UNIVERSITY OF AARHUS
DEPARTMENT OF MATHEMATICS

HOMOGENEOUS BUCHBERGER ALGORITHMS AND SULLIVANT'S COMPUTATIONAL COMMUTATIVE ALGEBRA CHALLENGE

by Niels Lauritzen

Homogeneous Buchberger algorithms and Sullivant's computational commutative algebra challenge

Niels Lauritzen

Web: `home.imf.au.dk/niels`   Email: `niels@imf.au.dk`

Abstract

We give a variant of the homogeneous Buchberger algorithm for positively graded lattice ideals. Using this algorithm we solve the Sullivant computational commutative algebra challenge[1].

1 Introduction

Suppose that I is a homogeneous ideal in a polynomial ring $R = k[x_0, \ldots, x_n]$ over a field k. The usual homogeneous Buchberger algorithm builds a Gröbner basis for I by successively constructing truncated Gröbner bases of increasing degrees. Suppose that I is *saturated* i.e. $I = \bar{I} = \{g \in R \mid (x_0 \cdots x_n)^m g \in I, m \gg 0\}$. If we encounter a polynomial f divisible by a variable in degree d of the homogeneous Buchberger algorithm, then we may conclude that f reduces to zero modulo the already constructed truncated Gröbner basis in degree $< d$ for I. This simple observation also allows for detection of non-saturated ideals in some cases.

Sullivant's challenge is about deciding if a specified set B of $145,512$ binomials generate the kernel P of the (toric) ring homomorphism $\varphi : k[x_{ijk}] \to k[u_{ij}, v_{ik}, w_{jk}]$ given by

$$\varphi(x_{ijk}) = u_{ij}v_{ik}w_{jk},$$

where $1 \le i, j, k \le 4$. We give a version of the homogeneous Buchberger algorithm with a Gebauer-Möller criterion specifically tailored to positively graded lattice ideals. Using an implementation of this algorithm in the software package `GLATWALK`[2] we deduce that *the ideal J generated by B is strictly contained in P* by showing that J cannot be saturated. In fact, we exhibit a specific binomial b of degree 14 in $\bar{J} \setminus J$.

I am grateful to B. Sturmfels for stimulating my interest in Sullivant's computational commutative algebra challenge. R. Hemmecke has made me aware that he and P. Malkin already computed the full Gröbner basis of J using new algorithms in a new version of `4ti2` thereby answering Sullivant's challenge. In fact they prove that the "missing" binomials in Sullivant's challenge have degree 14 and form an orbit under the action of a certain symmetry group. I am grateful to Hemmecke for verifying that b lies in this orbit.

---

[1] `http://math.berkeley.edu/~seths/ccachallenge.html`
[2] `http://home.imf.au.dk/niels/GLATWALK`

2 Preliminaries

We let $R = k[x_1, \ldots, x_n]$ denote the ring of polynomials over a field k. We assign degrees to the variables by $\deg(x_1) = a_1, \ldots, \deg(x_n) = a_n$, where $a_1, \ldots, a_n$ are positive integers. A monomial $x^v \in R$ has degree $\deg(x^v) = v_1 a_1 + \cdots + v_n a_n$, where $v = (v_1, \ldots, v_n)$. This gives the (positive) grading

$$R = \oplus_{s \geq 0} R_s,$$

where $R_s = \mathrm{span}_k\{x^v \mid \deg(x^v) = s\}$. For a monomial order $\prec$ on R and a subset $S \subset R$ we let $\mathrm{in}_\prec(S) = \{\mathrm{in}_\prec(f) \mid f \in S \setminus \{0\}\}$. A Gröbner basis for an ideal $I \subset R$ is a finite subset $G \subset I$, such that $\langle \mathrm{in}_\prec(G) \rangle = \mathrm{in}_\prec(I)$.

2.1 Truncated Gröbner bases

For a homogeneous ideal I in R and $d \in \mathbb{N}$ we let

$$I_{<d} = \bigoplus_{s<d} I_s.$$

A d-truncated Gröbner basis for I is a finite subset $G_{<d} \subset I_{<d}$, such that $\langle \mathrm{in}_\prec(G_{<d}) \rangle_{<d} = \mathrm{in}(I)_{<d}$ i.e. we require only match of initial ideals up to degree d. Using the division algorithm it is easy to show that $f \in I_{<d}$ reduces to zero modulo the polynomials in a d-truncated Gröbner basis for I.

