 -1 -1
```

この入力は P_1 の頂点をそのまま書いたものに等しい。この線形不等式系の定義する凸多面体の頂点を計算したいので、

```
$ polymake dual_p1.txt VERTICES
VERTICES
1 2 -1
1 -1 -1
1 -1 2
```

だから、 P_1 の双対凸多面体は

$$P_1^* = \text{conv}(\{(2, -1), (-1, -1), (-1, 2)\})$$

同様の計算を P_2, P_3, P_4 について行えば、

$$\begin{aligned}P_2^* &= \text{conv}(\{(0, 0, 1), (0, 1, 0), (1, 0, 0), (1, -1, -1), (-1, 1, -1), (-1, -1, 1)\}) \\P_3^* &= \text{conv}(\{(3, -2, -1), (-4, -2, -1), (-1/2, 3/2, -1), (2/3, 1/3, 4/3)\}) \\P_4^* &= \text{conv}(\{(1, -1, -1), (1, -1, 1), (1, 1, 1), (1, 1, -1), (-1, 1, 1), \\&\quad (-1, 1, -1), (-1, -1, 1), (-1, -1, -1)\})\end{aligned}$$

10. (研究課題) 与えられた Z^d の点集合できまる凸多面体 P に対してその双対多面体の頂点を計算するプログラムを作成せよ。 P が smooth fano 多面体かどうか判定するプログラムを作成せよ。 (P が原点を内点にもつか?, P の facet の頂点が Z^d の基底か判定する.)

参考: sfano.rr, polymake.rr, dual_p.rr

参考文献

- [Stu] B.Sturmfels, Gröbner Bases and Convex Polytopes, 1995, American Mathematical Society, University lecture series 8.
- [SST] M.Saito, B.Sturmfels, N.Takayama, Gröbner Deformations of Hypergeometric Differential Equations, 2000, Springer.
- [GBN] 日比孝之編, グレブナー基底の現在, 数学書房, 2006

付 録

グレブナー基底の50年

日比孝之

大阪大学大学院情報科学研究科情報基礎数学専攻

2009年9月30日

1

$f = x^3 - x^2y - x^2 - 1$ を $g_1 = x^2 - z$ と $g_2 = xy - 1$ で割り算する。(その1)

$$\begin{aligned} f &= x^3 - x^2y - x^2 - 1 \\ &= x(g_1 + z) - x^2y - x^2 - 1 \\ &= xg_1 - x^2y - x^2 + xz - 1 \\ &= xg_1 - (g_1 + z)y - x^2 + xz - 1 \\ &= xg_1 - yg_1 - x^2 + xz - yz - 1 \\ &= xg_1 - yg_1 - (g_1 + z) + xz - yz - 1 \\ &= (x - y - 1)g_1 + (xz - yz - z - 1) \end{aligned}$$

は割り算であり、その余りは $xz - yz - z - 1$ である。

3

(その1) と (その2) の余りは異なる。ところが、 g_1 と g_2 に $g_3 = -yz + x$ を加え、 f を g_1, g_2, g_3 で割り算すると

$$\begin{aligned} f &= (x - y - 1)g_1 + (xz - yz - z - 1) \\ &= (x - y - 1)g_1 + xz + (g_3 - x) - z - 1 \\ &= (x - y - 1)g_1 + g_3 + (xz - x - z - 1) \end{aligned}$$

となり、(その1) と (その2) の余りは一致する。

● 実際、任意の多項式を g_1, g_2, g_3 で割り算するときその余りは一意である。

● 多項式 g_3 は g_1 と g_2 の最高次の項 x^2 と xz を打ち消し合うことで得られる多項式 $g_3 = yg_1 - xg_2$ である。

5

1. 多変数の多項式の割り算
2. グレブナー基底の定義
3. グレブナー基底の源
4. 計算ソフト Macaulay
5. Buchberger アルゴリズム
6. トーリック生成系
7. 整数計画
8. 凸多面体
9. 計算代数統計
10. 総括

2

$f = x^3 - x^2y - x^2 - 1$ を $g_1 = x^2 - z$ と $g_2 = xy - 1$ で割り算する。(その2)

$$\begin{aligned} f &= x^3 - x^2y - x^2 - 1 \\ &= x(g_1 + z) - x^2y - x^2 - 1 \\ &= xg_1 - x^2y - x^2 + xz - 1 \\ &= xg_1 - x(g_2 + 1) - x^2 + xz - 1 \\ &= xg_1 - xg_2 - x^2 + xz - x - 1 \\ &= xg_1 - xg_2 - (g_1 + z) + xz - x - 1 \\ &= (x - 1)g_1 - xg_2 + (xz - x - z - 1) \end{aligned}$$

も割り算であり、その余りは $xz - x - z - 1$ である。

4

● 変数 x_1, \dots, x_n の単項式とは変数の積 $x_1^{a_1} \dots x_n^{a_n}$ (a_1, \dots, a_n は非負整数) のことである。

