Gröbner-Fächer für lineare Codes

Daniel Rembold

Technische Universität Hamburg Harburg

daniel.rembold@tuhh.de

August 27, 2014

Inhaltsverzeichnis

Motivation

- Gröbnerbasen
-

Monome

Monom

- Produkt von Variablen über ein endliches Feld $\mathbb{K}[X_1, X_2, \ldots, X_n]$
- Schreibweise $m = X_1^{u_1} X_2^{u_2} \cdots X_n^{u_n}$ und $u_i \in \mathbb{N}_0$

Grad eines Monoms: $deg(m) = \sum_{i=1}^{n} u_i$.

Termordnung

Termordnung $>$

- Relation $>$ zu der Menge von allen Monomen in $\mathbb{K}[X_1, X_2, \ldots, X_n]$

Termordnung

- Lexikographische Ordnung $>_{lex}$
- grad $>_{grlex}$
- Ordnung mit Gewichtsvektor $c = (c_1, \ldots, c_n) \in \mathbb{R}_+^n$

Leitterm

Leitterm $\mathrm{LT}(f)$

- Polynom $p \in \mathbb{K}[X_1, X_2, \ldots, X_n]$ besitzt Term höchster Ordnung in Bezug auf $>$

Beispiel

Sei $f = x^2 + 3xyz + y^3$

- lex-Order : $f = \underline{x^2} + 3xyz + y^3$
- grlex-Order : $f = \underline{3xyz} + y^3 + x^2$
- $(1, 2, 1)$: $f = \underline{y^3} + 3xyz + x^2$

Ideale

Ideal

- Kollektion von Polynomen $f_1, \ldots, f_s$:

$$\langle f_1, \ldots, f_s \rangle = \left\{ \sum_{i=1}^{s} h_i f_i \mid h_1, \ldots, h_s \in \mathbb{K}[X_1, \ldots, X_n] \right\}.$$

Beispiel

Sei $I = \langle f_1, f_2 \rangle = \langle x^2 + y, x + y + 1 \rangle$ und $f = x^2y + x^2 + y^2 + xy + x$
Dann gilt $f = y \cdot f_1 + x \cdot f_2, \ f \in I$.

Divisionsalgorithmus (1)

- Notwendig zum Lösen des Idealzugehörigkeitsproblems

Ziel des Algorithmus

Polynom g durch Ideal $I = \langle f_1, \ldots, f_s \rangle$ teilen, so dass
$g = a_1 f_1 + \ldots + a_s f_s + r, \quad a_i, g, I, r \in \mathbb{K}[X_1, \ldots, X_n]$

Sei Polynom p und Ideal I

- Wenn p % I = 0, dann gilt $p \in I$

Divisionsalgorithmus (2)

Algorithm 1 Divisionsalgorithmus (Header)

Require: Basis $I = \langle f_1, \ldots, f_m \rangle$ of nonzero polynomials **and** a fixed termorder LT

Ensure: $r = 0$ **or** none of the terms in r are divisible by $LT_{\leq}(f_1), \ldots, LT_{\leq}(f_m)$

- Reihenfolge der Polynome in I beeinflusst Ergebnis
- $r \neq 0$ möglich, obwohl $p \in I$

Gröbnerbasis (1)

Gröbnerbasis

Sei Termordnung $>$ und Ideal I, dann hat Gröbnerbasis $G = \{f_1, \ldots, f_m\}$ (in Bezug auf $>$) von I die Eigenschaft:

- Von jedem Polynom $p \in I$ ist $\mathrm{LT}_>(p)$ teilbar durch $\mathrm{LT}_>(f_i)$

- Divisionsrest eindeutig bestimmt und unabhängig von Reihenfolge
- Gröberbasis aus jedem Ideal mit Hilfe des Buchberger-Algorithmus und einer festen Termordnung

Gröbnerbasis (2)

- Gröbernbasen sind nicht eindeutig
- Reduzierte Gröbnerbasen jedoch eindeutig für Ideale