3 The homogeneous Buchberger algorithm with sat-reduction

We call an ideal I *saturated* if $I = \bar{I} = \{g \in R \mid (x_0 \cdots x_n)^m g \in I, m \gg 0\}$. This means that $mf \in I$ implies $f \in I$, where m is a monomial and f a polynomial in R. Let $\prec$ be a term order on R. For a polynomial $f \in R$ we let $\mathrm{sat}(f)$ denote f divided by the greatest common divisor of the monomials in f. We say that f *sat-reduces* to h modulo g if either $h = \mathrm{sat}(f)$ and $\deg(h) < \deg(f)$ or f reduces to h modulo g in the usual sense i.e. $\mathrm{in}_\prec(g)$ divides a term t in f and

$$h = f - (t/\mathrm{in}_\prec(g))g.$$

Notice that if f sat-reduces to h modulo g and f, g belongs to a saturated ideal I, then $h \in I$. A remainder in the division algorithm of f modulo a set of polynomials G using sat-reduction in each step is denoted $f^{G(\mathrm{sat})}$.

The S-polynomial of two homogeneous polynomials is homogeneous of degree no less than the degrees of the polynomials. The (usual) reduction of a homogeneous polynomial of degree d modulo a set of homogeneous polynomials gives a homogeneous polynomial of degree d. These observations give the homogeneous Buchberger algorithm as explained in ([1], Theorem 11). We tailor the homogeneous Buchberger algorithm to the special case where input consists of a set $B = \{f_1, \ldots, f_r\} \subset R$ of homogeneous polynomials generating a saturated ideal. This has the consequence that reduction of a homogenous polynomial f of degree d divisible by a variable x_i is not necessary, since $f/x_i \in I_{<d}$ reduces to zero using the already computed d-truncated Gröbner basis $G_{<d}$.

Algorithm 3.1 (Homogeneous Buchberger algorithm for saturated ideals)

INPUT: Term order $\prec$. Homogeneous polynomials $B = \{f_1, \ldots, f_r\} \subset R$ generating a saturated homogeneous ideal I.

OUTPUT: Homogeneous polynomials $G = \{g_1, \ldots, g_s\}$ such that $\{g_1, \ldots, g_s\}$ is a minimal Gröbner basis over $\prec$ for the ideal generated by B.

(i) $Spairs := \emptyset$; $G := \emptyset$;

(ii) while $(B \neq \emptyset$ or $Spairs \neq \emptyset)$ do

    (a) Extract[3] a polynomial f of minimal degree in $B \cup Spairs$.

    (b) Compute $g := f^{G(\text{sat})}$, continue if the degree drops in a sat-reduction step in the division algorithm;

    (c) if $(g = 0)$ continue;

    (d) $G := G \cup \{g\}$;

    (e) Append S-polynomials $S(g, h)$ to $Spairs$ for every $h \in G \setminus \{g\}$.

Remark 3.2 After step (iid) in Algorithm 3.1, the polynomials of degree $< d$ in G form a minimal d-truncated Gröbner basis of I, where d is the minimal degree of the polynomials in $B \cup Spairs$.

An easy modification to algorithm (3.1) may detect if I is not saturated. If the sat-reduction $f^{G(\text{sat})}$ of f is non-zero and has lower degree than f, then we may deduce the existence of a monomial x^v and a polynomial g such that $x^v g \in I$, but $g \notin I$.

Algorithm 3.3 (Homogeneous Buchberger algorithm with saturation check)

INPUT: Homogeneous polynomials $B = \{f_1, \ldots, f_r\}$ and a term order $\prec$.

OUTPUT: Homogeneous polynomials $G = \{g_1, \ldots, g_s\}$ such that $\{g_1, \ldots, g_s\}$ is a minimal Gröbner basis over $\prec$ for the ideal I generated by B or proof that I is not saturated.

(i) $Spairs := \emptyset$; $G := \emptyset$;

(ii) while $(B \neq \emptyset$ or $Spairs \neq \emptyset)$ do

    (a) Extract a polynomial f of minimal degree d in $B \cup Spairs$.

    (b) $g := f^{G(\text{sat})}$;

    (c) if $(g = 0)$ continue;

    (d) if$(\deg(g) < d)$

        (i) OUTPUT f as proof that I is not saturated and HALT.

    (e) $G := G \cup \{g\}$;

    (f) Append S-polynomials $S(g, h)$ to $Spairs$ for every $h \in G \setminus \{g\}$.