● 単項式全体の集合における全順序 $<$ が単項式順序であるとは

- (a) $1 < u$ が任意の単項式 $u \neq 1$ について成立し、
- (b) $u < v$ ならば $uw < vw$ が任意の単項式 u, v, w について成立するときに言う。

● たとえば、辞書式順序は単項式順序である。

● 割り算を議論するときには単項式順序を一つ固定する。

6

2. グレブナー基底の定義

定義 多項式の有限集合 G がグレブナー基底であるとは、任意の多項式 f を G に属する多項式で割り算するとき、その余りが割り算を実行する手続きとは無関係に一意的に定まるときに言う。

7

4. 計算ソフト Macaulay

- グレブナー基底は、廣中と Buchberger の仕事の後、20 余年はそれほどの着目はされなかったようであるが、
- 1980 年代後半、David Bayer と Michael Stillman が可換代数と代数幾何の計算ソフト Macaulay を開発し、
- その背景にはグレブナー基底が潜むことから、グレブナー基底は可換代数と代数幾何の研究者の周知の概念となる。
- その後、CoCoA、Singular あるいは Macaulay2 など、幾多の計算ソフトが現れ、1990 年代以降の可換代数と代数幾何の進展に大きく貢献している。

9

(Buchberger 判定法) 多項式の有限集合 $\{f_1, \dots, f_s\}$ がグレブナー基底となるためには、条件「任意の $1 \leq i < j \leq s$ について $S(f_i, f_j)$ を f_1, \dots, f_s で割り算するとき (うまく割り算すれば) 余りを 0 とすることができる」が満たされることが必要十分である。

- Buchberger 判定法は、多項式の有限集合から出発し、自然に得られるグレブナー基底を探すアルゴリズム (Buchberger Algorithm) を導く。

11

3. グレブナー基底の源

- [1] 廣中平祐の代数多様体の特異点解消論文 (1964 年)
- [2] Bruno Buchberger の学位論文 (1965 年) (師匠は Gröbner)

歴史を遡ると、グレブナー基底の着想は、Macaulay の Hilbert 関数についての研究 (1927 年) にその発祥の源を持つと言える。しかし、Macaulay は、その源が大河に育むとは考えず、一般的な理論を展開するには至らなかったようだ。

8

5. Buchberger アルゴリズム

- S 多項式 の定義

$$f = x_1x_4 - x_2x_3, g = x_4x_7 - x_5x_6$$
$$S(f, g) = x_7f - x_1g = x_1x_5x_6 - x_2x_3x_7$$

- Buchberger の顕著な功績の一つは、多項式の有限集合 G があつたとき、それがグレブナー基底であるか否かを効果的に判定する方法 (Buchberger Criterion) を発見したことである。

10

昨今、Buchberger アルゴリズムには日進月歩の多角的な改良が施されており、グレブナー基底を高速に計算するソフトウェアも開発され、計算速度は飛躍的に進歩している。

たとえば、富士通と富士通研究所において開発が始まった計算機代数ソフトウェア Risa/Asir は、高速なグレブナー基底の計算ソフトウェアの一つとして高い評価を受けている。2000 年 9 月以降は、その開発の拠点が神戸大学理学部に移り、高山信毅と野呂正行を中心としたグループによって開発が続けられている。

12

6. トーリック生成系

行列 $A = (a_{ij})_{\substack{i=1,\dots,d \\ j=1,\dots,n}} \in \mathbb{Z}^{d \times n}$ が配置行列であるとは、 \mathbb{R}^n の原点を通過しない超平面で、 A の列ベクトルのすべてを含むものが存在するときを言う。

● 変数 $x_1, \dots, x_n, y_1, \dots, y_d$ を準備する。

● 連立方程式系

$$x_j - y_1^{a_{1j}} y_2^{a_{2j}} \cdots y_d^{a_{dj}} = 0, \quad j = 1, \dots, n$$

から y_1, y_2, \dots, y_d を消去する。

● すると、 x_1, \dots, x_n の連立方程式系

$$f_1 = 0, \dots, f_s = 0$$

が得られる。

● 二項式 (binomial) の集合

$$I_A = \{f_1, \dots, f_s\}$$

を配置 A の トーリック生成系と呼ぶ。

13

14

例 $A = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 2 & 1 & 0 \\ 0 & 0 & 1 & 2 & 1 \end{bmatrix}$

$$\begin{aligned} x_1 - y_1 &= 0 \\ x_2 - y_1 y_2 &= 0 \\ x_3 - y_1 y_2^2 y_3 &= 0 \\ x_4 - y_1 y_2 y_3^2 &= 0 \\ x_5 - y_1 y_3 &= 0 \end{aligned}$$

$$I_A = \{x_2 x_5^2 - x_1^2 x_4, x_2 x_4 - x_3 x_5, x_2^2 x_5 - x_1^2 x_3\}$$

15

たとえば、二項式 $x_2 x_5^2 - x_1^2 x_4$ を整数ベクトル $[-2, 1, 0, -1, 2]^T$ と同一視すると

