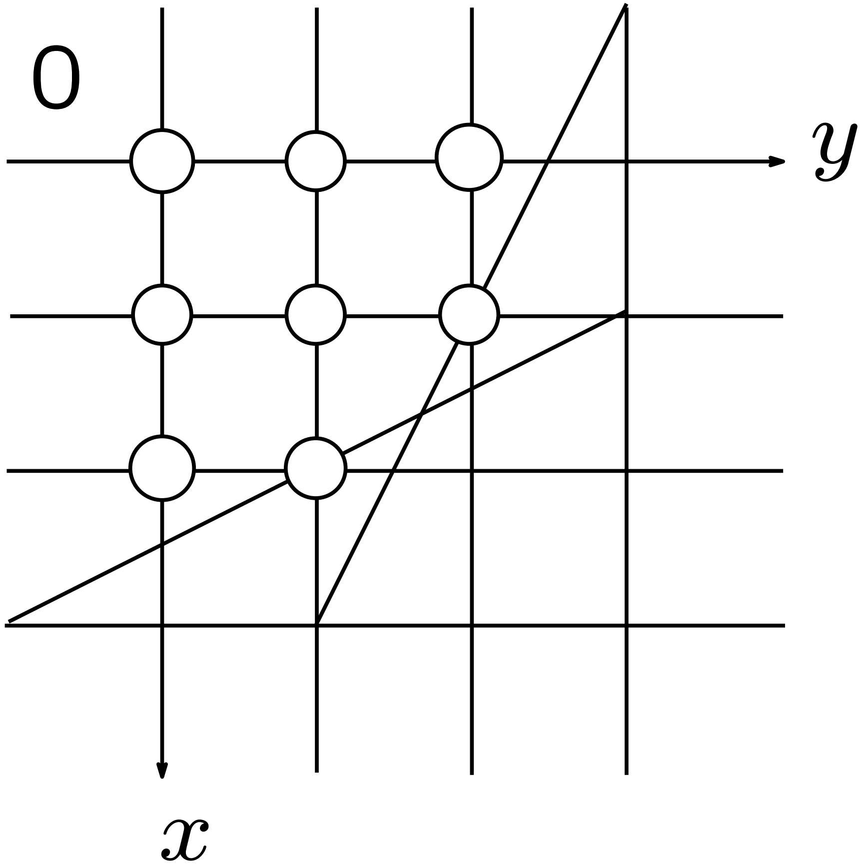
判別式を計算することなしに出現する項を予測する方法

木村 欣司(京都大学大学院情報学研究科)

判別式を計算することなしに，項数を上から抑える方法

凸包の格子内点は，数えられる



$$x \geq 0, y \geq 0, x + 2y \leq 5, 2x + y \leq 5$$

線形不等式により，凸包領域が記述される

凸包領域の格子内点を，数えるソフトウェアが存在する

LattE macchiato
http://www.math.ucdavis.edu/～mkoeppe/latte/

判別式の特徴のまとめ

(1)n方程式の判別式は，主係数も含めて，$2n-2$次の斉次多項式になっている

(2)主係数を1に規格化すると，$n(n-1)$次の斉重多項式になっている

(3)各変数の最大次数は，上限n次まである(より正確には，定数項は$n-1$次まで)

判別式の特徴を線形不等式で表現してみる

３次方程式の判別式の場合

$$
\begin{aligned}
b + c + d &\leq 2 \times 3 - 2 \\
b + 2c + 3d &\leq 3 \times (3 - 1) \\
b + 2c + 3d &\geq 3 \times (3 - 1) \\
0 \leq b \leq 3, 0 \leq c &\leq 3, 0 \leq d \leq 2
\end{aligned}
$$

この凸包の格子内点を数え上げれば，判別式の項数の**上界**が求まる，３次式の場合には，格子内点の数は，5，真の項数は，5であり，上界ではなく，そのものの値が求まる

16次の判別式ではどうか?

いままで説明してきたやり方で凸包を作成し，その格子内点を数え上げると，9016085211である

しかし, 実際の16次の判別式の項数は, 3798697446である

何次の判別式まで，このやり方で正確な項数が計算できるのか?
この緩い上界をどうしたらシャープにできるのか?

(1),(2),(3)のみの特徴により正確に項数を計算できるのは，何次までか?