Example 3.4 We give a simple example illustrating algorithm (3.3).

(i) Consider the input $B = \{xz - y^2, x^4 - y^3\}$ along with the reverse lexicographic term order $x \prec y \prec z$.

(ii) The ideal I generated by B is homogeneous in the grading $\deg(x) = 3, \deg(y) = 4, \deg(z) = 5$ and $\deg(xz - y^2) = 8 < \deg(x^4 - y^3) = 12$.

---

[3]This means that f is deleted from the relevant list after it is extracted

(iii) After the first loop we have $B = \{y^3 - x^4\}, G = \{y^2 - xz\}$ and $Spairs = \emptyset$, where G is a 12-truncated Gröbner basis of I.

(iv) In the second loop we sat-reduce $y^3 - x^4$ modulo $y^2 - xz$ and get $yz - x^3$. As $\deg(yz - x^3) = 9 < \deg(y^3 - x^4) = 12$, we conclude that I is not saturated.

(i) Now suppose that $B = \{y^2 - xz, yz - x^3\}$ in the same grading.

(ii) After the second loop we have

$$B = \emptyset$$
$$G = \{y^2 - xz, yz - x^3\}$$
$$Spairs = \{yx^3 - z^2x\},$$

where G is a 13-truncated Gröbner basis of I.

(iii) Now $yx^3 - z^2x$ sat-reduces to $z^2 - yx^2$ modulo G. We conclude that I is not saturated.

(i) Now proceed with $B = \{y^2 - xz, yz - x^3, z^2 - yx^2\}$.

(ii) After a few loops we have

$$B = \emptyset$$
$$G = \{y^2 - xz, yz - x^3, z^2 - yx^2\}$$
$$Spairs = \{y^2x^2 - zx^3\},$$

where G is a 14-truncated Gröbner basis of I. Since $y^2x^2 - zx^3$ sat-reduces to zero, G is the reduced Gröbner basis of I.

The number of S-pairs considered for reduction can be reduced drastically by using a version of the Gebauer-Möller criterion in algorithms (3.1) and (3.3). The framework for properly explaining the Gebauer-Möller criterion is in the context of Gröbner bases for modules (cf. [1], §4).

3.1 The Gebauer-Möller criterion

Let $e_1, \ldots, e_m$ denote the canonical basis of the finitely generated free module $F = R^m$. A monomial in F is an element $x^\nu e_i$, where x^ν is a monomial in R. Every element in F is a k-linear combination of monomials. By definition a monomial $x^\alpha e_i$ divides a monomial $x^\beta e_j$ if and only if $i = j$ and x^α divides x^β in R. We write this as $x^\alpha e_i \mid x^\beta e_j$. A monomial order on F is a total order $\prec$ on monomials in F satisfying

$$x^\alpha e_i \prec x^\beta e_j \implies x^{\alpha+\gamma} e_i \prec x^{\beta+\gamma} e_j$$

for every $i, j = 1, \ldots, m$ and $\alpha, \beta, \gamma \in \mathbb{N}^n$. We let $\mathrm{in}_\prec(f)$ denote the largest monomial in f. Now the Gröbner basics for ideals in R can be generalized to submodules of F almost verbatim. For a subset $B \subset F$ we let $\mathrm{in}_\prec(B)$ denote the submodule generated by $\mathrm{in}_\prec(f)$, where $f \in B$. A Gröbner basis of a submodule $M \subset F$ is a set of elements $G = \{m_1, \ldots, m_t\} \subset M$ satisfying $\mathrm{in}_\prec(M) = \mathrm{in}_\prec(G)$. It is called minimal if $\mathrm{in}_\prec(m_i) \nmid \mathrm{in}_\prec(m_j)$ for $i \neq j$. We will use Gröbner bases for submodules in reasoning about syzygies of monomial ideals. Consider a monomial ideal

$$M = \langle x^{\nu_1}, \ldots, x^{\nu_m} \rangle \subset R.$$

4

The syzygies of M are the relations in M i.e. the kernel K of the natural surjection $R^m \to M$. Now consider the $\mathbb{Z}^n$-grading $\deg(x_i) = e_i$ on R. Then K is a homogeneous submodule of F in the $\mathbb{Z}^n$-grading given by $\deg(e_i) = v_i$. A natural set of homogeneous generators are

$$S_{ij} = x^{v_i \vee v_j - v_j} e_j - x^{v_i \vee v_j - v_i} e_i \in K$$

for $1 \leq i < j \leq m$ (see [2], Proposition 2.8). Define a monomial order $\prec$ (The Schreyer order) on F by $x^\alpha e_i \prec x^\beta e_j$ if and only if

$$\alpha + v_i < \beta + v_j \text{ or } \alpha + v_i = \beta + v_j \text{ and } i < j,$$

where $<$ is any term order on R. Then we have the following

Proposition 3.5 The homogeneous generating set $\{S_{ij} \mid 1 \leq i < j \leq m\}$ is a Gröbner basis for K over the Schreyer order $\prec$.