$$A \begin{bmatrix} -2 \\ 1 \\ 0 \\ -1 \\ 2 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 2 & 1 & 0 \\ 0 & 0 & 1 & 2 & 1 \end{bmatrix} \begin{bmatrix} -2 \\ 1 \\ 0 \\ -1 \\ 2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

となる。

16

7. 整数計画

● グレブナー基底が応用数学の舞台に登壇したのは、整数計画問題を解くためのグレブナー基底を使ったアルゴリズムを Conti と Traverso が提唱 (1991 年) したときである。

● トーリック生成系とそのグレブナー基底を世に披露し、整数計画問題の最適解の一つがグレブナー基底による割り算の余りから求められることを唱ったのである。

● 彼らのアルゴリズムは、グレブナー基底の代数的な純粹理論からの興味はきわめて深く、グレブナー基底の理論を紹介する講義には不可欠な話題である。

たとえば、標準型の整数計画問題

$$\min\{c \cdot z : Az = b, z \in \mathbb{Z}_{\geq 0}^5\}$$

を考える。但し、 $z = [z_1, z_2, z_3, z_4, z_5]^T$ 、 $b = [25, 34, 18]^T$ 、 $c = [0, 1, 0, 1, 1]$ 、

$$A = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 2 & 1 & 0 \\ 0 & 0 & 1 & 2 & 1 \end{bmatrix}$$

$$\begin{aligned} I_A &= \{x_2 x_5^2 - x_1^2 x_4, x_2 x_4 - x_3 x_5, x_2^2 x_5 - x_1^2 x_3\} \\ \mathcal{G} &= \{x_3 x_5^3 - x_1^2 x_4^2, x_2 x_5^2 - x_1^2 x_4, x_2 x_4 - x_3 x_5, \\ &\quad x_2^2 x_5 - x_1^2 x_3, x_2^3 x_4 - x_1^2 x_3^2\} \end{aligned}$$

17

18

8. 凸多面体

● 単項式順序 $<_c$ は、単項式と非負整数ベクトル（たとえば $x_1^2 x_4$ と $[2, 0, 0, 1, 0]^T$ ）を同一視し、 c と非負整数ベクトルの内積を考え、その内積の大小によって単項式の大小を決める。タイブレークのときは辞書式順序で決着をつける。

● 実行可能解の一つ $[1, 10, 10, 4, 0]^T$ に単項式 $x_1 x_2^{10} x_3^{10} x_4^4$ を対応させる。その単項式を G で割り算する。余りは $x_1^7 x_3^{17} x_5$ となる。すると、 $[7, 0, 17, 0, 1]^T$ が最適解の一つとなる。

19

グレブナー基底が純粋数学における市民権を獲得したのは、グレブナー基底と凸多面体のテキスト

[3] B. Sturmfels, "Gröbner Bases and Convex Polytopes," Amer. Math. Soc., 1995.

の影響も大きい。Gelfand–Kapranov–Zelevinsky らの正則三角形分割の理論を、トーリック生成系のグレブナー基底を使って解釈し、その後の凸多面体の組合せ論の潮流を示唆した。

20

[4] H. Ohsugi and T. Hibi, A normal $(0, 1)$ -polytope none of whose regular triangulations is unimodular, Discrete and Comput. Geom. 21 (1999), 201–204.

における「最大個数の単体から成る三角形分割も最小個数からなる三角形分割も両者とも正則ではない」という凸多面体の発見はそのような潮流におけるもっとも顕著な成果の一つである。

21

9. 計算代数統計

グレブナー基底の統計数学への画期的な応用は

[5] P. Diaconis and B. Sturmfels, Algebraic algorithms for sampling from conditional distributions, The Annals of Statistics 26 (1998), 363–397.

に始まる。マルコフ基底がトーリック生成系のグレブナー基底と対応することを示し、計算代数統計と呼ばれる斬新な研究分野が誕生する契機となった。我が国では、東京大学の竹村彰通のグループが計算代数統計の研究を強力に推進している。

22

Hardy–Weinberg モデル (ABO 式血液型)

遺伝子型	AA	AB	AO	BB	BO	OO
人数 (観測値)	23	10	15	6	17	29

$$\begin{bmatrix} 2 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 2 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 2 \end{bmatrix} \begin{bmatrix} 23 \\ 10 \\ 15 \\ 6 \\ 17 \\ 29 \end{bmatrix} = \begin{bmatrix} 71 \\ 39 \\ 90 \end{bmatrix}$$

Veronese 配置

十分統計量

23

遺伝子型と期待値

AA	AB	AO	BB	BO	OO
12.6025	13.845	31.95	3.8025	17.55	20.25

- 人数が “少ない” から漸近分布論は (?)
- Fisher の正確確率検定は人数が “多い” から (?)
- マルコフ連鎖モンテカルロ法が一つの有効な方法 (!)

