Journal of Symbolic Computation 36 (2003) 595–611

Journal of Symbolic Computation

www.elsevier.com/locate/jsc

Computing in algebraic geometry and commutative algebra using *Macaulay 2*

Michael Stillman

Department of Mathematics, Malott Hall, Cornell University, Ithaca, NY 14853-4201, USA

Received 9 December 2002; accepted 25 April 2003

Abstract

We present recent research of Eisenbud, Fløystad, Schreyer, and others, which was discovered with the help of experimentation with *Macaulay 2*. In this invited, expository paper, we start by considering the exterior algebra, and the computation of Gröbner bases. We then present, in an elementary manner, the explicit form of the Bernstein–Gelfand–Gelfand relationship between graded modules over the polynomial ring and complexes over the exterior algebra, that Eisenbud, Fløystad and Schreyer found. We present two applications of these techniques: cohomology of sheaves, and the construction of determinantal formulae for (powers of) Macaulay resultants. We show how to use *Macaulay 2* to perform these computations.

1. Introduction

This invited talk at ISSAC 2002 has three goals. We wish to present some exciting new research of Eisenbud, Fløystad, Schreyer, and others, which was discovered with the help of experimentation with *Macaulay 2*. In the process, we hope to convince the reader that it is possible to compute with relatively abstract notions in algebraic geometry, and finally, we show how these computations can be performed using *Macaulay 2*.

Macaulay 2 is computer software for algebraic geometry, commutative algebra and related fields. Grayson and I have been working on *Macaulay 2* since we started the project in 1993. *Macaulay 2* is freely available (Grayson and Stillman, 1993–2003).

This paper is an introduction to the work of Eisenbud, Fløystad, and Schreyer. More details and proofs can be found in the papers (Eisenbud et al., 2001; Eisenbud and Schreyer, 2001; Decker and Eisenbud, 2001) and in the book (Eisenbud, 2003). In particular, they prove considerably more than we explain here. What we do instead is explain their constructions and apply them to examples. We present two applications of these techniques: cohomology of sheaves, and the construction of determinantal formulae for

E-mail address: mike@math.cornell.edu (M. Stillman).

0014-5793/03/$ - see front matter
doi:10.1016/S0747-7171(03)00096-8

(powers of) Macaulay resultants. We culminate with finding an explicit skew symmetric 8×8 matrix whose pfaffian (square root of the determinant) is the Macaulay resultant of three ternary quadratic forms. This is one of the new determinantal-like formulae that Eisenbud and Schreyer found in Eisenbud and Schreyer (2001).

2. The exterior algebra

Fix a field k. The exterior algebra E on n letters $e_1, \ldots, e_n$ is defined to be the free associative k-algebra on $e_1, \ldots, e_n$, modulo the relations

$$e_1^2 = \cdots = e_n^2 = 0,$$

and

$$e_i e_j = -e_j e_i,$$

for all i and j. There is no truly standard notation for this algebra. We will denote this exterior algebra by $E = k\langle e_1, \ldots, e_n\rangle$. A k-basis for E consists of all square-free monomials, so has dimension 2^n as a vector space over k.

Since multiplication is almost commutative, Gröbner bases and Buchberger's algorithm both extend, with simple modifications, to ideals in E, as well as to modules over E. (We always consider right-ideals and right modules.) The only difference is that the notion of "S-pair" must be modified. An example should make it clear.

Example 1. Let $E = \mathbb{Q}\langle a, b, c, d\rangle$ be the exterior algebra on letters $a, \ldots, d$. Let I be the ideal generated by $F = ac - bd$ and $G = bc - ad$.

Over the usual polynomial ring, the Buchberger algorithm works by selecting pairs of polynomials, cancelling their lead terms by taking a linear combination of the two polynomials, and computing its remainder.

