Risa/Asir 2006-2007

野呂 正行
(神戸大 · 理)

Contents

1. `f_res` について
2. `nd` 系関数の改良, 機能拡張
3. Modular change of ordering 再び
4. KNOPPIX/Math について

1. f_res について

- 藤原健志 (H17 修士) による終結式パッケージ
  (dense, sparse) multipolynomial resultant, dixon resultant の他, 単独で使える mixed volume 計算関数が実装されている
- マニュアルが修士論文の中にあるだけ
  ソースは既に OpenXM/asir-contrib, OpenXM/src に入っている
  ⇒ マニュアルを整備して, 一般的に使えるようにしたい

Multipolynomial resultant

- Sylvester の resultant

  1 変数の多項式 $f(x)$, $g(x)$ が共通零点を持つ条件を与える, f, g の係数の多項式

- Multipolynomial resultant

  n 変数多項式 $f_1(x_1, \ldots, x_n), \ldots, f_{n+1}(x_1, \ldots, x_n)$ が共通零点を持つ条件を与える, f_i の係数の多項式

さまざまな multipolynomial resultant

- (Dense) Multipolynomial resultant

  F_i $(i = 0, \ldots, n)$ を $(x_0, \ldots, x_n)$ の, 全次数が d_i の斉次式とし, F_i に d_i 次の単項式が全て現れているとした場合の resultant

- Sparse resultant

  多項式毎に指定されたサポートを持つとして, それらが共通零点をもつ条件を与える resultant

- Dixon resultant

  Cayley の方法の拡張による resultant

パッケージ f_res の構成

- OpenXM サーバ ox_sres

  cdd による凸包, 線形計画法計算

  MixedVol による mixed volume 計算

- asir contrib パッケージ f_res

  トップレベル関数 f_res.mres, f_res.sres, f_res.dres および, 付随する関数が実装されている

- texinfo マニュアル (整備中)

  OpenXM/asir-contrib/packages/doc/f_res

インストール, 準備

2007.3.18 以降に取得したソースなら自動的にインストールされる

1. `OpenXM/src/ox_cdd` で make, make install
   `OpenXM/lib/asir` に `ox_sres` ができる.
2. ディレクトリ `f_res` のコピー
   `OpenXM/asir-contrib/packages/src/f_res` を `OpenXM/lib/asir-contrib` にコピー.
3. ファイルのロード (OpenXM 版 asir)
   `load("f_res/f_res.rr")` すれば準備完了.

主な関数 (1)

- `f_res.gmp()`
  gmp 版の `ox_sres` を起動する.
- `f_res.float()`
  浮動小数版の `ox_sres` を起動する.
- `f_res.mres`$(Equations, Vars)$
  Dense multipolynomial resultant の多項式倍を返す.
- `f_res.sres`$(Equations, Vars)$
  Sparse resultant の多項式倍を返す.
- `f_res.dres`$(Equations, Vars)$
  Dixon resultant を返す.

主な関数 (2)

- f_res.conv($[[x_1, y_1], \ldots]$)
  凸包を求める
- f_res.np($[poly_1, \ldots]$)
  Newton polytope を求める
- f_res.msum($[polytope_1, \ldots]$)
  Minkowski sum を求める
- f_res.mvol($[polytope_1, \ldots]$)
  Mixed volume を求める

2. nd 系関数の改良, 機能拡張

- dp 系関数

  古くからある分散多項式計算関係の関数

  効率面で不満 (特に有限体上)

- nd 系関数

  種々の改良 (可変長指数ベクトル, geobucket, reducer history, インライン展開など) を採り入れた, 新実装

  dp 系の機能をすべてはカバーしていなかった.