24

例 (Groupwise Selection Problem)

- 2種類のケーキ (CケーキとTケーキ) が沢山ある。
- 小テーブルが幾つかあり、それぞれの小テーブルには小皿2枚、中皿2枚、大皿2枚がある。
- 一つの小テーブルに一人が座り、ケーキを6個選び、1枚の皿に1個ずつ置く。

小皿中皿大皿/小皿中皿大皿	人数	小皿中皿大皿/小皿中皿大皿	人数
CCT/CCT	10	TTT/TCT	6
CCT/TTT	27	TTT/CCC	6
CCT/TCT	12	TCT/TCT	0
CCT/CCC	9	TCT/CCC	2
TTT/TTT	7	CCC/CCC	0

31

十分統計量

小皿C	小皿T	中皿C	中皿T	大皿C	大皿T
85	73	105	53	17	141

32

[7] S. Aoki, T. Hibi, H. Ohsugi and A. Takemura, Markov basis and Gröbner basis of Segre–Veronese configuration for testing independence in groupwise selections, *Annals of the Institute of Statistical Mathematics*, to appear, arXiv:0704.1074v2 [math.ST].

- 大学入試センター試験の科目選択の統計的検証
- Oguma et al., 2004 の遺伝子データの検証

33

10. 総括

科学技術振興機構 (JST) の戦略的創造推進研究事業の研究領域「数学と諸分野の協働によるブレークスルーの探索」の CREST の研究課題

現代の産業社会とグレブナー基底の調和
(2008年10月 – 2014年3月)

<http://www.math.jst.go.jp/ja/scientists/teamhibi/index.html>

は、以上のような背景を踏まえ、研究活動を展開している。

34

- 1964年 廣中平祐の特異点解消論文
- 1965年 Bruno Buchberger の学位論文
- 1986年 Macaulay
- 1991年 整数計画
- 1995年 凸多面体
- 1998年 計算代数統計
- 2008年10月 JST CREST Gröbner Bases
- 2014年3月

35

感想文

グレブナースクール感想文

東京女子大学大学院理学研究科数学専攻

D1 長廻 美和

全体的に内容は私にとっては難しかった。1日目のグレブナー基底の基本的な概念は勉強していたので理解できたが、2日目以降の講義は難しかった。最終日の展望1~5では、グレブナー基底がどのようなところに応用されているのかが聞けて勉強になった。公開鍵暗号とグレブナー基底の関係について初めて知った。情報の授業で暗号について少し学んだことがあるが、そのときにはグレブナー基底の話は全く出てこなかった。

今回のスクールでは日頃のゼミとは違う環境で勉強できてよかった。日頃とは他大学・他学部の人と勉強することはあまりないので、いい経験だったと思う。こういう機会があったら、また参加したいと思う。

KNOPPIX/mathが少し使えるようになったのもよかった。せつかくコンピュータ実習もやったので、もっとコンピュータも使いこなせるようになりたいと思う。

グレブナスクールに参加しての感想

北海道大学 理学院 数学専攻 M1 渡邊 亮

グレブナ基底を少し勉強したことがあったので臨んだスクールでしたが、グレブナ基底に限らずもっと広く、計算機を用いて数学を研究していく方法、とくにその導入を学んだように思っています。とても有意義でした。

計算機を用いた数学、その具体例としてグレブナ基底を扱ったとも思うのですが、グレブナ基底が計算機を用いた数学の中でどれほど重要な役割を担っているかということも、実際の演習のほか、講義で知ることができました。他にグレブナ基底の、多様な分野での応用例を知ることができたことも、大変参考になりました。今振り返るに、演習と講義をあのように織り交ぜるという形式は、グレブナ基底というものを理解するにはとても適していたと思います。

自分のパソコンでも気軽に計算できる環境を頂いたので、今回のスクールで習ったことは勿論、できればそれ以外のことも、数学を研究する上で利用していければと思います。

JST CREST GRÖBNER SCHOOL の感想

九州大学大学院数理学府数理学専攻

大輪拓也

2009年9月28日(火)

全体を通してとても丁寧な講演が多く、その意味で興味深く各々の問題に取り組むことが出来たが、私が特に興味を持ったのは9月15日(火)に講演された「マルコフ基底と実験計画」である。

私の専門は確率論であり、計算機科学における動機付けを持つ確率的な議論には関心が強い。最近の研究では、MCMCと呼ばれるマルコフ連鎖を用いたモンテカルロ法に関する数学的な考察を行っており、それは今回講演された内容と密接に関係がある。MCMCを行う際に必要となる議論を大きく二つに分けると、対象物を行き渡るマルコフ連鎖の設計についてと、そのマルコフ連鎖が所望の(定常)分布へ収束するまでの早さの算出もしくは評価である。私がこれまで取り組んできたのはどちらかと言えば後者についてだが、後者を議論するためには前者に対する詳しい考察が必要となることがしばしばある。つまり、マルコフ連鎖の設計方法やそこに使われる性質を詳しく調べることで、収束の早さを見極められることがある。

