代数幾何ゼミ

石井大海

早稲田大学基幹理工学部
数学科四年

2013 年 11 月 27 日

2.3 Gröbner 基底の変換

- 零次元イデアル I のある単項式順序に関する Gröbner 基底 G を，他の単項式順序に関する基底 G' に変換したい．
  - $\rightsquigarrow$ 剰余環 $A = k[\mathbf{X}]/I$ の線型代数を使おう！
    - Faugère, Gianni, Lazard and Mora による FGLM algorithm
- 実用上は lex に関する基底に変換出来れば十分なので，その場合を扱う．
- 最初に grevlex など効率的なので計算しておいて，必要になったら消去などに使う
- 入力：零次元イデアル I を生成する Gröbner 基底 G
- lex に関して増加順に $k[X]$ の単項式を処理していく．
- ステップが進むごとに，最終的な Gröbner 基底の部分集合 G_{lex}，A の基底 B_{lex} を増やしていく．
- アルゴリズムは三つの段階から成る．

FGLM アルゴリズム I

前処理 $G_{lex} = B_{lex} = \emptyset, \mathbf{X}^{\alpha} = 1$

Main Loop

1. 入力：単項式 $\mathbf{X}^{\alpha}$
2. $\overline{\mathbf{X}^{\alpha}}^{G}$ を計算する.
3. $\overline{\mathbf{X}^{\alpha}}^{G}$ が B_{lex} の各元の剰余と一次従属か？

Case (a) 一次従属の時：

$$\overline{\mathbf{X}^{\alpha}}^{G} + \sum_{j} c_j \overline{\mathbf{X}^{\alpha(j)}}^{G} = 0$$

となる線形結合を見付けだし,

$$g = \mathbf{X}^{\alpha} + \sum_{j} c_j \mathbf{X}^{\alpha(j)} \in I$$

を G_{lex} の末尾に付け加える. 単項式は増加順に見ているので, $\mathrm{LT}(g) = \mathbf{X}^{\alpha}, \mathrm{LC}(g) = 1$.

Case (b) 一次独立の時：$\mathbf{X}^{\alpha}$ を B_{lex} に付け加える.

FGLM アルゴリズム II

Termination Check Main Loop で g が G_{lex} に追加されており，$\mathrm{LM}(g)$ が最大の変数の冪なら G_{lex}, B_{lex} を出力し終了する．

Next Monomial lex 順序に関して $\mathbf{X}^{\alpha}$ よりも大きく，どんな $g_i \in G_{lex}$ に対しても $\mathrm{LT}(g_i)$ で割れないような最小の単項式を取り，それを次の入力として Main Loop を実行する．

- このアルゴリズムの正当性は定理 3.4 で示す．
- その前に実際の計算例を見てみよう．

FGLM アルゴリズムの計算例 I

Exercise 3-1

$I = \langle xy + z - xz, x^2 - z, 2x^3 - x^2yz - 1 \rangle \subseteq \mathbb{Q}[x, y, z]$ を考える．$x > y > z$ なる *grevlex* についての I の *Gröbner* 基底を G として，G を $z > y > x$ なる *lex Gröbner* 基底 G_{lex} に変換したい．

(a) *Main Loop* を $1, x, \ldots, x^6$ まで実行せよ．

(b) x^6 の次に y が選ばれることを示し，計算を実行せよ．

(c) y の次に z が選ばれることを示し，計算を実行せよ．

(d) z を処理した後，アルゴリズムは停止する事を示せ．

(e) G_{lex} が求める *lex Gröbner* 基底であることを示せ．

grevlex に関する基底 G は，

$$G = \left\{ \begin{array}{l} f_1 = z^4 - 3z^3 - 4yz + 2z^2 - y + 2z - 2 \\ f_2 = yz^2 + 2yz - 2z^2 + 1 \\ f_3 = y^2 - 2yz + z^2 - z, \quad f_4 = x + y - z \end{array} \right\}$$

FGLM アルゴリズムの計算例 II

で与えられる．よって $\langle \mathrm{LT}(I) \rangle = \langle z^3, yz^2, y^2, x \rangle$ である．この時，剰余環の基底は $B = \{1, y, z, z^2, z^3, yz\}$ であり，剰余環の元は全てこの $\mathbb{Q}$-線型結合で書ける．この事を踏まえて，問題を解く．

(a) 計算のため，まず各剰余の係数を求めておく．

	1	x	x^2	x^3	x^4	x^5	x^6
1	1					1	
y		-1					
z		1	1				
z^2				1	1	-2	
z^3						1	1
yz				-1		2	

これをよく見ると，$1, \ldots, x^5$ は全て一次独立である．よって，Main Loop は $\mathbf{X}^\alpha = 1, x, \ldots, x^5$ の間は B_{lex} を増やすのみであり，Termination Check は行われない．x^5 を処理した後に Next Monomial によって x^6 が選ばれるが，この表より