Reduzierte Gröbnerbasis

Alle Leitterme von Gröbnerbasis G in Bezug auf Termordnung $>$ monisch und relativ prim zueinander

Gröbner-Fächer

- Unendlich viele Termordnungen, endlich viele reduzierte Gröbnerbasen

Gröbner-Fächer

- Vielflächiges Komplex welches Kegel im $\mathbb{R}^n_+$ enthält
- Flächen werden durch Ebenenungleichungen der Polynome bestimmt

Beispiel Gröbner-Fächer (1)

Sei

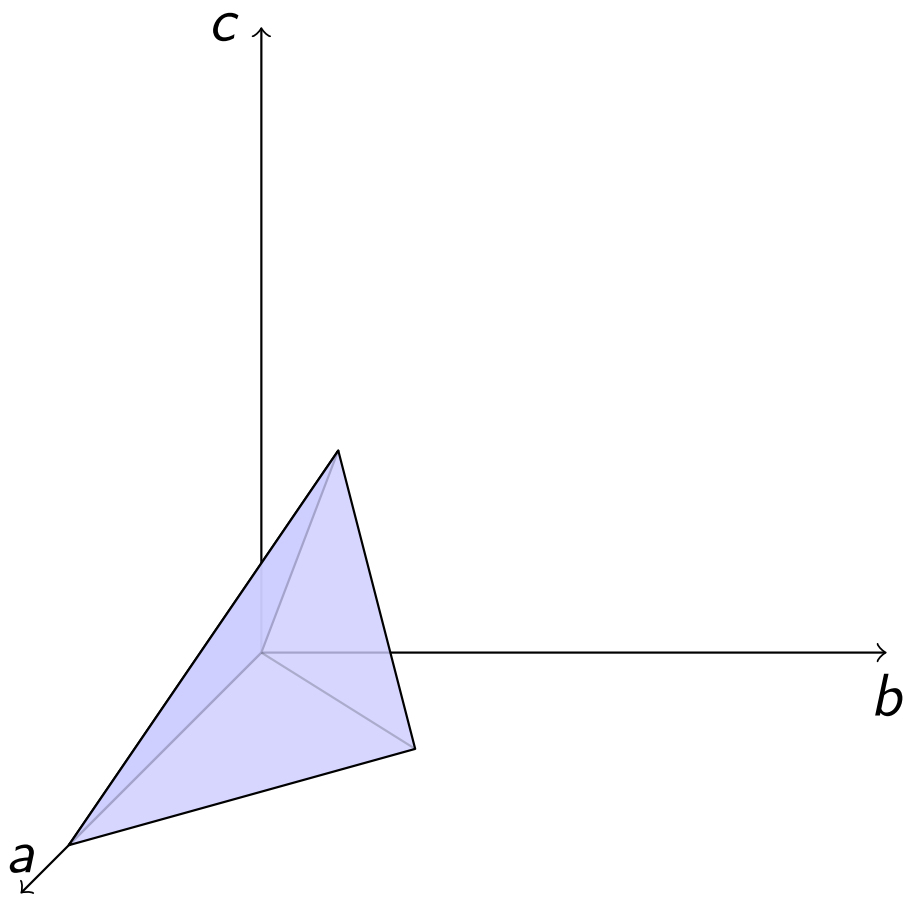
1. $I = \langle x^2 - z,\ y - x \rangle \in \mathbb{K}[x, y, z]$
2. $G_{>_{lex}} = \{\underline{y^2} - z,\ \underline{x} - y\}$
3. $\mathbf{w} = (a, b, c) \in \mathbb{R}^3_+$

- $w \in G_{>_{lex}}$ genau dann, wenn

  1. $(0, 2, 0) \cdot (a, b, c) \geq (0, 0, 1) \cdot (a, b, c) \vee 2b \geq c$
  2. $(1, 0, 0) \cdot (a, b, c) \geq (0, 1, 0) \cdot (a, b, c) \vee a \geq b$