「マルコフ基底と実験計画」の理論の中心は、MCMCを行うある対象物(分割表)に関するマルコフ連鎖の設計方法であり、そこで使われたのがグレブナー基底を用いた手法であった。これには大きな感銘を受け、この手法の更なる考察や応用に興味を持った。まず、この手法を用いることが収束スピードの算出にどれだけ結びつけられるかが、やはり気になるところである。また、より複雑な分割表やその他の対象物に関するマルコフ連鎖の設計もこのような手法を用いる事が出来るのか、についても興味がある。これらのことを考察するためには、計算機を用いて様々な例を具体的に計算してみることが大きな助けになるだろう。そのために、今回の研究集会で紹介された計算ソフトを駆使することで、今後の研究に大いに役立つことが期待される。

グレブナースクール 2009 に参加して

東京大学理学系研究科 博士課程 1年 野口 卓也

私がこのスクールの存在を知ったのは統計関連のメーリングリストへの投稿だった。学部時代に興味を持って登録してからそのままであった。大学院に進級するにつれ、少しずつ専門分野が代数から遠ざかっていたが、実践的内容（演習）が盛り込まれていることも魅力的に感じ、日程も何とか調整できたので、せっかくチャンスだと思い、今回参加することにした。多少のためらいはあり、会場に着いてからも不安であったが、スクールが始まると同時にそれにのめりこむことができたので、その点はよかった。初日午前中の日比先生講義の勢いには圧倒されたが、逆にそのおかげでスクールへの導入がうまく行っていたと、終わってみて感じている。午前の講義と午後の実習のスタイルは集中しやすくて好感を持たれた。私の研究分野の集会では時に朝早くから夜遅くまで人の話を聞くスタイルになってしまっていて集中力が続かない場合も出てきているのは、自分の中での以前からの課題であったので、今回のスクールで満足できた点であった。

演習が特に役に立ったと思っていて、その場限りではなく、今後につながるような形式の環境整備から始まるところが非常によかったと思う。仮想マシンの利用により、すぐに使える状態に持っていけるのは敷居が下がっているのは、スクール準備に携わってくださった研究者の方々のおかげであると感じました。

このスクールは講義・実習・応用事例の三つに分けられる中で、個人的には最終日の応用につながるセッションを聞いてみたかったが、やはり純粋科学の研究をすぐに応用に持っていくのは難しいのだな、とよくも悪くも感じてしまった。

また、ランチタイムが長めに取ってあるのも嬉しい配慮だった。ランチ後、実習が始まる前に、自分でそれまでの分を見直したり、近隣の席の人としゃべって親睦を深めたりなど主体的に動ける時間が持てたのはよかった。

4日目の懇親会で同世代の同志と交流ができたことは、どの「スクール」に行っても思うことなのだが、非常に有意義でありその場も楽しむことができてよかった。今後の博士課程の進め方やその後の生き方も人それぞれの考えも共有できたのも日ごろの励みにつながるができるだろう。

今回のスクール会場であった神戸大学理学研究科のある六甲台地区は駅から見て山の上的ほうにあり、毎朝くじけそうになることもあったが、休憩やランチタイムに眺める神戸市街地・神戸港を眺めて元気をもらった。繁華街から少し離れて自然も残る豊かな場所で受講できたのも効果的であった。

最後に、このように大変有意義な本スクールへ参加にあたって、経済的支援および準備・担当をいただいた CREST プロジェクトおよびその関係者にお礼を述べたい。またスクール参加者の多数の方々からも多くの刺激とアイデアをいただいたことに感謝を示したい。

Groebner school に参加した感想

新潟大学大学院 自然科学研究科

飯塚 由貴恵

Groebner school には研究室の教授に、今後グレブナー基底を研究で使うことになることからぜひ参加するよう言われたことがきっかけで参加した。グレブナー基底についてはまだ大学院の講義で軽く触れられただけだったので、スクールに参加することが場違いなのではないかと思っていた。実際に参加してみると、午前中の講義ではやはり聞いたことの無い用語が多く、理解しようと頑張るものの付いていけずという状態だった。その分理解できたときは嬉しかった。午後の実習では、私自身工学部を卒業したこともあってか UNIX の操作には慣れていたので、コマンド操作等については付いていくことができたが、各ソフトの使い方や関数の仕様等は難しく、講義資料(配布された講義資料はとても濃く素晴らしい内容だったと思う。作成者に感謝したい。)の通りにキーボードを打ち込んで、どのような結果が出力されるか、打ち込んだ関数はどのようなものなのかを1つ1つ見ていくのが精一杯だった。

他大学の参加者や教授陣のレベルの高さに圧倒され、何もできずに終わってしまったのではないかとも感じている。しかし、短い期間ではあったが、そのようなハイレベルな環境で勉強することで自分自身良い刺激を受けることができたと思う。

また、KNOPPIX/MATHには多くの数学系ソフトがインストールされており、これらのソフトも配布された講義資料を参考にしつつ、これから徐々に使いこなしていきたい。

グレブナ学校のご感想

北海道大学理学院数学専攻修士1年 成瀬俊輔

現在の私の研究対象ではグレブナー基底は特に使わないと思われるのですが、計算代数のソフトウェアは使うので、そのようなソフトウェアの使い方を覚えることを目的として参加しました。その観点からいけば、いくつかのソフトウェアの基本的な使い方は理解することができ、これから自分で使うときのための足がかりになったと思います。

一方、グレブナー基底の応用例の数々も興味深く聞きました。今回グレブナー基底に関係する諸分野の話ということで、通常ならばまず聞くことはなく、また聞こうともしない話を聞いたことは貴重な体験でした。特に初日の講義での純粋数学の方面での使われ方や、統計の方面への応用と統計と代数との関係は興味深く感じました。また今回講義されたのは組合せ論や統計、微分環などでの使用についてでしたが、その他の分野、たとえば代数幾何や数論、またより一般的な環での使用などがあれば聞きたいです。

