

# Slimgb

Gröbner bases with  
**slim** polynomials

# The Aim

avoid intermediate expression swell

Classical Buchberger algorithm with parallel reductions  
guided by new weighted length functions

- Often: big computations  $\rightarrow$  small results
- experiments show: blow up often avoidable
- use combination of several techniques
- no reduction to other methods (like Walk, homogenization, mod  $p$  approaches)

# Classical measures of expression swell

- Length (number of terms)
- Ecart (max degree - lead degree)
- Coefficient size

**Need them all!!!**

# Weighted Length

- $\mathbb{Z}/p[x_1, \dots, x_n]$ : length
- $\mathbb{Q}[x_1, \dots, x_n]$ :  $\text{size}(\text{lc}(p)) * \text{length}(p)$

- elimination orderings:

$$\sum_{m \in \text{supp}(p)} (1 + |\text{deg}(m) - \text{deg}(\text{lm}(p))|^{+})$$

where  $\text{supp}(p)$  are the monomials of  $p$   
e.g. for  $x > y$ :  $\text{wlen}(x^3 + y^5) = 1 + (1 + 5 - 3) = 4$

# Meaning for slimgb -

weighted length controls everything

- sort basis elements for reduction of S-Polynomials
- choice of critical pairs (together with Sugar)
- additional reductors
- exchange basis elements by “w. shorter” ones

# Extended product criterion

Theorem:  $f, g, m$  polynomials,  $\gcd(\text{lm}(f), \text{lm}(g)) = l$   
then  $\text{spoly}(m*f, m*g)$  has standard  
representation against  $\{m*f, m*g\}$

Proof:

- apply normal product criterion to  $f, g$
- multiply with  $m$

# Application of extended product criterion

- common factor  $m$  is a monomial
- for every basis  $f$  element compute  $\gcd m_f$  of its terms
- pull out  $\gcd(m_f, m_g)$
- $f = x*y^2 + x*y, g = x*z + x, x > y > z$  (lex)
- $m_f = xy, m_g = x, \gcd(y^2, z) = 1$

# nontrivial t-representations

- Let  $\text{spoly}(p,q) = g_1 * f_1 + \dots + g_n * f_n$  (\*)
- Let  $t$  monomial,  $t \geq \text{lm}(g_i) * \text{lm}(f_i)$  for all  $i$
- this is called a  $t$ -representation
- of course  $t \geq \text{lm}(\text{spoly}(p,q))$
- $t = \text{lm}(\text{spoly}(p,q))$ : (\*) standard representation
- $t < \text{lcm}(\text{lm}(p), \text{lm}(q))$ : (\*) nontrivial  $t$ -representation (or nontrivial syzygy)

# Example

- $f=xy+1$   
 $g=x^2+1$   
 $h=x$
- Ordering  $\text{lp}: x > y$
- $s := \text{spoly}(f, g) = x - y$
- $(*)$ :  $-y * f + (y^2 + 1) * h = s$  is a  $xy^2$ -representation
- $x < xy^2 < x^2y$ :  $(*)$  nontrivial, but not standard

# Theorem

- Let  $G = \{g_1, \dots, g_n\}$
- $G$  is a Gröbner basis iff  
for every  $i, j$ :  $\text{spoly}(g_i, g_j)$  has a nontrivial  $t$ -  
representation (some  $t < \text{lcm}(\text{lm}(g_i), \text{lm}(g_j))$ )
- Buchberger criterion:  $t = \text{lm}(\text{spoly}(g_i, g_j))$

cf. Becker, Weißpfenning

---

**Algorithm 1** slimgb main procedure, calculates a Gröbner basis of  $F$

---

**Require:**  $F$  finite tuple of polynomials (from  $K[x_1, \dots, x_n]^r$ ).

$P := \{(i, j) \mid 1 \leq i < j \leq \#F\}$

apply criteria to  $P$

**while**  $P \neq \emptyset$  **do**

  choose  $\emptyset \neq S \subset P$

$P := P \setminus S$

$(R, F) := \text{slimgbReduce}(S, F)$

**for**  $0 \neq r \in R$  **do**

$F := \text{append}(F, r)$

$P := P \cup \{(i, \#F) \mid 1 \leq i < \#F\}$

    apply criteria to  $P$

**end for**

**end while**

return  $F$

---

# slimgbReduce

- reduction of several polynomials
- modifies basis by replacing polynomials by shorter ones with same leading term
- no linear algebra

# Axioms for the reduction algorithm

- Input:  $F$ : Basis,  $S$ : S-Polynomials
- Output:  $F'$  new Basis,  $R$ : reduced S-Polynomials

1.  $\langle F' \cup R \rangle_{K[x_1, \dots, x_n]} = \langle F \rangle_{K[x_1, \dots, x_n]}$ ,

2.  $\#F = \#F'$ ,

3.  $F'$  preserves the order of  $F$ :  $\forall i : \text{lm}(F[i]) = \text{lm}(F'[i])$  and  $\text{spoly}(F[i], F'[i])$  has standard representation w. r. t.  $F' \cup R$ ,

4. each  $r \in S$  has a nontrivial t-representation w. r. t.  $F' \cup R$ ,

5. and for termination:  $R \neq \emptyset \Rightarrow \exists r \in R : \text{lm}(r) \notin L(F)$ .

