Groebner Basics

Talk given at the Special Semester on Groebner Bases Linz 2006

Hans-Gert Gräbe, Dept. Computer Science, Univ. Leipzig, Germany
`http://www.informatik.uni-leipzig.de/~graebe`

February 27, 2006

1 Basics and Notations

$R = k[x_1, \ldots, x_n]$ Polynomring über einem Körper k, K dessen algebraischer Abschluss

$\mathbb{A}^n := \{(a_1, \ldots, a_n) \ : \ a_i \in K\}$ der n-dim. *affine Raum* (über K)

$B = \{f_1, \ldots, f_s\} \subset S$ (endliches) System von Polynomen

$V = V(B) := \{(a_1, \ldots, a_n) \in \mathbb{A}^n \ : \ f_i(\mathbf{a}) = 0 \ \forall\, i\}$ deren gemeinsame Nullstellenmenge.

Mengen $V \subset \mathbb{A}^n$, die sich auf diese Weise darstellen lassen, heißen *affine Varietäten*.

$I = Id(B)$ das von B erzeugte Ideal in S

Dann gilt $V(B) = V(Id(B))$

$Id(V) := \{f \in R \ : \ f(\mathbf{a}) = 0 \ \forall\, \mathbf{a} \in V\}$ Menge der auf $V \subset \mathbb{A}^n$ verschwindenden polynomialen Funktionen.

Als *Monom* bezeichnet man ein Potenzprodukt

$$\mathbf{x}^\alpha = x_1^{\alpha_1} \cdot \ldots \cdot x_n^{\alpha_n}, \quad \alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{N}^n$$

Die Menge aller Monome

$$T = T(\mathbf{x}) = T(x_1, \ldots, x_n) = \{\mathbf{x}^\alpha \ : \ \alpha \in \mathbb{N}^n\}$$

ist eine Halbgruppe mit $1 = \mathbf{x}^0$ bzgl. der üblichen Multiplikation, das *Termmonoid*. Damit ist auch eine Teilbarkeitsrelation auf den Monomen definiert.

Als *Polynom* in $x_1, \ldots, x_n$ über k bezeichnet man jede endliche k-lineare (mit $c_\alpha \in k$) Kombination von Monomen

$$f = \sum c_\alpha \mathbf{x}^\alpha.$$

Die Darstellung $f = \sum c_\alpha \mathbf{x}^\alpha$ kann in den meisten CAS aus allgemeineren Darstellungen polynomialer Ausdrücke durch `expand` gewonnen werden.

Diese Darstellung ist eindeutig, d.h. eine kanonische Form für Polynome $f \in R$, wenn für die Koeffizienten, also die Elemente aus A, eine solche kanonische Form existiert und die Reihenfolge der Summanden festgelegt ist. Zur Festlegung der Reihenfolge definiert man gewöhnlich eine totale Ordnung auf $T(\mathbf{x})$.

Als *distributive Darstellung* eines Polynoms $f \in R$ bzgl. einer solchen Ordnung bezeichnet man eine Darstellung $f = \sum_a c_a \, \mathbf{x}^a$, in welcher die Summanden paarweise verschiedene Terme enthalten, diese in fallender Reihenfolge angeordnet sind und die einzelnen Koeffizienten in ihre kanonische Form gebracht wurden. In dieser Darstellung ist die Addition von Polynomen besonders effizient ausführbar. Ist die gewählte Ordnung darüberhinaus *monoton*, d.h. gilt

$$s < t \Rightarrow s \cdot u < t \cdot u \quad \text{für alle } s, t, u \in T(\mathbf{x}),$$

so kann man auch die Multiplikation recht effektiv ausführen, da dann beim gliedweisen Multiplizieren einer geordneten Summe mit einem Monom die Summanden geordnet bleiben. Ordnungen mit dieser Zusatzeigenschaft bezeichnet man als *Termordnungen*. Oft werden als Termordnungen nur wohlfundierte Ordnungen dieser Art bezeichnet.

Beispiele:

Lexikographische Ordnung (lex) auf $T(\mathbf{x})$ bzgl. $x_1 > x_2 > \ldots > x_n$

$$x_1^{a_1} x_2^{a_2} \cdot \ldots \cdot {x_n}^{a_n} >_{\text{lex}} x_1^{b_1} x_2^{b_2} \cdot \ldots \cdot {x_n}^{b_n}$$
$$\Leftrightarrow \begin{cases} a_1 > b_1 & \text{oder} \\ a_1 = b_1 & \text{und } x_2^{a_2} \cdot \ldots \cdot {x_n}^{a_n} >_{\text{lex}} x_2^{b_2} \cdot \ldots \cdot {x_n}^{b_n} \end{cases}$$

Revers lexikographische Ordnung (revlex) auf $T(\mathbf{x})$ bzgl. $x_1 < x_2 < \ldots < x_n$

$$x_1^{a_1} \cdot \ldots \cdot {x_{n-1}}^{a_{n-1}} {x_n}^{a_n} >_{\text{revlex}} x_1^{b_1} \cdot \ldots \cdot {x_{n-1}}^{b_{n-1}} {x_n}^{b_n}$$
$$\Leftrightarrow \begin{cases} a_n < b_n & \text{oder} \\ a_n = b_n & \text{und } x_1^{a_1} \cdot \ldots \cdot {x_{n-1}}^{a_{n-1}} >_{\text{revlex}} x_1^{b_1} \cdot \ldots \cdot {x_{n-1}}^{b_{n-1}} \end{cases}$$

Gradordnung auf $T(\mathbf{x})$ (bzgl. der Standardgraduierung)

$$x_1^{a_1} \cdot \ldots \cdot {x_n}^{a_n} >_{\text{deg}xxx} x_1^{b_1} \cdot \ldots \cdot {x_n}^{b_n}$$
$$\Leftrightarrow \begin{cases} \deg(\mathbf{a}) > \deg(\mathbf{b}) & \text{oder} \\ \deg(\mathbf{a}) = \deg(\mathbf{b}) & \text{und } x_1^{a_1} \cdot \ldots \cdot {x_n}^{a_n} >_{\text{xxx}} x_1^{b_1} \cdot \ldots \cdot {x_n}^{b_n} \end{cases}$$

Hier ist *xxx* eine andere Termordnung, nach welcher Terme gleichen Grades geordnet werden. Wichtige Gradordnungen sind insbesondere die *gradweise lexikographische* (deg-lex) und die *gradweise revers lexikographische* (deg-revlex) Termordnung.

Als *Wohlordnung* oder *noethersche Ordnung* bezeichnet man eine totale Ordnung $(T, <)$, in der eine der beiden äquivalenten Bedingungen gilt:

(a) Jede Teilmenge $M \subset T$ hat ein kleinstes Element.

(b) Jede (echt) absteigende Kette $t_1 > t_2 > \ldots$ in T ist endlich.

Während die lexikographische und jede Gradordnung Wohlordnungen sind, gilt dies für die (rein) revers-lexikographische Ordnung nicht: $x_1 > x_1^2 > x_1^3 > \ldots$ ist für diese Termordnung eine unendliche absteigende Kette von Termen.

Theorem 1 *Eine Termordnung $(T(\mathbf{x}), >)$ ist genau dann eine Wohlordnung, wenn gilt*

(c) $m > 1$ *für alle* $m \in T, m \neq 1$.

Sei $0 \neq f(\mathbf{x}) = \sum_{i=0}^{N} c_i \mathbf{x}^{\alpha_i} \in R$ ein Polynom, so dass in der fixierten Termordnung $\mathbf{x}^{\alpha_i} > \mathbf{x}^{\alpha_j}$ für $i < j$ gilt. Bezeichne weiter $T(f) := \{\mathbf{x}^{\alpha_i},\, i = 0, \ldots, N\}$ die Menge der in der Darstellung von f auftretenden Terme. Dann können wir die folgenden Begriffe definieren:

- den Leitterm $lt(f) := \mathbf{x}^{\alpha_0}$,
- den Leitkoeffizienten $lc(f) := c_0$,
- das Leitmonom $lm(f) := lc(f) \cdot lt(f)$,
- das Reduktum $red(f) := f - lm(f)$.

2 Characterization of Term Orderings

Mit $\widetilde{T} = \{\mathbf{x}^{\alpha} \,:\, \alpha \in \mathbb{Z}^n\}$ bezeichnen wir die Gruppe der *verallgemeinerten Terme*, deren Exponenten beliebig ganzzahlig sein können.

(1) Jede Termordnung auf T kann man eindeutig auf $\widetilde{T}$ ausdehnen:

Für $\alpha, \alpha', \beta, \beta' \in \mathbb{N}^n$ setzen wir

$$\mathbf{x}^{\alpha-\alpha'} < \mathbf{x}^{\beta-\beta'} \Leftrightarrow \mathbf{x}^{\alpha+\beta'} < \mathbf{x}^{\alpha'+\beta}$$

Remark: *Partial* monotone orderings on $\mathbb{N}^n$ can be extended iff the cancellation rule holds

$$\mathbf{x}^{\alpha}\mathbf{x}^{\gamma} < \mathbf{x}^{\beta}\mathbf{x}^{\gamma} \Rightarrow \mathbf{x}^{\alpha} < \mathbf{x}^{\beta}.$$

For total orderings the cancellation rule follows from the monotonicity property.

(2) Dann gilt

$$\mathbf{x}^{\alpha} < \mathbf{x}^{\beta} \Leftrightarrow 1 < \mathbf{x}^{\beta-\alpha},$$

so dass die Termordnung durch ihren *Positivkegel* $C_+ = \left\{\mathbf{x}^{\alpha} \in \widetilde{T} \,:\, \mathbf{x}^{\alpha} > 1\right\}$ bestimmt wird.

(3) Da die Ordnung eine lineare Ordnung ist, ist der Positivkegel ein Halbraum, der durch ein lineares Funktional $w \in (\mathbb{Z}^n)^* \cong \mathbb{R}^n$ beschrieben werden kann, so dass für $\alpha \in \mathbb{Z}^n$ gilt

$$w(\alpha) > 0 \Rightarrow \mathbf{x}^{\alpha} > 1$$

und folglich auch (wegen $w(-\alpha) = -w(\alpha)$)

$$w(\alpha) < 0 \Rightarrow \mathbf{x}^{\alpha} < 1$$

Wir setzen kurz auch $w(\mathbf{x}^{\alpha}) = w(\alpha)$.

(4) Einzig über Terme $\mathbf{x}^{\alpha}$ mit $w(\alpha) = 0$ kann allein aus diesem *Gewichtsvektor* w keine Aussage getroffen werden. Diese liegen jedoch in einem linearen Unterraum von $\mathbb{Z}^n$ und wir

können für diese Gitterpunkte dieselbe Argumentation mit einem weiteren Gewichtsvektor wiederholen.