4次方程式の判別式の場合

$$b + c + d + e \leq 2 \times 4 - 2$$

$$b + 2c + 3d + 4e \leq 4 \times (4 - 1)$$

$$b + 2c + 3d + 4e \geq 4 \times (4 - 1)$$

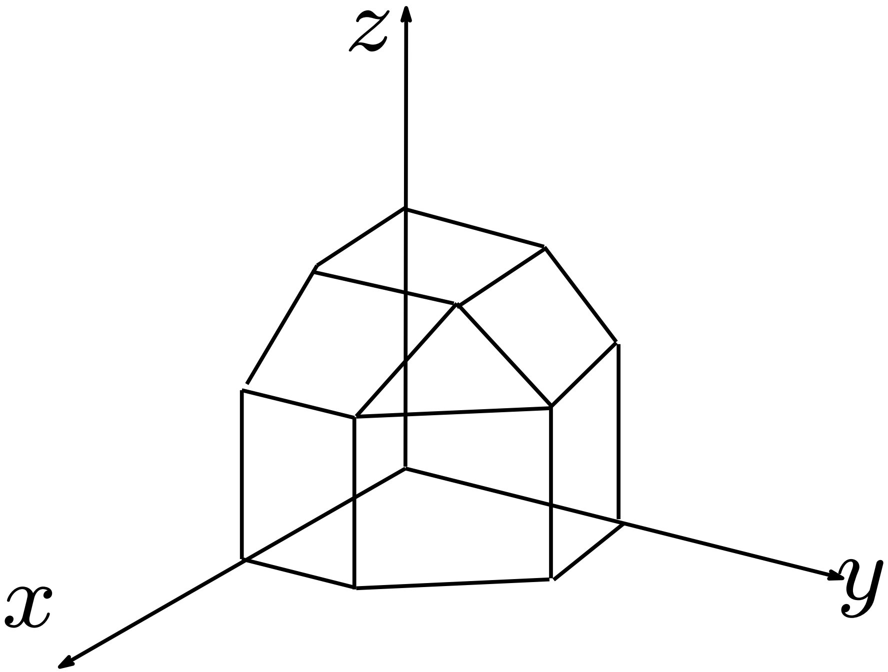
$$0 \leq b \leq 4, 0 \leq c \leq 4, 0 \leq d \leq 4, 0 \leq e \leq 3$$

格子内点の数は，17，真の項数は，16であり，すでに，4次式ですら，真の値ではなく上界になる

緩い上界をシャープにするには，どうしたらよいか?

$$b + c + d + e \leq 2 \times 4 - 2$$

上の条件の代わりに，下のような凸包を用意する



複数の重み(weight)を利用して, それぞれの重みに対して$2^m - 1$(mは変数の数)のtotal degreeのboundを用意すればよい

例えば, $[b, c, d, e] = [1, 2, 3, 4]$と$[b, c, d, e] = [3, 2, 1, 4]$のweightを使う

4次の判別式のLattE macchiatoへの入力ファイル

```
36 5
12 -1 -2 -3 -4
-12 1 2 3 4
0 1 0 0 0
0 0 1 0 0
0 0 0 1 0
0 0 0 0 1
4 -1 0 0 0
8 0 -2 0 0
8 -1 -2 0 0
12 0 0 -3 0
12 -1 0 -3 0
12 0 -2 -3 0
12 -1 -2 -3 0
12 0 0 0 -4
12 -1 0 0 -4
12 0 -2 0 -4
```

12 -1 -2 0 -4
12 0 0 -3 -4
12 -1 0 -3 -4
12 0 -2 -3 -4
12 -1 -2 -3 -4
12 -3 0 0 0
8 0 -2 0 0
12 -3 -2 0 0
4 0 0 -1 0
12 -3 0 -1 0
8 0 -2 -1 0
12 -3 -2 -1 0
12 0 0 0 -4
20 -3 0 0 -4
12 0 -2 0 -4
20 -3 -2 0 -4
12 0 0 -1 -4
20 -3 0 -1 -4
12 0 -2 -1 -4
20 -3 -2 -1 -4

効果の検証

4次の判別式においては，格子内点の数は16で，真の値を計算できる

5次の判別式においても，格子内点の数は59で，真の値を計算できる

しかし，6次の判別式においては，格子内点の数は251になる，真の値は246項である

251と246の差は?

$$x^6 + bx^5 + cx^4 + dx^3 + ex^2 + fx + g$$

$$c^2d^4ef^2$$
$$c^2d^4e^2g$$
$$b^2d^4f^2g$$
$$b^2cd^4ef^2$$
$$b^2cd^4e^2g$$

この5項を凸包から排除するアイディアは，いまのところない

weightの種類を増やせばよいのでは?

現在の2種類のweightの場合，計算時間はどうか?