  $\mathbf{Q}$ 上の F_4 は使い物にならなかった.

nd_gr, nd_gr_trace の改良

- 斉次化 + 頻繁な inter-reduction
- 逐次代数拡大体上のグレブナー基底計算

  newalg() で定義された代数的数を係数に含む

  ⇒ 自動的に代数体上で計算
- 有理関数体上のグレブナー基底計算

  変数リストにない変数は係数体に回される

  (dp_gr_main(), gr() と同じ)

斉次化 + 頻繁な inter-reduction

Risa/Asir における Buchgerger 算法実装の基本部品

- 斉次化 trace 算法
- Content 除去
- Inter-reduction

  一般：グレブナー基底候補生成後

  斉次：全次数が同じ S-多項式の処理が終った後にも

斉次の場合の inter-reduction

S_d = d 次の S 多項式の集合

d が最低のものから処理

$\Rightarrow$ S_d の処理後には, d 次の S 多項式は現れない

$\Rightarrow$ この時点で, d 次以下のグレブナー基底の元が確定

$\Rightarrow$ d 次の基底を inter-reduction して, 置き換えても OK

実は, S_d の処理中に inter-reduction しても OK

$\Rightarrow$ その都度 content が除ける

頻繁な inter-reduction

G_d : S_d から得られる基底

命題

S_d の処理の途中に, 得られた d 次の基底間で任意に inter-reduction を行って, その結果でもとの基底を置き換えても, S_d の処理後の d 次の基底全体は, G_d と同じ線形空間を張る.

(F_4 の正当性と同じ理由による)

逐次代数拡大体上のグレブナー基底計算

- $\mathbf{Q}(\alpha_1, \ldots, \alpha_m)$ 係数の計算

  α_i は $\texttt{newalg}(Poly)$ により定義される.

- 実際には $\mathbf{Q}$ 上で計算

  定義多項式イデアルに添加され, 中間基底も monic 化される

  $\Rightarrow$ 剰余は $\mathbf{Q}$ 上での計算となる

- 頻繁な inter-reduction

  係数膨張が押えられる場合あり

有理関数体上のグレブナー基底計算

- `nd_gr()`, `nd_gr_trace()` の新機能
- heuristic な content 除去

  前処理：ある多項式集合 S で割れるだけ割る

  S = 現れた先頭係数を, 既出の因子で割った商の集合

  さらに, GCD を使って完全に除去

  (要検討)
- 頻繁な inter-reduction

  頻度が問題

nd_gr($Plist, Vlist, Char, Order$)

- $Char = 0$

  $\mathbf{Q}$ 上または $\mathbf{Q}(t_1, \ldots, t_m)$ 上の計算

  (t_i は, $Vlist$ にない変数)

- $Char = p$ $(0 < p < 2^{29})$

  $\mathbf{F}_p$ 上の計算 (有理関数体は未実装)

- $Order$

  0 (grlex), 1 (glex), 2 (lex) および $[[O_1, N_1], \ldots]$ (積順序) のみ

nd_gr_trace($Plist, Vlist, Homo, P, Order$)

$\mathbf{Q}$ 上または $\mathbf{Q}(t_1, \ldots, t_m)$ 上の計算
(t_i は, $Vlist$ にない変数)

- $P < 0$ なら, 候補生成のみ
  使いみち：change of ordering
- $|P| = 1$
  システムが勝手にえらんだ素数を使って modular trace
- $|P| > 1$
  $\mathbf{F}_{|P|}$ 上でのみ modular trace (失敗もあり)

$\mathbf{Q}$ 上の F_4 の新実装

- 論文通りの実装

  ある次数の全 S-poly + reducer から行列を作る

- `nd_f4`

  ある次数の全 S-poly を, その時点での中間基底で reduction した剰余から行列を作る

- `nd_f4_trace` (引数は `nd_gr_trace` と同じ)

  あらかじめ 0 に行きそうな S-poly を trace で除いたあと, 上と同じ操作で行列を作る

  $\Rightarrow \mathbf{Q}$ 上でも `nd_gr_trace` レベルになった

3. Modular change of ordering 再び

- 動機

  trace 算法におけるチェックのコスト低減

- 方法

  別の order でのグレブナー基底を入力にする

  $\Rightarrow$ change of ordering の一般的枠組を提供

- 現状

  12 年前にボツにしたが, 結果は随所で使っている

  (`tolex()`, `tolex_gsl()`, b-関数 ...)