# Sketch of proof (main)

- Algorithm terminates as usual: termination property of `slimgb reduce`
- the algorithm ensures, that every pair gets nontrivial  $t$ -representation at some time
- if you exchange an element of  $F$  the old one has still a standard representation
- in particular the property of having a nontrivial  $t$ -representation is never lost

# Gaussian-like algorithm + extra operations

Each polynomial in  $S$  corresponds to a row in Gauss

- $f, g$  in  $S$ ,  $\text{lm}(f) = \text{lm}(g)$ :  $g \rightarrow \text{spoly}(f, g)$
- $f$  in  $F$ ,  $g$  in  $S$ :  $\text{lm}(f) | \text{lm}(g)$ :  $g \rightarrow \text{spoly}(f, g)$
- $f$  in  $F$ ,  $g$  in  $S$ :  $\text{lm}(f) = \text{lm}(g)$ : replace  $f$  in  $F$  by  $g$ ,  $g$  in  $S$  by  $\text{spoly}(f, g)$

Every choice is controlled by the weighted length

# Sketch of proof for reduction

- All properties hold at the beginning, except

$$R \neq \emptyset \Rightarrow \exists r \in R : \text{lm}(r) \notin L(F)$$

- The other properties are preserved in each step
- No element with leading term in  $L(F)$  can remain, as there is an operation to reduce it

# Example

```
ring r = 32003, (x,y,z,t,u,v,w,a,b,c,d,e,f,g,h,i,j,k), dp;  
ideal i0=x3-x2y+x2z+xt-7uv+8xa+bc+gh+ij+vw+ak+tu,  
a2b-x2,  
abx-1,  
....
```

- at some points computes  $\text{spoly}(a^2b-x^2, abx-1) = x^3+a$
- same leading term as  $x^3-x^2y+\dots$
- put  $x^3+a$  at the place of  $x^3-x^2+\dots$  (in the basis) and vice versa
- critical pairs are updated automatically (only indices)
- use  $x^3+a$  for reduction of  $x^3-x^2y+\dots$

# Strengths of slimgb

- function fields
- elimination orderings
- rational numbers
- noncommutative rings
- treats also the case of modules

# Implementation in Singular

- uses same low level functions and data structures as std (Buchberger)
- slightly tuned result of a diploma thesis
- still much room for optimization
- very easy to implement efficiently compared to F4

# rational numbers

	Var.	Gen.	slimgb	std
Turaev / Viro 3 colors	44	1661	1 min 100MB	> 1d > 400MB
Turaev / Viro m3n1OrAns+	111	10159	20 min	> 1 week
Turaev / Viro m4n1UnorAnsSimpl	53	892	1h	> 1 week
Diaz1	7	9	0.25s 5,8MB	>45h >1GB

measured on Dual Opteron 2,2GHz 16GB RAM, Singular-3-0-1/CVS

# Smaller Coefficients

	<b>slimgb</b>	<b>std</b>
<b>time</b>	<b>0.42s</b>	<b>243.49s</b>
<b>aver. inter. coeff. size</b>	<b>109</b>	<b>1188</b>
<b>multiplications</b>	<b>1190</b>	<b>3867</b>

Chou 274 2 over  $F_{32003}(p_1, \dots, p_5)[x_1, \dots, x_7]$ , Singular-2-1-99

# Function fields

	O.	Ch.	Par.	Var.	Slimgb		Std	
					s	MB	s	MB
H. Simson 3	dp	p	4	10	128	128	> 66735	>13160
H. Butterfly 1	dp	p	4	8	0.86	6.1	110	103
Chou 303 1	lp	p	5	8	2.6	6.3	>158370	>2200
Chou 274 2	lp	p	4	7	102	26.8	>163709	>500
Chou 302 1	lp	p	5	8	2.32	9.8	>150634	>2700

p:=32003

# Further Elimination

	O.	Ch.	Var.	Slimgb		Std	
				s	MB	s	MB
Katsura 6	lp	p	7	0.19	2.5	> 1819	>21000
Katsura 5	lp	p	5	0.01	0.8	1.8	69.8
ZeroDim 57	lp	p	8	0.3	3.0	>1591	>15000
ZeroDim 29	lp	p	8	0.03	0.8	>1451	>15000

# Noncommutative

	O.	Ch.	Par.	Var.	Slimgb	Std
ucha 2	prod.	0	0	7	5,6s 5,9MB	16m 332MB
ucha 4	lp	0	0	6	0.01s 0,6MB	0,27 0,6MB
tarasov 2	dp	0	2	4	1,45min 26MB	>3,5h >2,4GB
bern5	prod.	0	0	6	2,11min 16,8 MB	>2h >7,5GB