次数	判別式の計算時間	項数の上界計算
4	0.001sec	0.04sec
5	0.001sec	0.21sec
6	0.002sec	1.68sec

判別式の計算時間よりも，項数の上界計算のほうが長くなる!

最も重要な注意

判別式(行列式)そのものを計算する方法である「多項式補間法」が，項数の上界9016085211を必要とすることはまったくない

16次の判別式の項数にどれだけ迫れるか?

実時間で計算が終わる程度に複雑な線形不等式で，16次の判別式の項数を抑えてみる

```
55 17
240 -1 -2 -3 -4 -5 -6 -7 -8 -9 -10 -11 -12 -13 -14 -15 -16
-240 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
30 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
16 -1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
16 0 -1 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

16 0 0 -1 0 0 0 0 0 0 0 0 0 0 0 0 0
16 0 0 0 -1 0 0 0 0 0 0 0 0 0 0 0 0
16 0 0 0 0 -1 0 0 0 0 0 0 0 0 0 0 0
16 0 0 0 0 0 -1 0 0 0 0 0 0 0 0 0 0
16 0 0 0 0 0 0 -1 0 0 0 0 0 0 0 0 0
16 0 0 0 0 0 0 0 -1 0 0 0 0 0 0 0 0
16 0 0 0 0 0 0 0 0 -1 0 0 0 0 0 0 0
16 0 0 0 0 0 0 0 0 0 -1 0 0 0 0 0 0
16 0 0 0 0 0 0 0 0 0 0 -1 0 0 0 0 0
16 0 0 0 0 0 0 0 0 0 0 0 -1 0 0 0 0
16 0 0 0 0 0 0 0 0 0 0 0 0 -1 0 0 0
16 0 0 0 0 0 0 0 0 0 0 0 0 0 -1 0 0
16 0 0 0 0 0 0 0 0 0 0 0 0 0 0 -1 0
15 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 -1
32 -1 -2 0 0 0 0 0 0 0 0 0 0 0 0 0 0
32 0 0 0 0 0 0 0 0 0 0 0 0 0 -2 -1 0
48 -1 -2 -3 0 0 0 0 0 0 0 0 0 0 0 0 0
48 0 0 0 0 0 0 0 0 0 0 0 0 -3 -2 -1 0
64 -1 -2 -3 -4 0 0 0 0 0 0 0 0 0 0 0 0
64 0 0 0 0 0 0 0 0 0 0 0 -4 -3 -2 -1 0
80 -1 -2 -3 -4 -5 0 0 0 0 0 0 0 0 0 0 0
80 0 0 0 0 0 0 0 0 0 0 -5 -4 -3 -2 -1 0
96 -1 -2 -3 -4 -5 -6 0 0 0 0 0 0 0 0 0 0
96 0 0 0 0 0 0 0 0 0 -6 -5 -4 -3 -2 -1 0
112 -1 -2 -3 -4 -5 -6 -7 0 0 0 0 0 0 0 0 0
112 0 0 0 0 0 0 0 0 -7 -6 -5 -4 -3 -2 -1 0
128 -1 -2 -3 -4 -5 -6 -7 -8 0 0 0 0 0 0 0 0

```
128 0 0 0 0 0 0 0 -8 -7 -6 -5 -4 -3 -2 -1 0
144 -1 -2 -3 -4 -5 -6 -7 -8 -9 0 0 0 0 0 0 0
144 0 0 0 0 0 0 -9 -8 -7 -6 -5 -4 -3 -2 -1 0
160 -1 -2 -3 -4 -5 -6 -7 -8 -9 -10 0 0 0 0 0 0
160 0 0 0 0 0 -10 -9 -8 -7 -6 -5 -4 -3 -2 -1 0
176 -1 -2 -3 -4 -5 -6 -7 -8 -9 -10 -11 0 0 0 0 0
176 0 0 0 0 -11 -10 -9 -8 -7 -6 -5 -4 -3 -2 -1 0
```

計算結果は,

```
Total Unimodular Cones: 397541805

Maximum number of simplicial cones in memory at once: 55

****  The number of lattice points is: 7762102998  ****

Total time: 688410 sec
```

LattE macchiatoの実行環境

CPU:Intel Xeon X5570(4Core)*2 Mem:32G

OS:Fedora13 GCC:4.4.5

LattE macchiato:

latte-for-tea-too-1.2-mk-0.9.3

Serial

LattE macchiatoの問題とその克服

LattE macchiatoは，凸包の内点格子の個数を数えることができるのみであり，凸包の内点格子をリストアップできない

もし，凸包の内点格子をリストアップしたいのであれば，
http://typo.zib.de/opt-long_projects/Software/Porta/index.html