(5) Jeder solche Gewichtsvektor ist durch den Zeilenvektor $(w(x_i), i = 1, \ldots, n)$, die *Gewichte der Variablen*, eindeutig bestimmt. Beschränkt man sich auf rationale Gewichte, so kann man alle Gewichte sogar als ganzzahlig annehmen, da sich die durch $w(\alpha) = 0$ beschriebene Gitterebene durch Skalieren nicht ändert. Durch Skalierung auf die Länge 1 kann man die Gewichtsvektoren mit Punkten auf der Sphäre S^{n-1} identifizieren und hat damit auch eine genaue Fassung des Begriffs »nahe beieinander liegender« Termordnungen.

Definition 1 *Jede Termordnung mit dem Gewichtsvektor w bezeichnen wir als* Termordnung, die w verfeinert.

(6)

Theorem 2 (Charakterisierungssatz für Termordnungen)
Jede Termordnung lässt sich durch eine Folge von Gewichtsvektoren $w_1, w_2, \ldots, w_k \in \mathbb{R}^n$ beschreiben, wobei gilt

$$\mathbf{x}^\alpha > 1 \Leftrightarrow \exists j < k : w_i(\alpha) = 0 \textit{ für } i \leq j \textit{ und } w_{j+1}(\alpha) > 0$$

Hierbei ist w_1 eindeutig bestimmt, während w_j um Vielfache von $w_i,\ i < j$, abgeändert werden kann.

(7) Jede Termordnung lässt sich damit als *Matrix-Termordnung* darstellen, indem die Gewichte der Variablen bzgl. der w_i als Zeilen einer Matrix notiert werden.

Eine Termordnung ist offensichtlich genau dann eine Wohlordnung, wenn der erste Eintrag verschieden Null in jeder Spalte der Gewichtsmatrix positiv ist.

Die Matrizen für die oben beschriebenen noetherschen Termordnungen sind

$$>_{\text{lex}}: \begin{pmatrix} 1 & 0 & \ldots & 0 \\ 0 & 1 & \ldots & 0 \\ & & \ldots & \\ 0 & 0 & \ldots & 1 \end{pmatrix} \quad >_{\text{deglex}}: \begin{pmatrix} 1 & 1 & \ldots & 1 & 1 \\ 1 & 0 & \ldots & 0 & 0 \\ 0 & 1 & \ldots & 0 & 0 \\ & & \ldots & & \\ 0 & 0 & \ldots & 1 & 0 \end{pmatrix} \quad >_{\text{degrevlex}}: \begin{pmatrix} 1 & 1 & \ldots & 1 & 1 \\ 0 & 0 & \ldots & 0 & -1 \\ 0 & 0 & \ldots & -1 & 0 \\ & & \ldots & & \\ 0 & -1 & \ldots & 0 & 0 \end{pmatrix}$$

Beispiele mit CoCoA: Standardordnung ist `degrevlex`, andere Ordnungen können durch Kürzel vereinbart werden. Interne Darstellung erfolgt offensichtlich als Matrixordnung.

```
Use R ::= Q[x,y,z];
Ord(R);
Mat[
  [1, 1, 1],
  [0, 0, -1],
  [0, -1, 0]
]

Use S::=Q[x,y,z],Lex;
Ord(S);
```

```
Mat[
  [1, 0, 0],
  [0, 1, 0],
  [0, 0, 1]
]
```

(8) Ist $\Sigma \subset \widetilde{T} \setminus \{1\}$ eine endliche Menge verallgemeinerter Terme, so können wir nach den ersten Gewichtsvektoren aller Termordnungen fragen, in welchen alle Terme aus Σ positiv sind. Genauer betrachten wir die Menge

$$W_\Sigma = \{w \in \mathbb{R}^n \ : \ \forall \mathbf{x}^\alpha \in \Sigma \ w(\alpha) > 0\} = \bigcap_{\mathbf{x}^\alpha \in \Sigma} \{w \in \mathbb{R}^n \ : \ w(\alpha) > 0\}$$

This is a finite (as Σ is finite) intersection of open halfspaces and thus either empty or an open cone and hence n-dimensional.

Die entsprechenden Gewichtsvektoren bilden also ebenfalls einen Kegel im $\mathbb{R}^n = (\mathbb{Z}^n)^*$, welcher dual zum Kegel ist, der von den Exponenten der $\mathbf{x}^\alpha \in \Sigma$ aufgespannt wird.

Für $\Sigma = \{x_1, \ldots, x_n\}$ bekommt man genau die noetherschen Termordnungen heraus. Der Gewichtsvektor $(11 \ldots 1)$ der Gradordnungen liegt im Inneren dieses Kegels, die Gewichtsvektoren der lexikographischen Ordnungen (bzgl. verschiedener Variablenordnungen) auf dessen Rand.

3 PP-Ideals and Monoid Ideals. Dickson Lemma

Definition 2 *Ein Ideal $I \subset R$ heißt* Potenzproduktideal *(PP ideal), wenn mit $f = \sum c_\alpha \mathbf{x}^\alpha \in I$ auch alle Potenzprodukte $\mathbf{x}^\alpha, c_\alpha \neq 0$ zu I gehören.*

Offensichtlich besitzt jedes Ideal mit dieser Eigenschaft eine Basis aus Potenzprodukten. Umgekehrt ist auch ein von Potenzprodukten erzeugtes Ideal ein PP-Ideal in diesem Sinne.

Die Menge aller in einem PP-Ideal enthaltenen Potenzprodukte bildet ein *Monoidideal*, d.h. eine Teilmenge $\Sigma \subset T$ mit

$$\Sigma \cdot T := \{\mathbf{x}^\alpha \cdot \mathbf{x}^\beta \ : \ \mathbf{x}^\alpha \in \Sigma, \mathbf{x}^\beta \in T\} \subset \Sigma.$$

Beispiel: $I = Id(x^4y^2, x^3y^4, x^2y^5)$. Grafische Darstellung im $\mathbb{N}^2$. Wir schreiben $\Sigma = \Sigma(I)$ für das zugehörige Monoidideal. Dieses besteht aus genau den Termen, die (in T) durch wenigstens eines der Basismonome teilbar sind.

Definition 3 *Eine Teilmenge $\Sigma_0 = \{\mathbf{x}^{a_1}, \ldots, \mathbf{x}^{a_m}\}$ eines Monoidideals Σ bezeichnet man als* Basis*, wenn $\Sigma_0 \cdot T = \Sigma$ gilt und als* Minimalbasis*, wenn Σ_0 minimal bzgl. Inklusion mit dieser Eigenschaft ist.*

Theorem 3 *Jedes Monoidideal $\Sigma \subset T$ hat eine eindeutig bestimmte Minimalbasis. Diese besteht genau aus den $\mathbf{x}^\alpha \in \Sigma$, die minimal in Σ bzgl. der Teilbarkeitsrelation sind, d.h. für die*

$$\mathbf{x}^\beta \in \Sigma, x^\beta \,|\, \mathbf{x}^\alpha \ \Rightarrow \ x^\beta = \mathbf{x}^\alpha$$

gilt. Für diese Menge schreiben wir $Gen(\Sigma)$.

Theorem 4 *(Dickson-Lemma) Jedes Monoidideal $\Sigma \subset T := T(x_1, \ldots, x_n)$ besitzt eine endliche Basis.*

4 Normal Forms

Mit der Fixierung eines ≫größten≪ Monoms können wir jedes Polynom f als algebraische Ersetzungsregel auffassen, die monomiale Vielfache des Leitterms $lt(f)$ durch geeignete monomiale Vielfache des Reduktums $red(f)$ ersetzt. Genauer gesagt lautet die abzuleitende Ersetzungsregel

$$lt(f) \mapsto -lc(f)^{-1}\, red(f).$$

Entsprechend erhalten wir für eine endliche Menge $B = \{f_1, \ldots, f_m\}$ von Polynomen ein System von Ersetzungsregeln.
Beispiel: $B_1 = \{f_1 = x^2 + xy + y^2, f_2 = xz + yz, f_3 = y^3 - z^3\}$ liefert (bzgl. $<_{lex}$) das Ersetzungssystem

$$x^2 \mapsto -xy - y^2,\ xz \mapsto -yz,\ y^3 \mapsto z^3.$$

Wenden wir die Regeln in der genannten Reihenfolge auf das Polynom $g = x^2y^2 + x^2z^2 + y^2z^2$ an, so erhalten wir nacheinander

$$\begin{aligned} g &\mapsto x^2z^2 - xy^3 - y^4 + y^2z^2 \mapsto -xy^3 - xyz^2 - y^4 \mapsto -xyz^2 - xz^3 - y^4 \\ &\mapsto -xz^3 - y^4 + y^2z^2 \mapsto -y^4 + y^2z^2 + yz^3 \mapsto y^2z^2 \end{aligned}$$

```
Use R::=Q[x,y,z],Lex;
B1:=[x^2+xy+y^2,xz+yz,y^3-z^3];
G:=x^2y^2+x^2z^2+y^2z^2;
NR(G,B1);
y^2z^2
------------------------------
```

Das aus B abgeleitete Ersetzungssystem erlaubt es also, alle Terme aus dem Monoidideal

$$\Sigma(B) := \{x^\alpha \ :\ \exists\, f \in B :\ lt(f) \,|\, x^\alpha\}$$

durch eine Linearkombination ≫kleinerer≪ Terme zu ersetzen. Diese Terme bezeichnen wir deshalb auch als *Nichtstandardterme*, die verbleibenden Terme $T(X) \setminus \Sigma(B)$ dagegen als die *Standardterme* bzgl. B. Das von $\Sigma(B)$ erzeugte PP-Ideal bezeichnen wir mit $Lt(B)$.

Beispiel, dass es ganz wesentlich auf die Reihenfolge beim Berechnen der Normalform ankommt: $B_2 := \{ux - y^2, uy - z^2, uz - x^2\}$ und Reduktion des Monoms u^2xyz. Verschiedene Pfade liefern eine der Normalformen y^2z^3, z^3x^2 oder x^3z^2. Begriff des *Reduktionspfads*.

Der folgende Algorithmus **NormalForm** erlaubt es, in einem Polynom $f \in R$ so lange Ersetzungen vorzunehmen, bis der Leitterm des entstehenden Polynoms ein Standardterm bzgl. B ist:

NF(f : Polynom, B : Basis) : Polynom

Input: Polynom $f \in R$, endliche Menge $B \subset R$.

Output: Polynom $f' \in R$ mit $f \equiv f' \pmod{Id(B)}$ und $f' = 0$ oder $lt(f') \notin \Sigma(B)$.

```
while (f ≠ 0) and (M := {b ∈ B : lt(b) | lt(f)} ≠ ∅) do
  choose b ∈ M
  f := f − (lm(f)/lm(b)) b
return f
```

Dieser Algorithmus terminiert offensichtlich, weil die Folge der Leitmonome der in den einzelnen Schritten entstehenden Zwischenergebnisse eine streng monoton fallende Folge von Monomen darstellt, die nach der Definition einer noetherschen Termordnung endlich sein muss.