ところで今回実習において、あらかじめ用意されたプログラム（Risa/Asir用のコードなど）を与えられ、それを用いて計算するということがありましたが、この方法だと結果は出せても内部でどのような操作が行われているのかわからないことがありました。時間との関係上仕方のない方法なのかもしれませんが。例えば基本的な関数とグレブナー基底を計算させる関数だけを用いて、それらを組み合わせて自らプログラムを書くといった形式の方がより理解できるのではないかと思います。

グレブナースクール（9月14日（月）～18日（金））に参加させていただいた感想

この度は JST CREST より神戸大学にて開かれたグレブナースクールにて、グレブナー基底にかかわる様々な発表を聞かせていただきました。私はグレブナー基底について統計学と関連付けて研究を行っており、グレブナー基底について広い知識を求めていました。

スクールではいろいろな方が様々な研究内容の発表を講じてくださり、現在グレブナー基底がどのような分野で扱われるのか、またどのような発展の展望があるのか、その用途は多種多様であり、数学だけにとどまらず計算機、経済関係など、この基底の重大さを改めて認識いたしました。また、自分の研究に関係のあるもの、興味深い内容であるもの、様々なお話を聞かせていただくことで自分の知識の幅を大きく広げることができました。

交流の場としてもとても有意義なものでした。自分に必要な知識を持っている方、興味のある内容の研究をしている方とお互いに交流を深めることでこれからの研究を助け合っていくことができ、参加されたみなさんにとって非常に嬉しい場であったと思います。僕自身、いろいろな方とお話をし、研究の視野が広がり、これからの研究の方向性が見えてくる要因となりました。

また、とても助かったことの一つが仮想マシンのインストールでした。私は計算機が若干苦手でしたが自分のパソコンに仮想マシンと様々な計算ソフトをスクール内でインストールしてくださり、実際に計算機の演習もあったので使い方を覚え、それからの研究がとてもはかどる結果となりました。

このような有意義な5日間を企画してくださった、大阪大学大学院情報科学研究科 / JST CREST の日比孝之先生には多大なる感謝をしております。このスクールがなければ、（まだ途中ではありますが）現在の研究成果は得ることができていませんでした。

私も次は発表をする側の立場になり、皆さんの研究に役立てられるような発表ができるよう精進をしていきたいと思っております。

立教大学大学院 理学研究科 数学専攻 前期課程 2年
平山尚樹

グレブナースクール参加の感想

東京大学大学院 情報理工学系研究科
数理情報学専攻 数理第4研究室 修士1年
工藤 大誠

tkudo@stat.t.u-tokyo.ac.jp

2009年11月27日

グレブナースクールの中間の3日間は午前中に講義をし、午後に演習をするという構成になっていましたが、その点がとてもありがたかったです。講義でついていけなかった内容も、演習でTAの方に教わりながら手を動かすことで理解できたり、雰囲気だけでもつかめたように思います。

また、各問題を解くためのツールの使い方を教えていただいたこと、及びそれらのツールを整える環境であるKNOPPIX/Mathの導入をTAの方に質問しながらできたこともありがたかったです。Risa/Asir、Macaulay2、gfan等のツールの使い方を、ツールを起動する所から、プロジェクトを使って実演しながら教えていただき、わかりやすかったです。これらは一人でやっていたら多くの時間と労力が必要となっただろうと思うので、大変助かりました。

さらに、それらのツールだけでなく、演習の個別の問題を解くためのプログラムのソースがあったこともありがたかったです。例えば2日目の分割表のMCMC法の実装等は今後より大きい問題を解くためのプログラミングの参考になるように思いました。MCMC法等のアルゴリズムは理解できても、一から実装となると中々手間がかかるので大変役に立つと思いました。

最後に、スクール自体の感想ではないかもしれませんが、NOPPIX/Mathは大変便利なものだな、と思いました。少し使ってみたいがインストールやその後の管理等の手間を考えると敬遠していたツールがいくつかありましたが、NOPPIX/Mathではツールがまとまっており、またドキュメントも豊富で助かりました。今後も使おうと思います。

グレブナー・スクール in 神戸大学

佐藤 宏平*

首都大学東京大学院理工学研究科

2009年9月14日～9月18日

神戸大学で9月14日から18日に行われたグレブナー・スクールに参加させて頂きました。ホテルは三宮にとりましたが、三宮の街は、大震災の後影もすっかりなくなっており、大都市としての華やかさに満ち溢れていました。街の人に伺ったところ、街が一丸となって努力し今の復興後の三宮の姿になったそうです。三宮の街の風景を見ていると、感動させられるものがありました。

さて、今回のグレブナースクールでは、いろいろな分野の専門家による講演を拝聴することで、沢山の興味深い研究や概念に出会うことができ、また特に統計の専門家の方々が考える問題を詳しく知ることができ、貴重な知識を得ることができました。計算機の演習時間には、具体的な演習問題に対し、計算機を使いながら答えを出すという作業を行いました。今まで使ったことのない、いろいろな計算プログラムをパソコン上で動かさせたのは、大変に有意義な時間でした。