FGLM アルゴリズムの計算例 III

$\overline{x^6}^G = \overline{x^5}^G + 2\overline{x^3}^G - \overline{1}^G$ となる．よってこの後の Main Loop により，次の結果が得られる：

$$B_{lex} = \{1, x, \ldots, x^5\} \quad G_{lex} = \{x^6 - x^5 - 2x^3 + 1\}$$

(b) x^6 を処理した後，G_{lex} の元がはじめて得られた．この先頭項は x^6 である．よって，x について 6 以上の次数を持つ単項式は処理から弾かれる．よって，x^6 より大きく，次に処理されるべき単項式は y である．すると，先程の表から $\overline{y}^G = \overline{x^2}^G - \overline{x}^G$ となり，一次従属．よって，B_{lex} は変化せず，

$$G_{lex} = \{x^6 - x^5 - 2x^3 + 1, y - x^2 + x\}.$$

(c) 前と同様の議論により，次に選ばれる単項式は z となる．この時，表から $z = x^2$ なので，

$$G_{lex} = \{x^6 - x^5 - 2x^3 + 1, y - x^2 + x, z - x^2\}.$$

FGLM アルゴリズムの計算例 IV

(d) すると，$\mathrm{LT}(z - x^2) = z$ であり，今計算しているのは $z > y > x$ なる lex 順序であったので，g の先頭単項式は最大の変数となった．よって停止する．

(e) ここでそれぞれ求めた順に g_1, g_2, g_3 とおくと，

$$
\begin{aligned}
S(g_1, g_2) &= -yx^5 - 2yx^3 + y + x^8 - x^7 \\
S(g_1, g_3) &= -zx^5 - 2zx^3 + z + x^8 \\
S(g_2, g_3) &= -zx^2 + zx + yx^2
\end{aligned}
$$

となる．これらを (g_1, g_2, g_3) で割った剰余はいずれも 0 となるので，G_{lex} は $\langle G_{lex} \rangle$ の lex Gröbner 基底である．特に，I の生成系はいずれも G_{lex} で割り切れ，G_{lex} の元は G で割り切れるので，I の Gröbner 基底となることもわかる．

アルゴリズムの正当性 I

Th. 4

FGLM アルゴリズムは，零次元イデアル I の適当な単項式順序に関するどんな *Gröbner* 基底 G を入力としても，必ず停止し，lex に関して G_{lex} は I の *Gröbner* 基底，B_{lex} は剰余環 $k[\mathbf{X}]/I$ のベクトル空間の基底となる．

観察．B_{lex} に追加される単項式は，lex 順序について真に増加する．同様に $G_{lex} = \{ g_1, \ldots, g_k \}$ の時 $\mathrm{LT}(g_1) <_{lex} \cdots <_{lex} \mathrm{LT}(g_k)$ となる．また，Main Loop で G_{lex} に g_{k+1} が追加される時，$\mathrm{LT}(g_{k+1})$ は Main Loop の入力単項式になっており，その入力は Next Monomial 手続で生成されるので，

$$\mathrm{LT}(g_{k+1}) \text{ は } \mathrm{LT}(g_1), \ldots, \mathrm{LT}(g_k) \text{ のいずれでも割れない} \quad (0.1)$$

が各 k について成立していることに注意する．

アルゴリズムの正当性 II

停止性．G が零次元イデアルを生成するならアルゴリズムは必ず停止することを示す．そこで，停止しなかったとして矛盾を導こう．すると，Main Loop の Case (a) または (b) の少なくとも一方が無限回発生するので，どちらが無限回かで場合分けする：

- (a) G_{lex} は無限個の単項式 $\{\mathrm{LT}(g_1), \mathrm{LT}(g_2) \ldots\}$ を含む．そこで，$J = \langle \mathrm{LT} g_i \mid i < \omega \rangle$ とおけば，Dickson の補題よりある $N > 0$ があって $J = \langle \mathrm{LT}(g_1), \ldots, \mathrm{LT}(g_N) \rangle$ となる．このとき，$\mathrm{LT}(g_{N+1}) \in J$ なので，$\mathrm{LT}(g_{N+1})$ は $\mathrm{LT}(g_1), \ldots, \mathrm{LT}(g_N)$ のいずれかで割り切れることになる．しかし，これは (0.1) に反する．よって G_{lex} は有限個で止まる．
- (b) これが無限回繰り返されたとすると，B_{lex} は $k[\mathbf{X}]/I$ の一次独立な単項式を無限個含むが，これは I が零次元であることに反する．