Extended Division Algorithm

Ähnlich wie im Erweiterten Euklidschen Algorithmus kann man den Normalform-Algorithmus so modifizieren, dass sogar eine Darstellung von $f' - f \in Id(B)$ als polynomiale Kombination der Basiselemente zurückgegeben wird. In obigem Beispiel etwa erhalten wir bei der Berechnung von $g' := \mathtt{NF}(g, B)$ nacheinander

$$\begin{aligned}
g \mapsto g_1 &= g - y^2\, f_1 &&= x^2z^2 - xy^3 - y^4 + y^2z^2\\
\mapsto g_2 &= g_1 - z^2\, f_1 &&= -xy^3 - xyz^2 - y^4\\
\mapsto g_3 &= g_2 + x\, f_3 &&= -xyz^2 - xz^3 - y^4\\
\mapsto g_4 &= g_3 + y\, z\, f_2 &&= -xz^3 - y^4 + y^2z^2\\
\mapsto g_5 &= g_4 + z^2\, f_2 &&= -y^4 + y^2z^2 + yz^3\\
\mapsto g_6 &= g_5 + y\, f_3 &&= y^2z^2 = g'
\end{aligned}$$

also $g = (y^2 + z^2)f_1 + (-y\,z - z^2)f_2 + (-x + y)f_3 + g'$.

Allgemein lassen sich die Kofaktoren während der Reduktion auf dieselbe Weise in einem Vektor $(v_1, \ldots, v_m)$ aufsammeln:

NFwithRelations(f: Polynom, B: Basis): (Polynom,Vektor)

Input: Polynom $f \in R$, endliche Menge $B = \{b_1, \ldots, b_m\} \subset R$

Output: Polynom $f' \in R$ mit $f' = 0$ oder $lt(f') \notin \Sigma(B)$ und Vektor $v = (v_1, \ldots, v_m)$ mit $f = \sum_i v_i\, b_i + f'$.

`for` $i = 1, \ldots, m$ `do` $v_i := 0$
`while` $(f \neq 0)$ `and` $(M := \{b \in B \ : \ lt(b) \,|\, lt(f)\} \neq \emptyset)$ `do`
  `choose` $b_i \in M$
  $f := f - \frac{lm(f)}{lm(b_i)} b_i$
  $v_i := v_i + \frac{lm(f)}{lm(b_i)}$
`return` (f, v)

Diese Darstellung als polynomiale Kombination der Basisvektoren hat eine weitere wichtige Eigenschaft; sie kommt ohne »große« intermediäre Terme aus:

Theorem 5 *Sei $B = \{b_1, \ldots, b_m\} \subset R$ eine endliche Menge von Polynomen. Dann liefert der Algorithmus* **NFwithRelations** *für jedes Polynom $f \in R$ nach endlich vielen Schritten eine Darstellung*

$$f = v_1\, b_1 + \ldots + v_m\, b_m + r$$

mit $v_1, \ldots, v_m, r \in R$, in der $r = 0$ oder $lt(r) \notin \Sigma(B)$ und $lt(f) \geq lt(v_i)\, lt(b_i)$ für alle i gilt.

Das CoCoA-Kommando `DivAlg` (für »division algorithm«) führt diese Rechnungen aus. Im folgenden Beispiel wird die Normalform von xyz bzgl. der Basis B_3 berechnet und die Probe ausgeführt.

```
Use R::=Q[x,y,z],Lex;
B3:=[x + y + z^2 - 3, y^2 - y - z^2 + z, 2yz^2 - 4y + z^4 - 5z^2 + 6];
Res:=DivAlg(xyz,B3);
Res;
Record[Quotients = [-z, yz, -1/2z], Remainder = 1/2z^5 - 7/2z^3 + z^2 + 3z]
------------------------------
ScalarProduct(Res.Quotients,B3)+Res.Remainder;
xyz
------------------------------
```

Ideal Membership Test

Der Algorithmus `NF` erlaubt eine Antwort auf die erste Hälfte des Idealenthaltenseins-Problem:

Theorem 6 *Seien R, f und B wie in obigem Satz. Ist $NF(f, B) = 0$, so gilt $f \in Id(B)$.*

Die Umkehrung dieses Satzes gilt nicht, d.h. es kann durchaus Elemente $f \in I$ geben, für die $\texttt{NF}(f, b) \neq 0$ gilt. Mehr noch kann das Ergebnis vom gewählten Reduktionspfad abhängen. Der Satz kann dahingehend verschärft werden, dass $\texttt{NF}(f, b) = 0$ für einen einzigen Reduktionspfad ausreicht, um auf $f \in I$ zu schließen.

Total Normal Forms

Einen entsprechenden Algorithmus, der **NF** noch rekursiv auf das Reduktum anwendet, bezeichnet man als **totalen Normalform-Algorithmus**.

TNF(f: Polynom, B: Basis): Polynom

Input: Polynom $f \in R$, endliche Menge $B \subset R$
Output: Polynom $f' \in R$ mit $f \equiv f' \pmod{Id(B)}$ und $f' = 0$ oder $T(f') \cap \Sigma(B) = \emptyset$

```
f:=NF(f,B)
if f = 0 then return f else return lm(f) + TNF(red(f), B)
```

Für diesen Algorithmus kann man ebenfalls wieder eine Variante angeben, die $f - f'$ als polynomiale Kombination der Basiselemente $b \in B$ darstellt. Die CoCoA-Kommandos `NR` und `DivAlg` berechnen bereits solche totalen Normalformen.

Interreduced Bases

Ist $f \in B$ ein Element mit $lt(f) \notin Gen(\Sigma(B))$, so kann $lt(f)$ durch die anderen Basiselemente reduziert werden. Wenden wir diese Idee iteriert an, so erhalten wir ein Ergebnis, das der Triangulierung einer Matrix im Gaussverfahren entspricht.

Interreduce(B: Basis): Basis

Input: Basis $B = \{b_1, \ldots, b_m\} \subset R$
Output: Basis B' mit $Id(B) = Id(B')$ und $|B'| = |Gen(\Sigma(B'))|$

```
while exists f ∈ B, lt(f) ∉ Gen(Σ(B)) do
  B = B − {f}
  f' = NF(f, B)
  if f' ≠ 0 then B = B ∪ {f'}
return B
```

Theorem 7 *Der Algorithmus* **Interreduce** *terminiert, wenn $(T, <)$ eine noethersche Termordnung ist, und erfüllt die gegebene Spezifikation.*

Bemerkung: Die Eigenschaft, dass es sich um eine *noethersche* Termordnung handelt, wurde nur für die Termination von NF benötigt; die `while`-Schleife terminiert allein auf Grund des Dicksonlemmas.

5 Groebner Bases

Definition and Motivation

Die systematische Vergrößerung von $\Sigma(B)$ im Zuge des Interreduktionsprozesses kann man verallgemeinern: Wir können beliebige Polynome $f \in I$ mit $lt(f) \notin \Sigma(B)$ mit demselben Erfolg zu B hinzunehmen, um die »Reduktionskraft« von B zu verstärken.

Dafür müssen wir nicht einmal von einer Idealbasis ausgehen, sondern können diesen Prozess mit $B = \emptyset$ als Startmenge beginnen. In jedem Schritt ergänzen wir B um ein Element $0 \neq f \in I$ mit $lt(f) \notin \Sigma(B)$, so lange das möglich ist. Wir bekommen dabei eine Kette $\Sigma_0 \subset \Sigma_1 \subset \ldots$ von echt wachsenden Monoididealen, die nach dem Dicksonlemma nach endlich viele Schritten zu einer Basis G mit der Eigenschaft $\Sigma(G) = \Sigma(I)$ führt.

Es handelt sich dabei um besondere Teilmengen von I, denn selbst für eine Basis B des Ideals I gilt zwar $\Sigma(B) \subseteq \Sigma(I)$, muss aber nicht unbedingt $\Sigma(B) = \Sigma(I)$ gelten.

Beispiel: $I = Id(f_1 = x^3 - 2xy, f_2 = x^2y - 2y^2 + x)$. Dann gilt $B = \{f_1, f_2\}, \Sigma(B) = (x^3, x^2y)$, aber wegen $x^2 = x\, f_2 - y\, f_1 \in I$ außerdem wenigstens $x^2 \in \Sigma(I)$.

Definition 4 *Eine Teilmenge $G \subset I$ des Ideals I heißt* Gröbnerbasis *von I, wenn $\Sigma(G) = \Sigma(I)$ gilt.*

Das zum Monoidideal $\Sigma(I)$ gehörende Potenzproduktideal $Lt(I) \subset R$ wird als linearer Vektorraum von den Leittermen der Elemente $0 \neq f \in I$ aufgespannt und heißt deshalb das *Leittermideal* von I (bzgl. der festgewählten Termordnung $<$).

Hilbert's Basissatz

Obwohl unsere Argumentation nicht konstruktiv war, haben wir oben gezeigt, dass jedes Ideal eine Gröbnerbasis hat. Es bleibt noch der Begriff »Basis« zu rechtfertigen, d.h. zu zeigen, dass G wirklich eine Basis des Ideals I ist.

Theorem 8 *Jede Gröbnerbasis $G = \{g_1, \ldots, g_r\} \subset I$ ist eine Basis des Ideals I.*

Beweis: Wegen $G \subset I$ bleibt nur zu zeigen, dass jedes Element $f \in I$ auch tatsächlich als polynomiale Kombination dieser Polynome darstellbar ist. Berechnen wir die Normalform mit Relationen von f bzgl. G, erhalten wir eine Darstellung

$$f = p_1 g_1 + \ldots + p_r g_r + q$$

mit einem Polynom q, das entweder Null ist oder einen Leitterm $lt(q) \notin \Sigma(G) = \Sigma(I)$ hat. Da letzteres wegen $q \in I$ nicht möglich ist, folgt $f \in Id(G)$. $\square$

Wir sagen deshalb auch ohne Bezug auf ein Ideal, dass G eine Gröbnerbasis ist, wenn G dies bzgl. des Ideals $I = Id(G)$ ist.