We start to compute a Gröbner basis by considering the pair (F, G). The combination that cancels lead terms is

$$bF + aG = b(ac - bd) + a(bc - ad) = 0.$$

The one difference, other than arithmetic, is that we may uncover new lead terms by multiplying a polynomial such as F by any variable which occurs in its lead term. Thus,

$$aF = a(ac - bd) = -abd$$

and

$$cF = c(ac - bd) = bcd.$$

$H = abd$ is not divisible by the lead terms of F or G, and so we add it to the Gröbner basis. The element bcd does reduce to zero, by subtracting dG.

All other "S-pairs" on each of F, G and H reduce to zero, and so $\{F, G, H\}$ is a Gröbner basis of I.

By using this straightforward extension of Buchberger's algorithm, we can compute Gröbner bases. As with polynomial rings, if we extend the algorithm to modules, and keep track of how Gröbner basis elements are expressed in terms of the original generators, we can also compute *syzygies*, i.e. kernels of $r \times s$ matrices

$$\phi : E^s \longrightarrow E^r.$$

For example, let's compute with the above ideal I using *Macaulay 2*.

```
i1 : E = QQ[a..d, SkewCommutative=>true];
```

Multiplication is as defined above.

```
i2 : c*b

o2 = -b*c

o2 : E


i3 : b^2

o3 = 0

o3 : E


i4 : I = ideal(a*c-b*d, b*c-a*d)

o4 = ideal (a*c - b*d, b*c - a*d)

o4 : Ideal of E


i5 : transpose gens gb I

o5 = {-2} | bc-ad |
     {-2} | ac-bd |
     {-3} | abd   |

             3       1
o5 : Matrix E  <--- E


i6 : m = generators I

o6 = | ac-bd bc-ad |

             1       2
o6 : Matrix E  <--- E


i7 : m1 = syz m

o7 = {2}  | c  -d a b 0  |
     {2}  | -d c  b a bd |

             2       5
o7 : Matrix E  <--- E
```

We can compute the kernel of this map.

```
i8 : syz m1

o8 = {3}  | c  -d 0 0 a  b  0 0 0  0  |
     {3}  | -d c  0 0 b  a  0 0 0  bd |
     {3}  | 0  0  a b c  -d 0 0 0  0  |
     {3}  | 0  0  b a -d c  0 0 bd 0  |
     {4}  | 0  0  0 0 0  0  d b -a -c |

             5       10
o8 : Matrix E  <--- E
```

The `resolution` routine iterates this process as far as we want, to compute a (minimal) free resolution for some number of steps.

```
i9 : resolution(cokernel m, LengthLimit=>7)

      1      2      5      10      18      30      47      70
o9 = E  <-- E  <-- E  <-- E   <-- E   <-- E   <-- E   <-- E

     0      1      2      3       4       5       6       7

o9 : ChainComplex
```

There are two major differences between computing over the usual polynomial ring (the symmetric algebra) and the exterior algebra: first, Gröbner bases and syzygies are *much* easier to compute over the exterior algebra. This is in large part due to the small number of monomials in the exterior algebra. Second, Hilbert proved in 1890 that minimal free resolutions over the polynomial ring are always finite. Over the exterior algebra, free resolutions are almost never finite. However, because of the small number of monomials in E, finite parts of these resolutions can often be found quickly.

3. The link between the exterior algebra and the symmetric algebra

Let $S = k[x_1, \dots x_n]$, and let $E = k\langle e_1, \dots, e_n\rangle$ be the corresponding exterior algebra. Let

$$M = \bigoplus_{d\in\mathbb{Z}} M_d,$$

be a graded S-module, where each degree d piece M_d is a finite dimensional vector space over k. Once and for all, choose a basis of each vector space M_d. If $m \in M_d$, denote by $[m]$ its representation in this basis. Throughout this section, we set $m_d := \dim M_d$, for all d.

The whole theory rests on the following particular method for encoding the data of multiplication by linear forms from M_d to M_{d+1}.