アルゴリズムの正当性 III

G_{lex} が lex Gröbner 基底であること．$G_{lex} = \{ g_1, \ldots, g_k \}$ を出力して停止したとする．すると，停止条件より $\mathrm{LT}(g_k) = x_1^{a_1}$ である（但し $x_1 > \cdots > x_n$）．そこで，G_{lex} が lex に関する Gröbner 基底になっていないとして矛盾を導こう．G_{lex} は Gröbner 基底でないので，$\mathrm{LT}(g)$ が $\mathrm{LT}(g_1), \ldots, \mathrm{LT}(g_k)$ のいずれでも割れないような $g \in I$ を取れる．特に，g は G_{lex} について被約（$g = \overline{g}^{G_{lex}}$）であるとして一般性を失わない．特に，そのような g のうち $\mathrm{LT}(g)$ が最小となるものを取ろう．
$\mathrm{LT}(g)$ が lex 順序について $x_1^{a_1}$ よりも大きいとすると，$\mathrm{LT}(g_k) \mid \mathrm{LT}(g)$ となってしまうので，$\mathrm{LT}(g) < \mathrm{LT}(g_k)$ である．特に，アルゴリズムが単項式を lex で小さい方から処理していくことから，ある $i < k$ があって $\mathrm{LT}(g_i) < \mathrm{LT}(g) \leq \mathrm{LT}(g_{i+1})$ となることがわかる．また，g の非先頭単項式はみな $\mathrm{LT}(g)$ よりも真に小さく，g が被約であることからどんな $\mathrm{LT}(g_j)\, j \leq i$ でも割り切れない．以上のことから，g の非先頭単項式は，$\mathrm{LT}(g)$ が Next Monomial で検討されるより以前に B_{lex} に追加されていることが

アルゴリズムの正当性 IV

わかる．よって，g は g_i が G_{lex} に追加された次に G_{lex} に追加されることになる．よって $\mathrm{LT}(g) = \mathrm{LT}(g_{i+1})$ となるが，これは g の取り方に反する．よって G_{lex} は I の lex に関する Gröbner 基底である．

B_{lex} が A の基底であること（演習問題 4）．Main Loop より $B_{lex} = \{\mathbf{X}^{\alpha(1)}, \ldots, \mathbf{X}^{\alpha(\ell)}\}$ は一次独立である．そこで，更に $\mathbf{X}^{\alpha}$ が B_{lex} と一次独立になったとする．このとき，$\mathbf{X}^{\alpha} \leq X^{\alpha(\ell)}$ とすると，$\mathbf{X}^{\alpha}$ は B_{lex} に含まれていなくてはいけないので，矛盾である．よって $\mathbf{X}^{\alpha} > \mathbf{X}^{\alpha(\ell)}$ となる．この時，$\mathbf{X}^{\alpha}$ が A の基底であることから，$\overline{\mathbf{X}^{\alpha}}^{G_{lex}} = \mathbf{X}^{\alpha}$ であり，基底は G_{lex} のどの先頭多項式でも割れないので，$\mathbf{X}^{\alpha} < \mathrm{LT}(g_k) = x_1^{a_1}$ である．よって，$\mathbf{X}^{\alpha}$ は Next Monomial でどこかでリストアップされてなくてはならないので矛盾．よって B_{lex} は A の基底である．∎

関連事項

- FGLM アルゴリズムは零次元イデアルに関するアルゴリズム
- しかし，Gröbner 基底の元の次数の上界が与えられていれば，正次元のイデアルにも適用出来る．
- これと関連した，斉次イデアルに対する「Hilbert 函数駆動」アルゴリズムもある

? 次数の上界がわからない一般のイデアルを変換出来ないか？

  ⇝ Gröbner Walk（第八章）

FGLM アルゴリズムの一般化 I

- FGLM アルゴリズムは異なる文脈にも応用出来る
- FGLM アルゴリズムで元の Gröbner 基底 G を使うのは f に対し標準形 $\overline{f}^G$ を求めるときのみ
- そこで今までの議論を線型写像を用いて言い換えてみる
- $B : G$ に関する $A = k[\mathbf{X}]/I$ の基底, $L(f) = \overline{f}^G$, $V = \mathrm{span}(B)$ とおく．これにより，写像

$$L : k[\mathbf{X}] \to V$$

が得られ，これは $\ker L = I$ となるような線型写像である．
これを用いれば，Main Loop は次のように言い換えられる：

FGLM アルゴリズムの一般化 II

Main Loop・改

❶ 入力：単項式 $\mathbf{X}^{\alpha}$

❷ $L(\mathbf{X}^{\alpha})$ を計算する.

❸ $L(\mathbf{X}^{\alpha})$ が B_{lex} の L による像と一次従属か？

Case (a) 一次従属の時：

$$L(\mathbf{X}^{\alpha}) + \sum_{j} c_j L(\mathbf{X}^{\alpha(j)}) = 0$$

となる線形結合を見付けだす．すると，
$I = \ker L$ より

$$g = \mathbf{X}^{\alpha} + \sum_{j} c_j \mathbf{X}^{\alpha(j)} \in I$$

である．この g を G_{lex} の末尾に付け加える.

Case (b) 一次独立の時：$\mathbf{X}^{\alpha}$ を B_{lex} に付け加える.

FGLM アルゴリズムの一般化 III

- ⋆ この手続を Termination Check，Next Monomial と組合せれば，今までと同じことが出来る
- 更に，L を像が有限次元となるような任意の線型写像とすれば，L の核の G_{lex} が求まる！
- 次回：その応用例