Einer der zentralen Sätze der kommutativen Algebra ergibt sich nun als einfache Folgerung:

Corollary 1 (Hilberts Basissatz) *Jedes Ideal $I \subset R$ besitzt eine endliche Basis.*

Theorem 9 *Sei $G = \{g_1, \ldots, g_r\}$ eine Gröbnerbasis des Ideals I. Dann gilt*

$$f \in I \;\Leftrightarrow\; NF(f, G) = 0.$$

Corollary 2 *Für eine Gröbnerbasis G und ein Polynom $f \in R$ ist $TNF(f, G)$ unabhängig vom gewählten Reduktionspfad.*

S-Polynomials

Wir wollen nun gezielt Beispiele von Polynomen konstruieren, die im von der Basis B erzeugten Ideal liegen, aber deren Leitterm nicht in $\Sigma(B)$ enthalten ist. Einen Weg zur Konstruktion solcher Polynome hatten wir im Beispiel $B_2 := \{f_1 = ux - y^2, f_2 = uy - z^2, f_3 = uz - x^2\}$ gesehen: u^2xyz lässt sich auf zwei verschiedenen Pfaden jeweils zur Normalform y^2z^3 oder x^3z^2 reduzieren. Demzufolge ist $f = x^3z^2 - y^2z^3 \in I$, aber $lt(f) = x^3z^2 \notin \Sigma(B_2)$.

```
Use R::=Q[u,x,y,z],Lex;
F1:=ux-y^2; F2:=uy-z^2; F3:=uz-x^2;
```

Kleinste Gegenbeispiele können auf folgende Weise konstruiert werden:

$$\begin{aligned} s_{12} &= y \cdot f_1 - x \cdot f_2 = (uxy - y^3) - (uxy - xz^2) = xz^2 - y^3 \\ s_{13} &= z \cdot f_1 - x \cdot f_3 = (uxz - y^2z) - (uxz - x^3) = x^3 - y^2z \\ s_{23} &= z \cdot f_2 - y \cdot f_3 = (uyz - z^3) - (uyz - x^2y) = x^2y - z^3 \end{aligned}$$

In jedem der drei Beispiele kann man dem Ergebnis nicht mehr ansehen, wie es als Linearkombination der Basiselemente entstanden ist, da sich diese Kombination nur durch Hinzufügen zweier gleicher Terme mit entgegengesetztem Vorzeichen ergibt, die in der fixierten Termordnung *größer* als die verbleibenden Terme sind. Ein solches Element hat nur dann eine verschwindende Normalform, wenn es einen zweiten Weg zu seiner Darstellung als Element von I gibt, die *ohne Termüberschreitung* auskommt.

Das allgemeine Schema der Konstruktion solcher Elemente suggeriert die folgende

Definition 5 *Seien $f, g \in R$ zwei nichttriviale Polynome und $m = lcm(lt(f), lt(g))$ das kleinste gemeinsame Vielfache der Leitterme der beiden Polynome. Dann bezeichnen wir das Polynom*

$$S(f, g) := \frac{m}{lm(f)} f - \frac{m}{lm(g)} g = \frac{m}{lm(f)} red(f) - \frac{m}{lm(g)} red(g),$$

das die kleinste monomiale Kombination aus f und g ist, in der sich die beiden Kopfterme gegenseitig wegheben, als das S-Polynom *von f und g.*

Eine entsprechende Funktion kann man in CoCoA wie folgt vereinbaren:

```
Define SPoly(F,G)
  M:=LCM(LT(F),LT(G));
  Return M/LM(F)*F-M/LM(G)*G;
EndDefine;
```

Für die Elemente einer Idealbasis B definieren wir noch abkürzend

```
Define S(B,I,J)
  Return SPoly(B[I],B[J]);
EndDefine;
```

Ein solches S-Polynom, falls es nicht verschwindet, besitzt einen Leitterm, der echt kleiner als der erwartete Leitterm m ist. Auf diese Weise entstehen Polynome, deren Leitterm möglicherweise nicht in $\Sigma(B)$ enthalten ist. Durch Hinzunahme dieser Polynome in die Basis vergrößern wir also $\Sigma(B)$. In unserem Beispiel erhalten wir eine neue Basis

$$B_{2a} = \{f_1, f_2, f_3, f_4 := s_{12}, f_5 := s_{13}, f_6 := s_{23}\}$$

mit $\Sigma(B_{2a}) = (ux, uy, uz, xz^2, x^3, x^2y)$. Offensichtlich ist $NF(S(f_i, f_j), B_{2a}) = 0$ für $(i, j) \in \{(1,2), (1,3), (2,3)\}$. Andererseits können zwischen den neuen und alten Elementen neue S-Polynome konstruiert werden:

$$s_{14} = z^2 f_1 - u f_4 = (uxz^2 - y^2z^2) - (uxz^2 - uy^3) = uy^3 - y^2z^2$$

Im Gegensatz zu obigen Polynomen ist dieses Polynom nicht bereits in Normalform bzgl. B_{2a}. Es gilt

$$s_{14} \mapsto_{f_2} 0$$

also $NF(s_{14}, B_{2a}) = 0$. Damit entsteht aus diesem Polynom kein neues Polynom aus I mit bis dahin unbekanntem Leitterm. Dasselbe gilt für die S-Polynome $s_{ij}, i \in \{1,2,3\}, j \in \{4,5,6\}$. Die restlichen S-Polynome liefern

$$\begin{aligned} s_{45} &= -x^2y^3 + y^2z^3 \mapsto_{f_6} 0 \\ s_{46} &= -xy^4 + z^5 =: f_7 \\ s_{56} &= xz^3 - y^3z \mapsto_{f_4} 0 \end{aligned}$$

Wir erhalten also ein weiteres Polynom $f_7 \in I$ mit bis dahin unbekanntem Leitterm xy^4 und schließlich wegen

$$s_{47} = y^4 f_4 + z^2 f_7 = -y^7 + z^7 =: f_8$$

ein letztes solches Polynom. Die Normalformen aller weiteren S-Polynome verschwinden, so dass wir auf einfachem Wege keine neuen Polynome $f \in I$ mit $lt(f) \notin \Sigma(G)$ für $G = \{f_1, \ldots, f_8\}$ konstruieren können. Weiter unten werden wir sehen, dass G in der Tat bereits eine Gröbnerbasis ist.

Characterization of Groebner Bases

Theorem 10 (Charakterisierungssatz für Gröbnerbasen)
Die folgenden Bedingungen an eine Teilmenge G eines Ideals $I \subset R$ sind äquivalent:

1. *G ist eine Gröbnerbasis von I, d.h. $\Sigma(I) = \Sigma(G)$.*
2. *Für jedes Element $f \in I$ und jede Reduktionsstrategie gilt $NF(f, G) = 0$.*

2'. *Für jedes Element $f \in I$ gibt es eine Reduktionsstrategie mit $NF(f, G) = 0$.*

3. *Für jedes Paar $g_1, g_2 \in G$ und jede Reduktionsstrategie gilt $NF(S(g_1, g_2), G) = 0$.*

3'. *Für jedes Paar $g_1, g_2 \in G$ gibt es eine Reduktionsstrategie mit $NF(S(g_1, g_2), G) = 0$.*

4. *Jedes Element $f \in I$ hat eine Darstellung*

$$f = \sum_{g_i \in G} h_i g_i \quad mit \quad \forall\, i\ (lt(f) \geq lt(h_i g_i)).$$

5. *Die Standardterme $N(G) := T(X) \setminus \Sigma(G)$ sind linear unabhängig* $(\mathrm{mod}\ I)$.

5'. *Die Standardterme $N(G)$ bilden eine Vektorraumbasis des Faktorrings R/I, d.h. jedes Element $f \in R$ besitzt eine eindeutige Darstellung*

$$f \equiv \sum_{m \in N(G)} c_m m \quad (\mathrm{mod}\ I)$$

mit $c_m \in k$.

6 Buchberger's Algorithm

Ähnlich wie oben können wir auch im allgemeinen Fall versuchen, aus einer gegebenen Idealbasis B eine Gröbnerbasis zu konstruieren. Wir versuchen, nacheinander alle S-Polynome $S(f_i, f_j),\ f_i, f_j \in B$ vermöge B zu reduzieren. Ist $f := NF(S(f_i, f_j), B) \neq 0$, so wissen wir zumindest, dass $lt(f) \in \Sigma(I) \setminus \Sigma(B)$, d.h. $f \in I$ ein Element aus dem Ideal ist, dessen Leitterm man aus den bisher bekannten Basiselementen nicht herleiten kann. Fügen wir andererseits dieses Polynom zur Menge B hinzu, kann $S(f_i, f_j)$ nunmehr trivialerweise zu Null reduziert werden. Durch die Hinzunahme des neuen Basiselements vergrößert sich andererseits die Anzahl der möglichen S-Polynome, so dass die Termination dieses Vorgehens eines Beweises bedarf.

Die formale Spezifikation des entsprechenden Algorithmus, den man zu Ehren seines Erfinders den **Buchergeralgorithmus** nennt, sieht wie folgt aus:

GBasis(B: Basis): Basis

Input: Endliche Menge $B = \{f_1, \ldots, f_m\} \subset R$.

Output: Gröbnerbasis G des Ideals $I = Id(B)$.

```
G:=B;
P := {(f_i, f_j) | 1 ≤ i < j ≤ m};
While P ≠ ∅ do
  Choose p ∈ P; P := P \ {p};
  f := NF(S(p), G)
  if f ≠ 0 then
    P := P ∪ {(g, f) | g ∈ G};
    G := G ∪ {f};
return G;
```

Theorem 11 *Der Algorithmus terminiert nach endlich vielen Schritten.*

Beweis: In jedem Schritt mit $f \neq 0$ wird $\Sigma(G)$ echt vergrößert. Nach dem Dicksonlemma sind aber nur endlich viele solche Schritte möglich. $\square$

Theorem 12 *Ist G eine Gröbnerbasis des Ideals I und $G' \subset G$ eine Teilmenge, so dass $Gen(\Sigma(G)) = \{lt(g), g \in G'\}$ gilt, so ist auch G' eine Gröbnerbasis von I. Eine solche Gröbnerbasis nennt man* minimal.

Der Beweis dieses Satzes ergibt sich sofort aus der Definition einer Gröbnerbasis. Auf Grund der Eindeutigkeit der Minimalbasis des Monoidideals $\Sigma(I)$ ist die Menge $\{lt(g), g \in G'\}$ eindeutig bestimmt. Die Menge

$$\{lt(g) - TNF(lt(g), G'), g \in G'\} \subset I$$

bezeichnet man schließlich als *minimale reduzierte Gröbnerbasis*. Jedes dieser Polynome ist die Differenz zwischen einem minimalen Nichtstandardterm und dessen eindeutiger Darstellung $\pmod I$ als Linearkombination von (in der Termordnung kleineren) Standardtermen. Offensichtlich ist eine solche minimale reduzierte Gröbnerbasis eindeutig bestimmt.