Definition 2 (Bernstein–Gelfand–Gelfand Maps). For a given d, define the d **th BGG map**

$$\phi_d : E^{m_d} \longrightarrow E^{m_{d+1}}$$

of M by

$$[m] \mapsto \sum_{i=1}^{n} e_i [x_i m].$$

We could do everything without coordinates, and then this would be the adjoint map to the multiplication map. Given a vector space V of dimension n, define $W = V^*$ to be the dual vector space, and then set $S = S_* V$ and $E = \Lambda^* W$ to be the symmetric and exterior algebras respectively. If V is the span of $x_1, \ldots, x_n$, then this is exactly what we have considered already. The multiplication map is $V \otimes_k M_d \longrightarrow M_{d+1}$. The adjoint is $M_d \longrightarrow V^* \otimes_k M_{d+1}$. Considering the elements of $V^* = W$ as linear elements of E, the adjoint gives a matrix of linear forms ϕ_d defined by the above formula.

Example 3. As a simple example, consider $S = k[x_1, x_2]$ and the corresponding exterior algebra $E = k\langle e_1, e_2\rangle$. Let

$$M = S/(x_1^2, x_2^3) = M_0 \oplus M_1 \oplus M_2 \oplus M_3,$$

where $M_0 = k$, $M_1 = k^2$, $M_2 = k^2$, and $M_3 = k$. We choose monomials as the basis elements of these vector spaces. The basis of M_0 is $\{[1]\}$, the basis of M_1 is $\{[x_1], [x_2]\}$, the basis of M_2 is $\{[x_1 x_2], [x_2^2]\}$ and the basis of M_3 is $\{[x_1 x_2^2]\}$.

Setting $d = 1$ for example, $\phi_1 : E^2 \longrightarrow E^2$ is defined by

$$[x_1] \mapsto e_1[x_1^2] + e_2[x_1 x_2],$$

and

$$[x_2] \mapsto e_1[x_1 x_2] + e_2[x_2^2].$$

Since $[x_1^2] = 0$, $\phi_1 = \begin{pmatrix} e_2 & e_1 \\ 0 & e_2 \end{pmatrix}$.

Computing ϕ_0 and ϕ_2, we obtain a sequence

$$0 \longrightarrow k \xrightarrow{\begin{pmatrix} e_1 \\ e_2 \end{pmatrix}} k^2 \xrightarrow{\begin{pmatrix} e_2 & e_1 \\ 0 & e_2 \end{pmatrix}} k^2 \xrightarrow{\begin{pmatrix} e_2 & e_1 \end{pmatrix}} k \longrightarrow 0.$$

Note that this is a complex: applying two maps in a row gives zero.

In the general case, if we apply the construction for each d to a graded S-module M, we get a (possibly infinite) sequence of maps:

$$\cdots \longrightarrow E^{m_0} \xrightarrow{\phi_0} E^{m_1} \xrightarrow{\phi_1} E^{m_2} \xrightarrow{\phi_2} \cdots.$$

There are two basic facts about this sequence.

- This is a *complex*, i.e. $\phi_{i+1}\phi_i = 0$ for all i. This is a simple exercise, using the fact that multiplication in S is commutative.
- This complex is eventually exact, i.e. for $i \gg 0$, $\ker(\phi_{i+1}) = \operatorname{im}(d_i)$. This theorem is proved in Eisenbud et al. (2001). They also show that if the Castelnuovo–Mumford regularity of M is r, then this sequence is exact after the rth step.

The cohomology of this sequence would be interesting to investigate, but for now, we take a tail of this complex which is exact:

$$\cdots \longrightarrow E^{m_r} \xrightarrow{\phi_r} E^{m_{r+1}} \xrightarrow{\phi_{r+1}} E^{m_{r+2}} \xrightarrow{\phi_{r+2}} \cdots.$$

The crucial link is the following exact sequence.