1 グレブナー扇

今回のグレブナースクールの中で学んだものの中で、特にグレブナー扇について、自分なりにまとめてみました。主な参考文献は [1] です。

1.1 重み付き順序

以下、 k は体とし、 $S = k[x_1, x_2, \dots, x_n]$ とする。また、 $w \in \mathbb{R}^n$ を重みベクトルと呼ぶことにする。

定義 1.1 (イニシャル形式). 0 でない S の多項式 $f = \sum_a c_a x^a$ に対し、内積 $w \cdot a$ の値が最大となるような $c_a x^a$ の和を f の w についてのイニシャル形式といい $\text{in}_w(f)$ と表す。

次に、 S のイデアル I に対して、 I の w についてのイニシャルイデアルを以下のように定める。

$$\text{in}_w(I) = (\{\text{in}_w(f) \mid 0 \neq f \in I\}).$$

即ち、イデアル I に属する全ての多項式の w についてのイニシャル形式によって生成されるイデアルである。

命題 1.2 (重み付き順序). $w \in (\mathbb{R}_{\geq 0})^n$ とし、 $<$ を S の単項式順序とする。このとき以下のように重み付き順序を定める。

$$x^a <_w x^b \stackrel{\text{def}}{\iff} w \cdot a < w \cdot b \text{ 又は } w \cdot a = w \cdot b, x^a < x^b$$

このとき $<_w$ は S の単項式順序である。

*sato-kohei@ed.tmu.ac.jp

上の命題の中で、重みベクトルを $(\mathbb{R}_{\geq 0})^n$ に制限して考えたが、こうしないと重み付き順序は単項式順序にならない。しかし、イニシャル形式やイニシャルイデアルのみを扱い、重み付き順序の単項式性を要求しない時には、重みベクトルを制限せずに \mathbb{R}^n の元として考えてよい。

以降、イデアル I は 0 でないとする。

命題 1.3. 任意の単項式順序 $<$ と S の任意のイデアル I に対して、 $\text{in}_{<}(I) = \text{in}_w(I)$ を満たす $w \in \mathbb{R}^n$ が存在する。

注意 1.4. w の取り方によっては、 f のイニシャル形式が単項式とならない場合があるので、明らかに $\text{in}_w(I)$ は必ずしも単項式イデアルとはならない。

注意 1.5. 任意の S のイデアル I に対して、 $\text{in}_{<}(\text{in}_w(I)) = \text{in}_{<_w}(I)$ が成り立つ。

以上から、 I と $<$ を固定すると、 w は自然に定まることが分かる。次に、 I と $<$ を固定し、 w を分類することを考える。

定義 1.6. 重みベクトル $w, w' \in \mathbb{R}^n$ とイデアル $I \subset S$ を固定する。このとき、以下のように関係 \sim を定める。

$$w \sim w' \stackrel{\text{def}}{\iff} \text{in}_w(I) = \text{in}_{w'}(I)$$

注意 1.7. 上の関係 \sim は同値関係である。

1.2 グレブナー扇

上の節で定めた同値関係により、 \mathbb{R}^n を分類したものがグレブナー扇である。以下で詳しく述べる。

命題 1.8 (B. Sturmfels.). 任意のイデアル $I \subset S$ と任意の重みベクトル w に対して、 w の同値類 $c[w]$ は相対的開の凸多面錐となる。この $c[w]$ を w の I -同値類と呼ぶ。

$c[w]$ が相対的開であるとは、簡単に言い表すと、凸多面錐 $c[w]$ の境界が $c[w]$ に含まれないことである。

注意 1.9. $w \in (\mathbb{R}_{\geq 0})^n$ が十分にジェネリックな点のときには、 $c[w]$ は重み付き単項式順序 $<_w$ についての I の被約グレブナー基底 \mathcal{G} により定まる。即ち、

$$c[w] = \{w' \in \mathbb{R}^n \mid \text{in}_w(g) = \text{in}_{w'}(g), \forall g \in \mathcal{G}\}$$

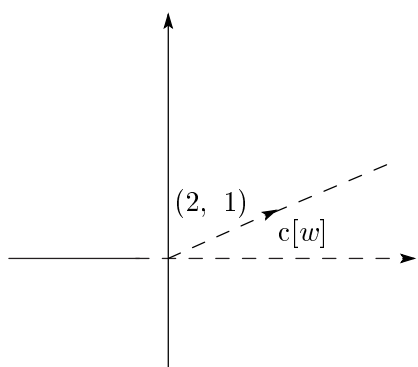
上で、十分にジェネリックな点とは、凸多面錐 $c[w]$ の境界に含まれない点という意味である。

例 1.10. $n = 2$ の場合について、以下のような例を考えよう。

$I = (x^3 - 1, y^3 - 1, xy - 1)$ とし、 $w = (100, 1)$ とする。また固定する順序 $<$ は順辞書式順序とする。このとき I の $<_w$ についての被約グレブナー基底 $\mathcal{R}\mathcal{G}_w$ は $\{y^3 - 1, x - y^2\}$ である。このとき、

$$c[w] = \{(w_1, w_2) \in (\mathbb{R}_{\geq 0})^2 \mid w_1 > 2w_2, 3w_2 > 0\}$$

(但し、上式の右辺の不等式の $>$ は数値的な大小を表す。)



次に、この同値関係 \sim を使って、 $w \in (\mathbb{R}_{\geq 0})^n$ を類別していくことを考える.