  $\Rightarrow$ 一般化したい

Compatibility

R : 整域, K : 体, $\phi : R \to K$: 全射準同型

$P = \mathrm{Ker}(\phi)$: 素イデアル

ϕ : ϕ の R_P への拡張

定義

$F \subset R[X]$, $\langle F \rangle \subset Q(R)[X]$.

1. ϕ が $(F, <)$ について permissible $\Leftrightarrow$
   $\phi(\mathrm{HC}(f)) \neq 0 \ (\forall f \in F)$ (order 依存)
2. ϕ が F と compatible $\Leftrightarrow \langle \phi(F) \rangle = \phi(\langle F \rangle \cap R[X])$
   (order 非依存だがチェック困難)

Compatibility は任意の GB により判定できる

定理

$G \subset \langle F \rangle \cap R[X]$ を $\langle F \rangle$ の $<$ に関する GB とする. もし ϕ $(G, <)$ について permissible で $\phi(G) \subset \langle \phi(F) \rangle$ ならば

1. $\phi(G)$ は $\langle \phi(F) \rangle$ の GB.
2. ϕ は F と compatible.

Compatible GB candidate

定義

$G \subset \langle F \rangle \cap R[X]$ が $\langle F \rangle$ の $<$ に関する
(ϕ, F)-compatible Gröbner basis basis candidate
$\Leftrightarrow$ ϕ が $(G, <)$ について permissible で $\phi(G)$ が
$\langle \phi(F) \rangle$ の GB

定理

もし R が PID で

1. ϕ が F と compatible
2. G が $I = \langle F \rangle$ の (ϕ, F)-compatible GB candidate

ならば G は I のグレブナー基底.

便利な系：change of ordering

R が PID で

1. $G_0 \subset R[X]$ が $<_0$ に関する I の GB
2. ϕ が $(G_0, <_0)$ について permissible
3. G が $<$ に関する I の (ϕ, G_0)-compatible GB candidate

ならば G は $<$ に関する I のグレブナー基底.

($\phi(G_0) \subset \langle \phi(G_0) \rangle$ は自動的に成立.)

一般化：$\mathbf{Q}(t_1, \ldots, t_m)$ 上の GB

$a_1, \ldots, a_m \in \mathbf{Z}$,

$$\begin{aligned} \phi : R_m = \mathbf{Z}[t_1, \ldots, t_m] &\rightarrow \mathbf{F}_p \\ \phi(h(t_1, \ldots, t_m)) &\mapsto h(a_1, \ldots, a_m) \bmod p \end{aligned}$$

定理

1. $G \subset R_m$ が $<$ に関する $\langle G \rangle \subset \mathbf{Q}(t_1, \ldots, t_m)[X]$ の GB
2. ϕ $(G, <_0)$ について permissible
3. $H \subset R_m \cap \langle G \rangle$ が $<$ に関する $\langle G \rangle$ の (ϕ, G)-compatible GB candidate

$\Rightarrow H$ は $<$ に関する $\langle G \rangle$ のグレブナー基底.

nd_gr_trace(Plist,Vlist,Homo,-P,Order)

trace 実装における, 引数 $Char$ が負の場合

$\Rightarrow$ チェックをしない

使い方

- 入力がグレブナー基底 G_0
- P が G_0 について permissible
- 候補生成が成功

$\Rightarrow$ 結果はチェックするまでもなくグレブナー基底

(by 系)

4. KNOPPIX/Math について

- 2007 年版 DVD

  濱田氏のご好意により, 50 枚配布

- 仮想マシン版

  とりあえず 20 枚程作りました. DVD 版でダウンロードが必要なものもできるだけダウンロード済みです.

  使い方は,

  `http://www.math.kobe-u.ac.jp/vmkm/vmkm-ja.html`

  にあります.