7 Applications

Trivial Ideals

Theorem 13 *Gegeben sei ein polynomiales Gleichungssystem $B \subset R = k[x_1, \ldots, x_n]$ mit Koeffizienten aus einem Körper k und ein algebraisch abgeschlossener Erweiterungskörper K von k. Folgende Aussagen sind dann äquivalent:*

1. *$V_K(B) = \emptyset$, d.h. B hat keine gemeinsamen Nullstellen über K.*
2. *$Id(B) = Id(1)$ ist das Einsideal.*
3. *Jede Gröbnerbasis $G = GBasis(B)$ enthält ein konstantes Polynom.*
4. *$\{1\}$ ist die minimale reduzierte Gröbnerbasis von B.*

Mit einer Modifikation dieses Vorgehens können wir auch die Frage beantworten, für welche $a \in \mathbb{C}$ das System

$$B = \left\{x^2 + y^2 - 2, x^3 + y^3 - 3, x^4 + y^4 - a\right\}$$

Lösungen hat. Dazu führen wir dieselben Rechnungen wie oben aus:

```
Use R::=Q[x,y,a],Lex;
B:=[x^2 + y^2 - 2, x^3 + y^3 - 3, x^4 + y^4 - a];
ReducedGBasis(Ideal(B));
[a^3 + 6a^2 - 108a + 274,
  x + y - 1/3a^2 - 11/3a + 62/3,
  y^2 - 1/3ya^2 - 11/3ya + 62/3y + 1/2a^2 + 5a - 31]
------------------------------
```

Die reduzierte Gröbnerbasis enthält insbesondere ein Element $g(a) = a^3 + 6a^2 - 108a + 274$, welches nur von a abhängt und im Ideal liegt, welches von B in $R = \mathbb{C}[x, y, a]$ erzeugt wird. Also gibt es eine polynomiale Kombination

$$g(a) = \sum_{b \in B} h_b(x, y, a) \cdot b(x, y, a)$$

und für konkrete Zahlen $a_0 \in \mathbb{C}$ ist $g(a_0)$ im Ideal I_0 enthalten, welches von $B_0 = B[a \mapsto a_0]$ erzeugt wird. I_0 ist also *höchstens* dann nicht trivial, wenn $g(a_0) = 0$ gilt.

Da B und G beides Basen des Ideals $I = Id(B) \subset k[x, y, a]$ sind, lassen sich die Elemente aus B als polynomiale Kombinationen der Elemente aus G und umgekehrt darstellen.

$$B^T = M_1 \cdot G^T \qquad G^T = M_2 \cdot B^T, \qquad M_1, M_2 \in Mat(R)$$

Dies gilt auch nach einer Substitution $a \mapsto a_0$: B_0 und $G_0 = G[a \mapsto a_0]$ erzeugen beide das Ideal I_0 – allerdings muss G_0 nicht unbedingt mehr Gröbnerbasis sein. Wählen wir a_0 mit $g(a_0) = 0$, so können wir die beiden Lösungen in unserem Fall aber aus G_0 unmittelbar ablesen.

Das Polynom $g(a)$ bezeichnet man auch als die *Diskriminante* des (parametrischen) Gleichungssystems $B \subset k(a)[x, y]$

Elimination Orders and the Elimination Theorem

Sei $B \subset R = k[\mathbf{x}]$ eine endliche Menge von Polynomen und die Menge der Variablen in zwei Teilmengen $\mathbf{x} = (x_1, \ldots, x_k, y_1, \ldots, y_m)$ aufgeteilt. Wir fragen nach den Polynomen im Ideal $I = Id(B)$, die $x_1, \ldots, x_k$ nicht enthalten, also nach einer Basis des *Eliminationsideals*

$$I' = Id(B) \bigcap k[y_1, \ldots, y_m].$$

Zu dessen Berechnung wählen wir auf $T(\mathbf{x})$ eine Termordnung, in der jeder Term, der eine Variable x_i enthält, größer ist als jeder Term, der nur Variablen y_j enthält. Solche Termordnungen bezeichnet man als *Eliminationsordnungen* für $(x_1, \ldots, x_k)$, da ein Polynom $f(x_1, \ldots, x_k, y_1, \ldots, y_m)$ genau dann keine der Variablen $x_1, \ldots, x_k$ enthält, wenn dies für dessen Leitterm $lt(f)$ gilt.

Neben der lexikografischen Ordnung gibt es eine Reihe anderer Eliminationsordnungen, bzgl. derer sich Gröbnerbasen gewöhnlich schneller ausrechnen lassen. So können wir etwa jede Matrix-Termordnung verwenden, deren erster Gewichtsvektor durch $w(x_i) = 1, w(y_j) = 0$ gegeben ist.

Theorem 14 *Ist $G = GBasis(B)$ eine (min. reduzierte) Gröbnerbasis des Polynomsystems $B \subset R = k[x_1, \ldots, x_k, y_1, \ldots, y_m]$ bzgl. einer Eliminationsordnung für $x_1, \ldots, x_k, y_1, \ldots, y_m$, so ist*

$$G' = \{g \in G \,:\, lt(g) \in T(y_1, \ldots, y_m)\}$$

eine (min. reduzierte) Gröbnerbasis des Eliminationsideals $I' = Id(B) \bigcap k[y_1, \ldots, y_m]$.

Die lexikografische Termordnung ist eine Eliminationsordnung für jedes Anfangssegment der Variablen. Damit hat eine Gröbnerbasis bgzl. dieser Ordnung eine »Dreiecksgestalt«, aus der heraus sich die Lösungsmenge eines polynomialen Gleichungssystems berechnen lässt.

Corollary 3 *Ist $G = GBasis(B)$ eine (min. reduzierte) Gröbnerbasis von $B \subset R = k[\mathbf{x}]$ bzgl. der lexikografischen Termordnung mit $x_1 > \cdots > x_n$, so ist*

$$G_i = \{g \in G : lt(g) \in T(x_i, \ldots, x_n)\}$$

eine (min. reduzierte) Gröbnerbasis des Eliminationsideals $Id(B) \bigcap k[x_i, \ldots, x_n]$.

Insbesondere enthält G_n das Polynom $g(x_n) \in I$ kleinsten Grades, das nur von x_n abhängt, wenn es ein solches Polynom gibt und G eine minimale reduzierte Gröbnerbasis ist.

Die letzte Aussage folgt unmittelbar aus der Tatsache, dass $k[x_n]$ ein Hauptidealring ist. $G_i, i < n$ kann dagegen mehr als $n - i$ Polynome enthalten.

Dies liefert ein **induktives Verfahren zum Lösen polynomialer Gleichungssysteme**:

> Kennt man eine gemeinsame Nullstelle $(x_{i+1}^0, \ldots, x_n^0)$ von G_{i+1}, so enthält $G_i \backslash G_{i+1}$ alle Polynome, die zur Bestimmung von solchen x_i^0 verwendet werden können, dass $(x_i^0, \ldots, x_n^0)$ eine Nullstelle von G_i ist.

Dies entspricht der Triangulierung eines linearen Gleichungssystem durch den Gauß-Algorithmus.

Beispiel (CoCoA):

```
------------------------------
Use R::=Q[x,y,z],Lex;
B:=[x^2 + y + z - 3, x + y^2 + z - 3, x + y + z^2 - 3];
I:=Ideal(B);
ReducedGBasis(I);
[ y^2 - y - z^2 + z,
  yz^2 - 2y + 1/2z^4 - 5/2z^2 + 3,
  z^6 - 10z^4 + 4z^3 + 19z^2 - 8z - 6,
  x + y + z^2 - 3]
------------------------------
```

Die Gröbnerbasis enthält mit $f = z^6 - 10z^4 + 4z^3 + 19z^2 - 8z - 6$ ein Polynom allein in z, dessen Nullstellen bestimmt werden können.

```
------------------------------
Factor(I.GBasis[3]);
[[z + 3, 1], [z - 1, 1], [z^2 - 2, 1], [z^2 - 2z - 1, 1]]
------------------------------
```

Setzen wir diese in gb ein. Für $z = 1$ erhalten wir zwei Gleichungen zur Bestimmung von y, die aber voneinander abhängig sind.

```
------------------------------
Subst(I.GBasis,[[z,1]]);
I1:=Ideal(It);
ReducedGBasis(I1);
[y - 1, x - 1]
------------------------------
```

Wir können daraus die Lösung $(x, y, z) = (1, 1, 1)$ leicht ablesen. Denselben Effekt erhält man, wenn man f durch einen dieser Faktoren ersetzt. Für $z = -3$ erhalten wir auf diese Weise.

Allgemein gilt: Ist $f = f_1 \cdot \ldots \cdot f_k$ eine Zerlegung in Faktoren und F eine Menge weiterer Polynome, so gilt offensichtlich

$$V(F \cup \{f\}) = \bigcup_k V(F \cup \{f_k\})$$

Neben den linearen enthält obige Faktorzerlegung noch quadratische Faktoren. Für den ersten Zugang – Substitution von Werten $z = \pm\sqrt{2}$ für z – müssen wir mit Polynomen rechnen, deren Koeffizienten in einem Erweiterungskörper liegen. Für den zweiten Zugang ist dies zunächst nicht erforderlich:

```
------------------------------
I3:=I+Ideal([z^2-2]);
ReducedGBasis(I3);
[z^2 - 2, y^2 - y + z - 2, x + y - 1]
I4:=I+Ideal([z^2 - 2z - 1]);
ReducedGBasis(I4);
[z^2 - 2z - 1, y + z - 1, x + z - 1]
------------------------------
```

In I_3 gibt es zu jeder der beiden Lösungen für z zwei Werte (x, y, z), also insgesamt 4 Lösungen. I_4 trägt zwei weitere Lösungen zur vollen Lösungsmenge bei, so dass $V(I)$ aus insgesamt 8 Punkten besteht.

Zusammenfassend hätten wir auch rechnen können

```
------------------------------
[ ReducedGBasis(Ideal(Concat(B,[J[1]]))) | J In Factor(I.GBasis[3])];
[ [y + 3, z + 3, x + 3],
  [y - 1, z - 1, x - 1],
  [z^2 - 2, y^2 - y + z - 2, x + y - 1],
  [z^2 - 2z - 1, y + z - 1, x + z - 1]]
------------------------------
```

Bessere Ergebnisse liefert eine integrierte Variante von Buchbergeralgorithmus und Faktorisierung, der **Gröbnerfaktorisierer** der aber nur in wenigen CAS implementiert bzw. nicht explizit zugänglich ist.

8 Independent Sets and Dimension

Die Dimension eines Ideals $I \subset R$ lässt sich berechnen als

$$\dim(R/I) = \max\left(d \ : \ \exists\,(x_{i_1}, \ldots, x_{i_d}) \ \text{ mit } I \bigcap k[x_{i_1}, \ldots, x_{i_d}] = \{0\}\right).$$

Für Primideale I – diese entsprechen irreduziblen Varietäten – ist dies gerade der Transzendenzgrad des Quotientenkörpers $Q(R/I)$ und jede solche maximale Teilmenge $x_{i_1}, \ldots, x_{i_d}$ eine Transzendenzbasis von $Q(R/I)$.