定義 1.11 (グレブナー扇). S のイデアル I のグレブナー扇を以下で定める.

$$\text{GF}(I) := \{\tau \mid \tau \prec \sigma, \sigma = \overline{c[w]}, w \in \mathbb{R}^n\}$$

即ち、全ての $w \in \mathbb{R}^n$ に対する I -同値類 $c[w]$ の closure を取る. これは相対的閉の錐体 σ (これをグレブナー錐体と呼ぶ) となり、 σ に含まれる全ての面 τ からなる集合を $\text{GF}(I)$ で表わす.

命題 1.12. $\text{GF}(I)$ は扇である.

これは、定義より明らかである.

注意 1.13. $\text{GF}(I)$ は、必ずしも強凸でない.

例 1.14. 例えば以下のような場合には、 $\text{GF}(I)$ は強凸でない.

$S = k[x, y]$, $I = (x)$ とする. このとき、例えば $w = (100, 1)$ に対して $\overline{c[w]} = \{(w_1, w_2) \in \mathbb{R}_{\geq 0} \times \mathbb{R}\}$ となり、明らかに $\text{GF}(I) \cap \{-\text{GF}(I)\} = \mathbb{R}$ となり、強凸ではない.

注意 1.15. $\text{GF}(I)$ を求めるアルゴリズムが存在する.

例 1.16. 例 1.10 に沿って、 $\text{GF}(I)$ を求めよう. ちなみに、以下の被約グレブナー基底は、計算機プログラム Risa/Asir¹ ([2] 参考) で計算したものである.

$w_1 = (100, 1)$ の時には、 $c[w_1] = \{(w_1, w_2) \in \mathbb{R}^n \mid w_1 > 2w_2, 3w_2 > 0\}$ であった.

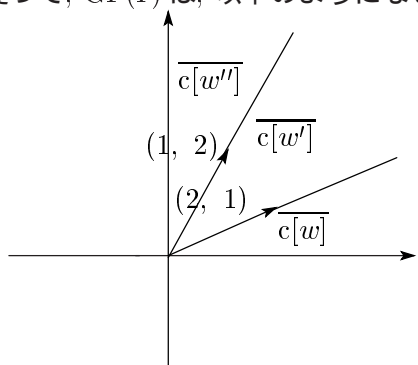
次に、 $w' = (9, 5)$ を取ると、 $\mathcal{RG}_{w'}$ は $\{2y^2 - x, xy - 1, x^2 - y\}$ である. よって、

$$c[w'] = \{(w_1, w_2) \in (\mathbb{R}_{\geq 0})^2 \mid 2w_2 > w_1, w_1 + w_2 > 0, 2w_1 > w_2\}$$

$w'' = (1, 3)$ を取ると、 $\mathcal{RG}_{w''}$ は $\{-x^2 + y, x^3 - 1\}$ である. よって、

$$c[w''] = \{(w_1, w_2) \in (\mathbb{R}_{\geq 0})^2 \mid 2w_1 > w_2, 3w_1 > 0\}$$

従って、 $\text{GF}(I)$ は、以下のようになる.



¹<http://www.math.kobe-u.ac.jp/Asir/asir-ja.html>

ちなみに、上の例は商特異点 \mathbb{C}^2/G のトーリック最小特異点解消に対応している。但し G は $\begin{pmatrix} \epsilon & 0 \\ 0 & \epsilon^2 \end{pmatrix}$, (ϵ は 1 の原始 3 乗根) で生成される $SL(2, \mathbb{C})$ の有限アーベル部分群とする。つまり、不変式環 $\mathbb{C}[x^3, y^3, xy]$ の Spec で定められる多様体の最小特異点解消を与えたのである。

これは、複素 2 次元では、 $GL(2, \mathbb{C})$ の有限アーベル部分群 G に対する商特異点 \mathbb{C}^2/G の最小特異点解消はグレブナー扇に対応するという事実を使ったものである。群 G の位数とグレブナー錐体の数が一致しており、各グレブナー錐体が非特異であることから、このグレブナー扇はちゃんと $GF(I)$ になっていることがチェックできる。参考文献としては大袈裟かもしれないが、[3] を挙げておく。

参考文献

- [1] B. Sturmfels, *Groebner Bases and Convex Polytopes*, University Lecture Series, Vol.8, AMS.
- [2] 野呂正行, 高山信毅, *Risa/Asir ドリル*, 2007,
<http://www.math.kobe-u.ac.jp/~taka/2007/asir-book-2007.pdf>.
- [3] T. Bridgeland, A. King and M. Reid, *The McKay correspondence as an equivalence of derived categories*, J. Amer. Math. Soc. 14 (2001), 535– 554.