The Gröbner basis in Proposition 3.5 is rarely minimal. In view of Theorem 2.9.9 in [2], it suffices to reduce the S-pairs corresponding to a minimal Gröbner basis of the syzygies (in Buchberger's algorithm). This procedure is in fact one of the Gebauer-Möller criteria for cutting down on the number of S-pairs. The point is that this minimization is easy and quite fast to perform in step (iie) of Algorithm 3.1. Suppose that we must update *Spairs* with a non-zero polynomial $g = g_m$, where $G = \{g_1, \ldots, g_{m-1}\}$ in step (iie). We put $x^{v_i} = \mathrm{in}_\prec(g_i)$ for $i = 1, \ldots, m$. Consider the syzygies $S_{1m}, \ldots, S_{m-1,m}$. In the Schreyer order we have $\mathrm{in}_\prec(S_{1m}) = x^{v_1 \vee v_m - v_m} e_m, \ldots, \mathrm{in}_\prec(S_{m-1,m}) = x^{v_{m-1} \vee v_m - v_m} e_m$. Thus the minimization can be done successively in step (iie) by throwing out superfluous monomials among

$$x^{v_1 \vee v_m - v_m}$$

$$\vdots$$

$$x^{v_{m-1} \vee v_m - v_m}.$$

This can be implemented as below ($u \leq v$ means that $v - u \in \mathbb{N}^n$ for $u, v \in \mathbb{N}^n$), where (iia) represents the usual criterion, where leading terms are relatively prime (cf. [2], Proposition 2.9.4).

Algorithm 3.6 updateSpairs:

(i) $MinSyz := \emptyset$;

(ii) for each v_i in $\{v_1, \ldots, v_{m-1}\}$ do

    (a) if ($v_m \wedge v_i = 0$) continue;

    (b) $a = v_i \vee v_m - v_m$;

    (c) if ($w \leq a$ for some $(w, p) \in MinSyz$) continue;

    (d) Delete $(w, p) \in MinSyz$ if $a \leq w$;

    (e) $MinSyz := MinSyz \cup \{(a, S(g_i, g_m))\}$;

(iii) for each $(a, p) \in MinSyz$ do

    (a) $Spairs := Spairs \cup \{p\}$;

4 Lattice ideals

Recall the decomposition of an integer vector $v \in \mathbb{Z}^n$ into $v = v^+ - v^-$, where $v^+, v^- \in \mathbb{N}^n$ are vectors with disjoint support. For $u, v \in \mathbb{N}^n$ we let $u \leq v$ denote the partial order given by $v - u \in \mathbb{N}^n$. For a subset $\mathscr{B} \subset \mathbb{Z}^n$ we associate the ideal

$$I_{\mathscr{B}} = \langle x^{v^+} - x^{v^-} \mid v \in \mathscr{B} \rangle \subset R.$$

In the case where $\mathscr{B} = \mathscr{L}$ is a lattice we call $I_{\mathscr{L}}$ the *lattice ideal* associated to $\mathscr{L}$. If $u - v \in \mathscr{L}$ for $u, v \in \mathbb{N}^n$, then

$$x^u - x^v = x^{u - (u-v)^+} (x^{(u-v)^+} - x^{(u-v)^-}) \in I_{\mathscr{L}}. \tag{1}$$