Ist $I = \cap_\alpha P_\alpha$ Durchschnitt von Primidealen und $I \bigcap k[x_{i_1}, \ldots, x_{i_d}] = \{0\}$, so gibt es ein α mit $P_\alpha \bigcap k[x_{i_1}, \ldots, x_{i_d}] = \{0\}$ und folglich $\dim(R/I) = \max(\dim(R/P_\alpha))$. Die obige Definition erweitert also den Dimensionsbegriff auf natürliche Weise auf reduzible Varietäten.

Eine Teilmenge $x_{i_1}, \ldots, x_{i_d}$ mit

$$I \bigcap k[x_{i_1}, \ldots, x_{i_d}] = \{0\}$$

wird auch als *unabhängig* modulo I bezeichnet. Dies lässt sich zwar durch eine Gröbnerbasis-Berechnungen bzgl. einer Eliminationsordnung überprüfen, aber für verschiedene Teilmengen der Variablen sind dazu Gröbnerbasen für je andere Termordnungen zu berechnen.

Wir fixieren deshalb eine Termordnung auf R und stellen die folgende

Frage: Welche Informationen über die Dimension kodiert $\Sigma(I)$ bzw. $Lt(I)$?

Sei $R' = k[x_{i_1}, \ldots, x_{i_d}]$ für eine fixierte Teilmenge $(x_{i_1}, \ldots, x_{i_d})$ der Variablen. Offensichtlich gilt

$$Lt(I) \bigcap R' = \{0\} \Rightarrow I \bigcap R' = \{0\}$$

Wir nennen deshalb die Teilmenge $(x_{i_1}, \ldots, x_{i_d})$ der Variablen *streng unabhängig* bzgl. I (und der Termordnung), wenn

$$\Sigma(I) \cap T(x_{i_1}, \ldots, x_{i_d}) = \emptyset$$

gilt, und

$$d' = \max\left(d \ : \ \exists\,(x_{i_1}, \ldots, x_{i_d}) \ \text{ mit } \Sigma(I) \cap T(x_{i_1}, \ldots, x_{i_d}) = \emptyset\right)$$

die *strenge Dimension von* R/I. Nach Definition gilt $d' \leq \dim(R/I)$.

Theorem 15 *Für jedes Ideal stimmen Dimension und strenge Dimension überein.*

9 Groebner Weighted Deformations

Der Beweis dieses Satzes macht von Techniken der algebraischen Deformationstheorie Gebrauch.

Wir betrachten wieder das Beispiel

$$B = \left\{x^2 + y + z - 3, x + y^2 + z - 3, x + y + z^2 - 3\right\}$$

und dessen Gröbnerbasis G bzgl. der lexikographischen Termordnung gerade aus den Polynomen

$$\begin{aligned}
&y^2 - y - z^2 + z,\\
&yz^2 - 2y + \frac{1}{2}z^4 - \frac{5}{2}z^2 + 3,\\
&z^6 - 10z^4 + 4z^3 + 19z^2 - 8z - 6,\\
&x + y + z^2 - 3
\end{aligned}$$

besteht. Es stellt sich zunächst heraus, dass es Gewichtsvektoren §w§ mit positiven ganzzahligen Gewichten gibt, so dass diese Menge auch Gröbnerbasis ist für jede Matrix-Termordnung, die w verfeinert. Dazu muss nur gewährleistet sein, dass

$$w(y^2) > w(y), w(z^2), w(z); \quad w(yz^2) > w(y), w(z^4), w(z^2), w(1); \quad w(z^6) > w(z^4), w(z^3), w(z^2), w(z), w(1);$$

gilt. Einige dieser Beziehungen sind für alle noetherschen Termordnungen gültig, andere ergeben sich als Folge aus dritten. Eine hinreichende Bedingung ist etwa $x > y > z^2$, was für den Gradvektor $(4, 3, 1)$ erfüllt ist. Und in der Tat, wenn wir eine Gröbnerbasis bzgl. einer Matrix-Termordnung mit diesem ersten Gradvektor berechnen, dann erhalten wir dasselbe Leittermideal $\Sigma(I)$ (und damit auch dieselben Standardterme und dieselben totalen Normalformen) wie bgzl. der lexikographischen Termordnung.

```
------------------------------
Use R::=Q[x,y,z],Ord(Mat[[4,3,1],[1,0,0],[0,1,0]]);
B:=[x^2 + y + z - 3, x + y^2 + z - 3, x + y + z^2 - 3];
I:=Ideal(B);
ReducedGBasis(I);
[yz^2 + 1/2z^4 - 2y - 5/2z^2 + 3,
 z^6 - 10z^4 + 4z^3 + 19z^2 - 8z - 6,
 x + y + z^2 - 3,
 y^2 - y - z^2 + z]
------------------------------
```

Lemma 1 *Sei $G = \left\{\mathbf{x}^\alpha - \sum_{\mathbf{x}^\beta \in N} c_{\alpha\beta}\mathbf{x}^\beta : \mathbf{x}^\alpha \in Gen(\Sigma)\right\}$ eine minimale reduzierte Gröbnerbasis des Ideals I, wobei $N = T \setminus \Sigma(I)$ die Menge der Standardterme bezeichnet. Dann gibt es einen positiven Gewichtsvektor $w \in \mathbb{Z}_+$, für den gilt:*

$$\forall \alpha,\ \beta\ \ (c_{\alpha\beta} \neq 0 \Rightarrow w(\alpha) > w(\beta))$$

Für jede Termordnung $<'$, die w verfeinert, gilt $\Sigma'(G) = \Sigma(G)$ und G ist damit eine Gröbnerbasis auch bgzl. $<'$.

Beweis: G ist eine Gröbnerbasis bzgl. jeder Matrix-Termordnung, deren erster Gradvektor w die Bedingung

$$\forall \alpha, \beta\ \ (c_{\alpha,\beta} \neq 0 \Rightarrow w(\alpha - \beta) > 0)$$

erfüllt. Wir fügen noch die Bedingungen $w(e_i) > 0, i = 1, \ldots, n$, hinzu, wobei e_i für den i-ten Einheitsvektor steht, d.h. $x_i = \mathbf{x}^{e_i}$ gilt.

Andererseits gilt $w_0(\alpha - \beta) \geq 0$ und auch $w_0(e_i) \geq 0$, wobei w_0 der erste Gradvektor der Termordnung ist, bzgl. welcher G berechnet wurde. Die (endlich vielen!) Vektoren $\alpha - \beta \in \mathbb{Z}^n$ und $e_1, \ldots, e_n$ liegen also innerhalb des Positivkegels $(C_0)_+$ und spannen somit einen Kegel mit Spitze auf. Der dazu duale Kegel enthält dann innere Punkte mit rationalen (und damit – nach Skalierung – auch solche mit ganzzahligen) Koordinaten, die wegen $w(e_i) > 0$ sämtlich positiv sein müssen. $\square$

Mit einem solchen Gradvektor kann nun eine Familie von Gröbnerbasen

$$G_t = \left\{ \mathbf{x}^\alpha - \sum_\beta c_{\alpha\beta}\mathbf{x}^\beta \cdot t^{w(\alpha)-w(\beta)} : \mathbf{x}^\alpha \in Gen(\Sigma) \right\}$$

über dem Ring $R_t = k[t][x_1, \ldots, x_n]$ konstruiert werden, die für $t = 1$ die Gröbnerbasis G des Ausgangsideals I und für $t = 0$ das PP-Ideal $Lt(I)$ ergeben. Eine solche Familie von Idealen $I_t = Id(G_t)$ bezeichnet man als *Deformation* des Ideals I, jedes einzelne Ideal I_t als *Faser* der Deformation.

Offensichtlich ist die Menge der Standardterme nicht nur eine k-Basis von R/I, sondern auch eine Basis des freien $k[t]$-Moduls R'/I_t mit $R' = k[x_1, \ldots, x_n, t]$ und der speziellen Matrix-Termorodnung, deren erster Gradvektor gerade w ist, erweitert um die Setzung $w(t) = 1$. Bzgl. dieses Gewichtsvektors sind die Elemente aus G_t homogene Polynome und G_t ist auch in diesem Ring eine Gröbnerbasis.

Die Deformation $Spec\,(R_t/I_t)$ ist also flach über der Basis $Spec(k[t])$ und damit haben alle Fasern dieselbe Dimension.

In unserem Beispiel ergäbe sich

```
yz^2 + 1/2z^4t - 2yt^2 - 5/2z^2t^3 + 3t^5,
z^6 - 10z^4t^2 + 4z^3t^3 + 19z^2t^4 - 8zt^5 - 6t^6,
x + yt + z^2t^2 - 3t^4,
y^2 - yt^3 - z^2t^4 + zt^5
```

10 Improvements – The Pair Criteria and Syzygies of $Lt(I)$

Many S-polynomials $S(f_i, f_j)$ reduce to zero. Such pairs (i, j) are called *useless*. In most cases the non-zero polynomials »missing« for a complete GBasis are early found and all the additional time is spent reducing such useless pairs. Moreover usually the reducibility to zero is much harder to detect than non-reducibility, since all intermediate terms have to be »eaten«. It happens quite often that more than 80 % of CPU time is spent to prove that the remaining S-polynomials reduce to zero.

Hence there is a big interest in rules to detect such useless pairs without computation. A first such criterion applies to polynomials with relatively prime lt's.

Theorem 16 *(Hauptsyzygienkriterium) Sind $f, g \in R$ nichttriviale Polynome mit teilerfremden Leittermen, so gilt $NF(S(f, g), \{f, g\}) = 0$.*

Beweis: Aus der Teilerfremdheit folgt $m = lcm(lt(f), lt(g)) = lt(f) \cdot lt(g)$ und somit

$$S(f,g) = \frac{m}{lm(f)} red(f) - \frac{m}{lm(g)} red(g) = \frac{1}{lc(f)\,lc(g)} \left(lm(g)\, red(f) - lm(f)\, red(g)\right)$$

Führen wir nun die Substitutionen $lm(f) \mapsto -red(f)$ und $lm(g) \mapsto -red(g)$ aus, so erhalten wir 0. □

For more advanced criteria we fix some notation.

$G = \{f_1, \ldots, f_N\}$ is the base under consideration in a running GBasis computation. Further we assume all $lc(f_i) = 1$, set $m_i = lt(f_i)$, $m_I = \mathrm{lcm}(m_i,\ i \in I)$ for a subset $I \subset \{1, \ldots, m\}$, $e_i \in R^N$ the i-th unit vector and

$$s_{ij} = \frac{m_{ij}}{m_i} e_i - \frac{m_{ij}}{m_i} e_j \in R^N$$

for $1 \le i < j \le N$. All the s_{ij} form a generating set for the first syzygy module $S_1 = Ker(\phi_1)$ of $Lt(G)$, i.e., the kernel of the map

$$\phi_1 : R^N \to R \quad \text{given by} \quad e_i \mapsto m_i$$

Hence two other criteria for G to be a GBasis are

6. For each $s \in S_1$ exists a reduction strategy such that $NF(s \cdot B, G) = 0$.

6'. For each $s \in S_1$ and every reduction strategy we have $NF(s \cdot B, G) = 0$.