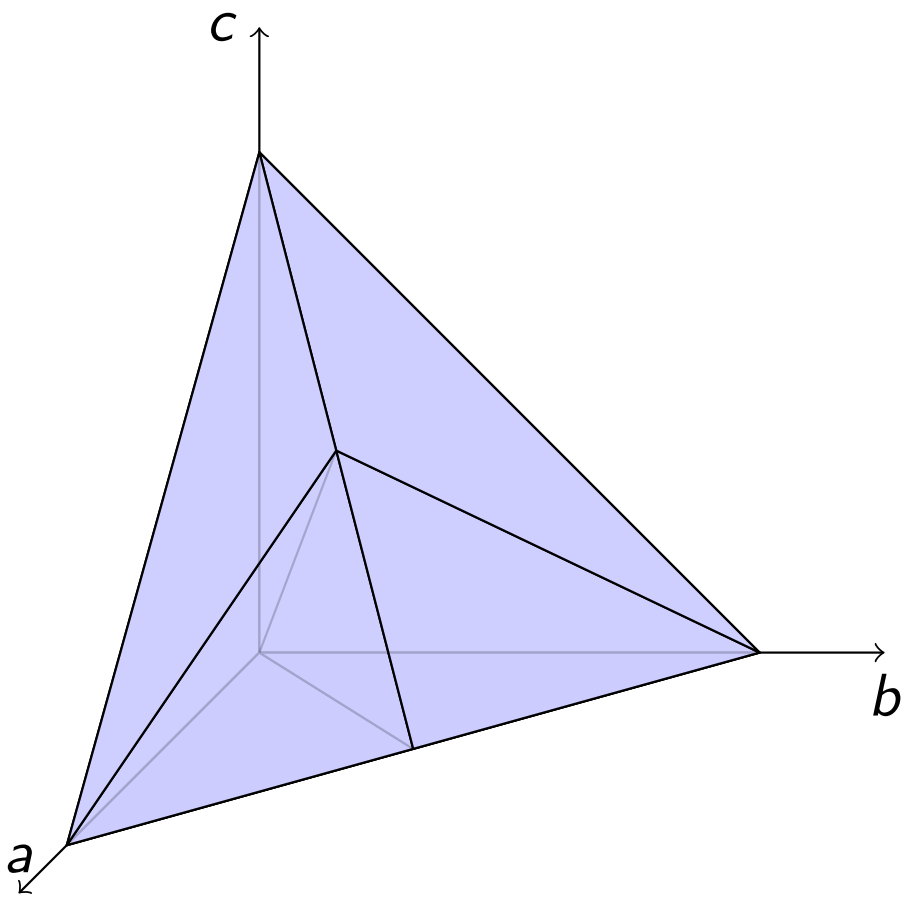
Beispiel Gröbner-Fächer (2)

Figure: Gröbner-Kegel für $G_{>_{lex}}$



Beispiel Gröbner-Fächer (3)

Figure: Kompletter Gröbner-Fächer



Torische Ideale

- Obermenge von Code-Idealen
- Besteht nur aus Binomen

Torisches Ideal

Gegeben seien $A = [a_1, \ldots, a_n] \in \mathbb{Z}^{d \times n}$ und $u \in \mathbb{Z}^n$ zerlegbar in u^+ und u^-, dann ist das torische Ideal I_A definiert durch

$$\mathbf{I}_A = \langle \mathbf{x}^{u^+} - \mathbf{x}^{u^-} \mid u \in ker\,(A) \rangle.$$

BLABLA

Performancemessung auf ATI und NVIDIA

Vergleich

Mögliche Verbesserungen

Quellen

University of Bristol

Optimizing OpenCL performance

http://www.cs.bris.ac.uk/home/simonm/workshops/OpenCL_lecture3.pdf

NVIDIA

OpenCL SDK Code Samples

https://developer.nvidia.com/opencl

Vasily Volkov (UC Berkeley , September 22, 2010)

Better Performance at Lower Occupancy

http://www.cs.berkeley.edu/~volkov/volkov10-GTC.pdf

Vielen Dank für eure Aufmerksamkeit!