The binomials $B_{\mathscr{L}} = \{x^u - x^v \mid u - v \in \mathscr{L}\} \subset I_{\mathscr{L}}$ are stable under the fundamental operations in Buchberger's algorithm: forming *S*-polynomials and reducing modulo a subset of $B_{\mathscr{L}}$. This means that starting with a generating set for $I_{\mathscr{L}}$ in $B_{\mathscr{L}}$ we end up with a Gröbner basis consisting of binomials in $B_{\mathscr{L}}$. Reducing a monomial x^w by an element of $B_{\mathscr{L}}$ amounts to replacing x^w by x^{w-v}, where $v \in \mathscr{L}$. Therefore if a binomial $x^u - x^v \in I_{\mathscr{L}}$, then $u - v \in \mathscr{L}$. This proves that $I_{\mathscr{L}}$ is saturated and algorithm (3.1) applies. The simple data structures in the specialization of algorithm (3.1) to lattice ideals are very appealing. If $f = x^u - x^v$, then

$$\text{sat}(f) = x^{(u-v)^+} - x^{(u-v)^-}.$$

by (1). With this in mind we define

$$\text{bin}(w) = x^{w^+} - x^{w^-}$$

for $w \in \mathbb{Z}^n$. Using this notation we have $\text{sat}(\text{bin}(u), \text{bin}(v)) = \text{bin}(u - v)$. Similarly if $v^+ \leq u^+$ we may reduce $\text{bin}(u)$ by $\text{bin}(v)$. This results in a binomial f with $\text{sat}(f) = \text{bin}(u - v)$. Notice that replacing u by $u - v$ if $v^+ \leq u^+$ corresponds to sat-reduction of $\text{bin}(u)$ by $\text{bin}(v)$. We have silently assumed that the initial term of $\text{bin}(w)$ is x^{w^+} for the term order in question. We will keep this convention throughout.

Usually a generating set $\mathscr{B}$ for $\mathscr{L}$ as an abelian group is given. Computing the lattice ideal $I_{\mathscr{L}} \supset I_{\mathscr{B}}$ can be done using that

$$I_{\mathscr{L}} = \bar{I}_{\mathscr{B}}.$$

If $\mathscr{B}$ contains a positive vector, then $I_{\mathscr{B}} = I_{\mathscr{L}}$ ([3], Lemma 12.4). If $\mathscr{L} \cap \mathbb{N}^n = \{0\}$, $I_{\mathscr{L}}$ may be computed from $I_{\mathscr{B}}$ using Gröbner basis computations for different reverse lexicographic term orders ([3], Lemma 12.1).

With these conventions it is quite easy to convert algorithm (3.1) into a specialized algorithm for lattice ideals representing binomials via integer vectors with additional structure (like the degree of $\text{bin}(v)$ and certain other (optimizing) features). We give the straightforward translation of algorithm (3.1) into the lattice case.

Algorithm 4.1 (Homogeneous Buchberger algorithm for lattice ideals)

INPUT: Term order $\prec$. Integer vectors $B = \{v_1, \ldots, v_r\}$ with respect to $\prec$ such that $\langle \text{bin}(v_1), \ldots, \text{bin}(v_r) \rangle$ is a positively graded lattice ideal $I_{\mathscr{L}}$.

OUTPUT: Integer vectors $G = \{w_1, \ldots, w_s\}$ such that $\langle \text{bin}(w_1), \ldots, \text{bin}(w_s) \rangle$ is a minimal Gröbner basis over $\prec$ for $I_{\mathscr{L}}$.

  (i) *Spairs* $:= \emptyset$; $G := \emptyset$;

(ii) while $(B \neq \emptyset$ or $Spairs \neq \emptyset)$ do

    (a) Extract a binomial $\mathrm{bin}(v)$ of minimal degree in $B \cup Spairs$.

    (b) Compute the reduction $\mathrm{bin}(w) := \mathrm{bin}(v)^{G(\mathrm{sat})}$, continue if the degree drops in a sat-reduction step in the division algorithm.

    (c) if $(\mathrm{bin}(w) = 0)$ continue;

    (d) $G := G \cup \{\mathrm{bin}(w)\}$;

    (e) **updateSpairs**

updateSpairs:

  (i) $MinSyz := \emptyset$;

 (ii) for each $\mathrm{bin}(v)$ in $G \setminus \{\mathrm{bin}(w)\}$ do

    (a) if $(w^+ \wedge v^+ = 0)$ continue;

    (b) $a = v^+ \vee w^+ - w^+$;

    (c) if $(u \leq a$ for some $(u, p) \in MinSyz)$ continue;

    (d) Delete $(u, p) \in MinSyz$ if $a \leq u$;

    (e) $MinSyz := MinSyz \cup \{(a, \mathrm{bin}(u - v)\}$;

(iii) for each $(a, \mathrm{bin}(u)) \in MinSyz$ do

    (a) $Spairs := Spairs \cup \{\mathrm{bin}(u)\}$;

Similarly algorithm 3.3 translates into

Algorithm 4.2

INPUT: Term order $\prec$. Normalized integer vectors $B = \{v_1, \ldots, v_r\}$ with respect to $\prec$, such that $\langle \mathrm{bin}(v_1), \ldots, \mathrm{bin}(v_r)\rangle$ generates the ideal I.