It is enough to check (6.) for s from a base of S_1 and hence it is enough to test a subset of the s_{ij} that generates S_1. Although it is well-known that all such bases have the same cardinality it may be useful to have different choices at hand since the resulting normal form computations can differ in expense.

To get a complete picture about that we have to determine the relations between the s_{ij}, i.e., to compute the second syzygy module $S_2 = Ker(\phi_2)$ of $Lt(G)$ with

$$\phi_2 : R^{\binom{N}{2}} \to R^N \quad \text{given by} \quad e_{ij} \mapsto s_{ij}$$

A (not necessarily minimal) generating set of S_2 are the elements

$$s_{ijk} = \frac{m_{ijk}}{m_{ij}} e_{ij} - \frac{m_{ijk}}{m_{ik}} e_{ik} + \frac{m_{ijk}}{m_{ijk}} e_{jk}$$

for $1 \leq i < j < k \leq N$. If one of the coefficients, e.g., $\frac{m_{ijk}}{m_{ij}}$ is equal to 1 the relation s_{ijk} tells that s_{ij} can be skipped in a generating set of S_1 provides the other two elements are contained in the generating set or can safely be generated without s_{ijk}. The alternatives are

1. None of the coefficients is constant, none of the s_{ij}, s_{ik}, s_{jk} can be skipped this way.
2. Exactly one of the coefficients is constant. The corresponding s can safely be skipped.
3. More than one of the coefficients is constant. Only one of the corresponding s can safely be skipped.

To handle that consistently the following strategy is usually applied if a new polynomial f_k has to be integrated into a partially computed GBasis $G = (f_i, 1 \leq i < k)$ with pair set P:

- Skip (j, k) if there is a $i < j$ with $m_{ijk} = m_{jk}$ (i.e., $m_i | m_{jk}$). The syzygy looks like [. . 1].
- Skip (i, k) if there is a $i < j$ with $m_{ijk} = m_{ik}$ (i.e., $m_j | m_{ik}$) **and** $m_{ijk} \neq m_{jk}$ (i.e., $m_i \not| m_{jk}$, hence (j, k) was not skipped in the first run). The syzygy looks like [. 1 $*$], where $*$ stands for a non-constant term.
- Scan the old pairs (i, j) and skip those with $m_{ijk} = m_{ij}$ (i.e., $m_k | m_{ij}$) **and** $m_{ijk} \neq m_{ik},\ m_{ijk} \neq m_{jk}$ (i.e., $m_i \not| m_{jk},\ m_j \not| m_{ik}$, hence neither (i, k) nor (j, k) was skipped in the first two runs).

This is more or less the **Gebauer-Möller criterion** for useless pairs. Together with special arrangements, pair selection and reduction strategies it is the heart of an efficient classical implementation of the Groebner algorithm.

11 Multimodular and Trace Algorithms

Another issue about Groebner bases addresses the observation that coefficient growth may have a significant influence on the computational complexity of computing a GBasis for an ideal $I \subset \mathbb{Q}[\mathbf{x}]$ given a basis $B = \{f_1, \ldots, f_m\}$ that we can assume to have integer coefficients.

For many purposes (e.g., computation of the dimension or the Hilbert Series) these coefficients don't matter but only $\Sigma(I)$ has to be determined correctly. One would expect that modular methods in »most« cases give the correct $\Sigma(I)$ but limit these expenditures to a necessary minimum.

To be more precisely, consider the ideal $I \subset \mathbb{Z}[\mathbf{x}]$ generated by B and its relation to $I_0 = I \cdot \mathbb{Q}[\mathbf{x}]$ and to $I_p = I \cdot \mathbb{Z}_p[\mathbf{x}]$ for different primes p. Since any $f \in I_0 \cap \mathbb{Z}[\mathbf{x}]$ can be written in the form $f = \sum_i \frac{h_i}{n} f_i$ with $h_i \in \mathbb{Z}[\mathbf{x}]$ and a common denominator $0 \neq n \in \mathbb{Z}$ we see that for a proper definition of $\Sigma(I)$ we get $\Sigma = \Sigma(I) = \Sigma(I_0)$. This is no more correct for I_p, since given $m \in \Sigma$ and $f \in I$ with $lt(f) = m$ it may happen that $lc(f) \equiv 0 \pmod{p}$. We say that p is a *lucky prime* if $\Sigma(I) = \Sigma(I_p)$.

For $m \in \Sigma$ define $C_m = \gcd\,(lc(f) \ : \ f \in I, \ lt(f) = m)$. Obviously p is a lucky prime if it does not divide any of these numbers. Since $C_m | C_n$ for monomials m, n with $n|m$ we can restrict this divisibility test to $m \in Gen(\Sigma)$. Hence for a given B and term order all but a finite number of primes are lucky.

Choosing a »random« prime yields the correct »modular trace« of the GBasis computation and further one can try to lift that trace to the integers. (Traverso 1988) proposed to store a trace of the modular computation and recompute the nonzero S-polynomials over the integers to get the GBasis over $\mathbb{Q}$. (Winkler 1988) and (Pauer 1992) gave algorithms for a direct Hensel lifting of the integer results from the modular ones. All these approaches take for granted that p is lucky – but this can be justified only a posteriori. In most cases it is justified, especially if p is not too small.

(Graebe 1993) proposed a multimodular approach that detects unlucky primes »on the way« if $lc(f) \equiv 0 \pmod{p}$ for only some of the multimodular »slots« p.

More recent approaches in that direction by Faugère use fast binary arithmetics to »lay a track« and pick up that trail.

12 Hilbert Series and Hilbert Driven GB Computation

Die Menge $[R]_d$ der homogenen Polynome vom Grad d ist ein k-Vektorraum Das Nullpolynom ist homogen von jedem Grad und gehört damit zu jeder dieser Mengen. R ist als k-Vektorraum die (unendliche) direkte Summe dieser homogenen Komponenten: $R = \oplus_d [R]_d$. Wir können homogene R-Module (H-Module) M betrachten, die eine ähnliche Struktur $M = \oplus_d [M]_d$ mit $[M]_d = 0$ für $d \ll 0$ haben. Insbesondere ist jeder Faktorring R/I nach einem homogenen Ideal ein solcher Modul.

Definition 6 *Die erzeugende Funktion*

$$H(M,t) = \sum_{d \in \mathbb{Z}} dim_k([M]_d)\, t^d$$

bezeichnet man als die Hilbertreihe *des homogenen R-Moduls M.*

Wegen $[M]_d = 0$ für $d \ll 0$ ist diese Reihe eine Laurentreihe (um $t = 0$). In den meisten Fällen gilt sogar $[M]_d = 0$ für $d < 0$ und die Reihe ist eine Potenzreihe. Oft lässt sich die erzeugende Funktion einer Zahlenfolge als Taylorreihe einer analytischen Funktion identifizieren und aus dem Eindeutigkeitssatz über die Koeffizienten der Reihenentwicklung die Zahlenfolge rekonstruieren.

Ist $M[a]$ aus M durch Gradshift entstanden, so gilt

$$H(M[a], t) = t^{-a} \cdot H(M, t).$$

(alt) lässt sich reformulieren als

$$\sum_i (-1)^i H(M_i, t) = 0. \tag{alt'}$$

Damit können wir als erstes Ergebnis eine Formel für die Hilbertreihe des Polynomrings $k[x_1, \ldots, x_n]$ herleiten.

Theorem 17 *Für $R = k[x_1, \ldots, x_n]$ gilt*

$$H(R, t) = \frac{1}{(1-x)^n}$$

Beweis: Mit Induktion. Für $n = 1$ ist die Aussage offensichtlich. Für den Beweis des Induktionsschritts sei $R' = k[x_1, \ldots, x_{n-1}]$ und $H(R', t) = \frac{1}{(1-x)^{n-1}}$ bekannt. Wir betrachten die exakte H-Sequenz

$$0 \longrightarrow R[-1] \xrightarrow{\cdot x_n} R \longrightarrow R/(x_n) \cong R' \longrightarrow 0$$

aus welcher wegen (alt') sofort $(1-t)H(R, t) = H(R', t)$ folgt. $\square$

Die Idee dieses Beweises kann sofort auf den Fall eines Hauptideals verallgemeinert werden. Sei $f \in R$ ein homogenes Polynom vom Grad d, so hasen wir die exakte Sequenz

$$0 \longrightarrow R[-d] \xrightarrow{\cdot f} R \longrightarrow R/(f) \longrightarrow 0$$

und erhalten wie oben

$$H(R/(f), t) = \left(1 - t^d\right) H(R, t).$$

Dabei spielte die Nullteilerfreiheit von R für die Exaktheit der Sequenz an der ersten Stelle eine wichtige Rolle:

$$Ker(\cdot f) = \{g \in R \ : \ f \cdot g = 0\} = \{0\}.$$

Für einen allgemeineren Ring $\tilde{R} = R/I$, wobei I ein homogenes Ideal ist, gilt

$$Ker\left(R \xrightarrow{\cdot f} \tilde{R}\right) = \{g \in R \ : \ f \cdot g \in I\} = I : (f).$$

Damit ist

$$0 \longrightarrow R/(I : (f))[-d] \xrightarrow{\cdot f} R/I \longrightarrow R/(I + (f)) \longrightarrow 0$$

eine exakte Sequenz und wir erhalten die Beziehung

$$H(R/(I + (f)), t) = H(R/I, t) - t^d H(R/(I : (f)), t) \tag{HQR}$$

zur Berechnung der Hilbertreihe allgemeiner Ideale.

Ist insbesondere f ein Nichtnullteiler bzgl. I, also $I : (f) = I$, so gilt wie oben

$$H(R/(I + (f)), t) = \left(1 - t^d\right) H(R/I, t).$$

Ist G eine Gröbnerbasis von I, so bildet die Menge $N(G) = T \setminus \Sigma(I)$ der Standardterme gerade eine solche k-Vektorraum-Basis, so dass

$$H(R/I, t) = \sum_{d \in \mathbb{Z}} |[N(G)]_d| \; t^d$$

gilt. Da die Menge der Standardterme nur von $\Sigma(I)$ abhängt, gilt auch $H(R/I, t) = H(R/Lt(I))$, so dass wir die Berechnung der Hilbertreihe allgemeiner Ideale auf die von PP-Idealen zurückführen können.

Mit der Formel (HQR) hatten wir bereits einen rekursiven Ansatz zur Berechnung der Hilbertreihe bei gegebener Basis formuliert, der aber mehrfache Berechnung von Idealquotienten erfordert. Dieser Ansatz ist für PP-Ideale besonders einfach auszuführen. Für unser Standardbeispiel ist

$$I = \left\{x^2 + y + z - 3, y^2 + x + z - 3, z^2 + x + y - 3\right\}$$

bgzl. der Ordnung degLex bereits eine Gröbnerbasis und wir erhalten aus (HQR)

$$H(R/I, t) = \frac{(1 - t^2)^3}{(1 - t)^3} = (1 + t)^3 = t^3 + 3\,t^2 + 3\,t + 1$$

Diese entspricht der Unterteilung der Standardterme

$$1 \mid x, y, z \mid xy, xz, yz \mid xyz$$

nach dem Grad.

Bezüglich der lex. Termordnung ergab sich $Lt(I) = \left\{x, y^2, yz^2, z^6\right\}$ und die Anwendung von (HQR) induziert die folgenden Rechnungen:

		$H(R, t) = \frac{1}{(1-t)^3}$
(x)		$H(R/x, t) = \frac{1-t}{(1-t)^3} = \frac{1}{(1-t)^2}$
(x, y^2)	$(x) : (y^2) = (x)$	$H(R/(x, y^2), t) = \frac{1-t^2}{(1-t)^2} = \frac{1+t}{1-t}$
(x, y^2, z^6)	$(x, y^2) : (z^6) = (x, y^2)$	$H(R/(x, y^2, z^6), t) = \frac{(1-t^6)(1+t)}{1-t} = t^6 + 2\,t^5 + \ldots t + 1$

Im letzten Schritt schließlich ergibt sich wegen $(x, y^2, z^6) : (yz^2) = (x, y, z^4)$

$$\begin{aligned} H(R/I, t) &= H(R/(x, y^2, z^6), t) - t^3\, H(R/(x, y, z^4), t) \\ &= t^6 + 2t^5 + 2t^4 + 2t^3 + 2t^2 + 2t + 1 - t^3\left(1 + t + t^2 + t^3\right) \\ &= t^5 + t^4 + t^3 + 2\,t^2 + 2\,t + 1 \end{aligned}$$

was auch hier genau der Verteilung der Standardterme nach Graden entspricht:

$$1 \mid y, z \mid yz, z^2 \mid z^3 \mid z^4 \mid z^5$$

For homogeneous ideal in many cases the Hilbert series is known in advance. In this case the computation of S-Polynomials in degree d can be terminated if $[Lt(\Sigma)]_d$ has the correct k-dimension. This version of Buchbergers algorithmus is called **Hilbert Driven Algorithm**.

13 GB Computation Facilities for Selected CAS

Alle großen Computeralgebrasysteme allgemeiner Ausrichtung (CAS) erlauben die Berechnung von Gröbnerbasen wenigstens für die lexikografische Termordnung. Die folgenden Ausführungen sollen einen kurzen Überblick über die Möglichkeiten ausgewählter Systeme geben.

CoCoA, Singular, Macaulay

The tree most advanced systems that implement also a great variety of algorithms from different areas of commutative algebra and algebraic geometry.

They use the concept of a `CurrentRing`, that encapsulates informations about the coefficient domain, variables and term order. All computations are executed wrt. this CurrentRing that is present either explicitely as global object or implicitely as attached to every ideal and module.

Here is a typical computation in CoCoA.

Da Berechnungen von Gröbnerbasen recht aufwändig sein können, werden die Ergebnisse zu einzelnen Idealen zwischengespeichert, wenn diese einem Bezeichner zugeordnet sind. Die zu einem Bezeichner aktuell gespeicherten Informationen können mit `Describe` abgefragt werden.

```
Use R::=Q[y,x],Lex;
B0:=[17x^2+22xy+13y^2-1, 8x^2+28xy+37y^2-1];
B1:=Ideal(B0);
B1.GBasis;
Null
------------------------------
GBasis(B1);
[4/75y + 25/36x^3 - 11/225x, -625/48x^4 + 25/12x^2 - 4/75]
------------------------------
B1.GBasis;
[4/75y + 25/36x^3 - 11/225x, -625/48x^4 + 25/12x^2 - 4/75]
------------------------------
Describe B1;
Record[Type = IDEAL,
Value = Record[Gens = [13y^2 + 22yx + 17x^2 - 1, 37y^2 + 28yx + 8x^2 - 1],
GBasis = [4/75y + 25/36x^3 - 11/225x, -625/48x^4 + 25/12x^2 - 4/75]]]
```

Es gibt in CoCoA verschiedene Möglichkeiten, Gröbnerbasisrechnungen zu steuern und zu verfolgen, auf die hier nicht näher eingegangen werden soll. Die wichtigsten Kommandos zu Normalformen und Gröbnerbasen sind:

`NR(F,B)`	(totale) Normalform eines Polynoms F bzgl. einer Liste B.
`Interreduce(B)`	Interreduktion einer Liste von Polynomen (in situ).
`I:=Ideal(B)`	Basis B einem Ideal I zuordnen.
`GBasis(I)`	Gröbnerbasis von I berechnen und unter dem Bezeichner I speichern.
`ReducedGBasis(I)`	Reduzierte Gröbnerbasis von I berechnen.
`NF(F,I)`	(totale) Normalform eines Polynoms F bzgl. eines Ideals I, wobei – wenn erforderlich – eine Gröbnerbasis von I berechnet und unter dem Bezeichner I gespeichert wird.
`I.Gens`	Idealbasis eines Ideals I als Liste.
`I.GBasis`	Bereits berechnete Gröbnerbasis eines Ideals I als Liste.

Maple 9.5

Maple 9.5 supplies an advanced implementation of GBasis facilities and flexible term ordering definitions.

The Groebner package is a collection of commands for doing Groebner basis calculations in skew algebras like Weyl and Ore algebras and in corresponding modules like D-modules. It can also be used in the case of usual commutative polynomials. Such calculations are also available in case of algebraic and modular coefficients and for modules.

The `gbasis` command computes a (reduced) Groebner bases either in an algebra of skew polynomials (an Ore algebra) or in an algebra of usual polynomials. It also computes Groebner bases for modules over (usual or skew) polynomial rings. The `pretend_gbasis` command makes a Groebner basis known to the system without performing any computation.

The `spoly` command computes the S-polynomial of two polynomials.

The `univpoly` command finds univariate polynomials of least degree in a polynomial ideal.

The `fglm_algo` command is a general-purpose and parametrizable iteration algorithm based on the FGLM algorithm.

The commands `hilbertdim`, `hilbertpoly` and `hilbertseries` compute the Hilbert dimension, Hilbert polynomial and Hilbert series of an ideal, respectively.

Applications in the example section:

- Legendre polynomials. From a differential equation and a mixed differential-difference equation that define them, elimination of Dx yields a difference equation that vanishes on them.
- Lauricella function. It illustrates Groebner basis computation in D-modules with parameters in the ground field.
- A third-order differential system of four equations.

Mathematica 5.2

Two standard versions

- `GroebnerBasis[polys,vars]` – standard lex. GBasis computation

- `GroebnerBasis[polys,vars,elimvars]` – GBasis for variable elimination

Special options are available to customize the computation

- `MonomialOrder`: Possible settings are Lexicographic, DegreeLexicographic, DegreeReverseLexicographic or an explicit weight matrix.
- `CoefficientDomain`: Possible settings for CoefficientDomain are InexactNumbers, Rationals, RationalFunctions and Polynomials[x].
- `Modulus`: GBasis over modular coefficients

For version 6 more options (SelectionStrategy, Method) are announced.

MuPAD 3.2

The groebner package contains some functions dealing with ideals of multivariate polynomial rings over a field. In particular, Gröbner bases of such ideals can be computed.

An ideal is given by a list of generators of the special type `poly` in the same ring. The generators may also be expressions (all of them must be, if any of them is); they are understood as polynomials over the rationals in this case, and thus all of their coefficients must be rational. Polynomials over Expr are also allowed if all of their coefficients are rationals.

Gröbner bases and related notions depend on the monomial ordering (also called term ordering) under consideration. MuPAD knows the following orderings:

- the lex. ordering, denoted by the identifier LexOrder.
- the ordering `deglex`, denoted by the identifier DegreeOrder.
- the ordering `degrevlex`, denoted by DegInvLexOrder.
- user-defined orderings. They constitute a domain Dom::MonomOrdering of their own.

Available functions:

- groebner::dimension – the dimension of the affine variety generated by polynomials
- groebner::eliminate – eliminate variables
- groebner::gbasis – computation of a reduced Gröbner basis
- groebner::normalf – complete reduction modulo a polynomial ideal
- groebner::spoly – the S-polynomial of two polynomials
- groebner::stronglyIndependentSets – strongly independent set of variables

Reduce

Reduce ist unter den großen Systemen das am schlechtesten dokumentierte. Dafür sind seine Quellen (fast) vollständig zugänglich.

In Reduce stehen zwei größere Pakete zur Berechnung von Gröbnerbasen zur Verfügung, das Paket `groebner` von Melenk, Möller, Neun und das von Gräbe entwickelte Paket `Cali`. Beide Pakete sind als Nutzerbeiträge klassifiziert, obwohl das Paket `groebner` auch von Kernfunktionen wie etwa `solve` verwendet wird. Zu jedem der beiden Pakete gibt es im Dokumentationsteil LaTeX-Beschreibungen und ein mehr oder weniger ausführliches Testfile, das die Möglichkeiten der wichtigsten Kommandos erläutert.

Axiom

- gb.spad: package GB GroebnerPackage (Gebauer, Trager)

  Computes groebner bases for polynomial ideals. The basic computation provides a distinguished set of generators for polynomial ideals over fields. When the provided coefficient domain is not a field, the result is equivalent to considering the extended ideal with `Fraction(Dom)` as coefficients, but considerably more efficient since all calculations are performed in Dom.

- gbeuclid.spad: package GBEUCLID EuclideanGroebnerBasisPackage (Gebauer, Moeller)

  Computes groebner bases for polynomial ideals over euclidean domains.

- groebf.spad: package GBF GroebnerFactorizationPackage (Moeller, Grabmeier)

- groebsol.spad: package GROEBSOL GroebnerSolve (Gianni)

  Solve systems of polynomial equations using Groebner bases. Total order Groebner bases are computed and then converted to lex ones. This package is mostly intended for internal use.

- ideal.spad: domain IDEAL PolynomialIdeals (Gianni)

  This domain represents polynomial ideals with coefficients in any field and supports the basic ideal operations, including intersection sum and quotient.

- idecomp.spad: package IDECOMP IdealDecompositionPackage (Gianni)