OUTPUT: Integer vectors $G = \{w_1, \ldots, w_s\}$ such that $\langle \mathrm{bin}(w_1), \ldots, \mathrm{bin}(w_s)\rangle$ is a minimal Gröbner basis over $\prec$ for I or proof that I is not a lattice ideal.

  (i) $Spairs := \emptyset$; $G := \emptyset$;

 (ii) while $(B \neq \emptyset$ or $Spairs \neq \emptyset)$ do

    (a) Extract a binomial $\mathrm{bin}(v)$ of minimal degree d in $B \cup Spairs$.

    (b) $\mathrm{bin}(w) := \mathrm{bin}(v)^{G(\mathrm{sat})}$;

    (c) if $(\mathrm{bin}(w) = 0)$ continue;

    (d) if$(\deg(\mathrm{bin}(w)) < d)$

        (i) OUTPUT $\mathrm{bin}(w)$ as proof that I is not a lattice ideal and HALT.

    (e) $G := G \cup \{\mathrm{bin}(w)\}$;

    (f) **updateSpairs**

5   The Sullivant challenge

Sullivant's challenge[4] is about deciding if the ideal J generated by a given set B of $145,512$ binomials generate the kernel P of the toric ring homomorphism

$$k[x_{ijk}] \rightarrow k[u_{ij}, v_{ik}, w_{jk}]$$

given by $x_{ijk} \mapsto u_{ij}v_{ik}w_{jk}$, where $1 \leq i, j, k \leq 4$. The $145,512$ binomials are constructed by acting with a symmetry group on carefully selected binomials[5]. In this setting we need to compute in the polynomial ring $k[x_{ijk}]$ in 64 variables! The ideal J is homogeneous in the natural grading $\deg(x_{111}) = \cdots = \deg(x_{444}) = 1$. The strategy is applying algorithm (4.2) to J using a reverse lexicographic order. If algorithm (4.2) finishes without halting in step (iid), then Sullivant has proved that J must generate P. If not, algorithm (4.2) will halt with a binomial in $P \setminus J$.

Running the gbasis command of GLATWALK with respect to the cost vector $-e_1$ and the grading $e_1 + \cdots + e_{64}$ we compute a Gröbner basis of J after converting the binomials in the two files[6][7] containing J into integer vector format. After computing a 15-truncated Gröbner basis, gbasis (in the incarnation of algorithm (4.2)) outputs the degree 14 binomial

$$x_{311}x_{221}x_{431}x_{212}^2x_{122}x_{342}x_{113}x_{433}x_{243}x_{424}x_{134}x_{334}x_{444}$$
$$- x_{211}x_{421}x_{331}x_{112}x_{312}x_{222}x_{242}x_{213}x_{133}x_{443}x_{124}x_{434}^2x_{344}$$

as a binomial in $\bar{J} \setminus J$ proving that J does not generate P thereby answering Sullivant's computational commutative algebra challenge. Running gbasis in the above setting is not a simple computation. In fact the 15-truncated Gröbner basis of J contains more than $300,000$ binomials and the whole computation takes close to two days on most modern PCs.

Details and more information, including the relevant files for Sullivant's challenge, are located at http://home.imf.au.dk/niels/GLATWALK.

References

[1] K. Caboara, M. Kreuzer, L. Robbiano. Effeciently computing minimal sets of critical pairs. *J. Symbolic Computation* **38** (2004), 1169–1190.

[2] Cox, Little and O'Shea. Ideals, Varieties and Algorithms. Undergraduate Texts in Mathematics. Springer Verlag, 1992.

[3] B. Sturmfels. Gröbner Bases and Convex Polytopes, University Lecture Series **8**, Amer. Math. Soc., Providence, RI, 1996.

---

[4] http://math.berkeley.edu/~seths/ccachallenge.html

[5] See http://math.berkeley.edu/~seths/ccachallenge.ps for details

[6] http://math.berkeley.edu/~seths/polyout.mac.gz

[7] http://math.berkeley.edu/~seths/polyout2.mac.gz