Definition 4. The **Tate resolution** $\mathbf{T}(M)$ of M is the exact complex (possibly infinite in both directions)

$$\cdots \longrightarrow E^{m_{r-2}} \xrightarrow{\psi_{r-2}} E^{m_{r-1}} \xrightarrow{\psi_{r-1}} E^{m_r} \xrightarrow{\psi_r} E^{m_{r+1}} \xrightarrow{\psi_{r+1}} \cdots$$

obtained by computing a *free resolution* of $\ker(\phi_r)$, where r is chosen large enough so that the tail of the complex is exact, and $\psi_i = \phi_i$, for $i \geq r$.

In the example above, the Tate resolution is the zero complex, since the complex is eventually zero.

Eisenbud, Fløystad, and Schreyer prove that the Tate resolution is independent (in a suitable sense) of the truncation location r, as long as it is chosen so that the tail is exact. This implies that if $M_{\geq e} := \bigoplus_{d \geq e} M_d$ is the eth truncation of M, then $\mathbf{T}(M) = \mathbf{T}(M_{\geq e})$.

This is reminiscent of a property of sheaves on projective space. So, before continuing, let us brush up on sheaves.

3.1. An aside: a crash course on implementing coherent sheaves on $\mathbb{P}^{n-1}$

Serre's famous FAC paper (Serre, 1955) introduced sheaves to algebraic geometry. What is perhaps less well known is that he describes (in essence) how to represent sheaves on projective space as modules, and how to compute their cohomology. In this sense, his paper is perhaps the first paper in computational abstract algebraic geometry.

A graded S-module M determines in a canonical manner a coherent sheaf $\widetilde{M}$ on $\mathbb{P}^{n-1}$, and all coherent sheaves on $\mathbb{P}^{n-1}$ arise in this manner. Unfortunately, the correspondence is not one to one. For example, if $M_{\geq e}$ is the eth truncation of the module M, then $\widetilde{M} = \widetilde{M_{\geq e}}$. In fact, two coherent sheaves $\widetilde{M}$ and $\widetilde{N}$ are isomorphic if and only if there is an integer e such that $M_{\geq e} \simeq N_{\geq e}$.

If $X \subset \mathbb{P}^{n-1}$ is a projective variety defined by an ideal $I_X \subset S$, then $\mathcal{O}_X := \widetilde{S/I_X}$ is called the sheaf of regular functions on X.

Another important construction is the twist of a sheaf. If e is an integer, then $\widetilde{M}(e) := \widetilde{M(e)}$, where $M(e)$ is the same module as M, but with a shift in the grading: $M(e)_d := M_{e+d}$.

Cohomology of sheaves can be computed using the representation of the sheaf as a graded S-module. We will not describe these algorithms here (see Eisenbud's chapter in Vasconcelos, 1998). Instead, we use *Macaulay 2* to compute some of the cohomology groups of the sheaves $\mathcal{O}_C(d)$, where $C \subset \mathbb{P}^3$ is the twisted cubic curve. Note that $h^i(\widetilde{M}) := \dim_k H^i(\widetilde{M})$ is the notation often used for the dimensions of the cohomology groups (and are also k vector spaces).

Example 5. The twisted cubic curve is the image of the map

$$\mathbb{P}^1 \longrightarrow \mathbb{P}^3$$

which sends

$$(s, t) \mapsto (W, X, Y, Z) = (s^3, s^2t, st^2, t^3).$$

The ideal I_C of the image is generated by $\{X^2 - WY, Y^2 - XZ, WZ - XY\}$.

```
i10 : S = QQ[W,X,Y,Z];

i11 : IC = ideal(X^2-W*Y, Y^2-X*Z, W*Z-X*Y);

o11 : Ideal of S

i12 : C = variety IC

o12 = C

o12 : ProjectiveVariety

i13 : (HH^0(OO_C), HH^0(OO_C(1)), HH^1(OO_C(-5)))

         1    4    14
o13 = (QQ , QQ , QQ  )

o13 : Sequence
```

So $h^0(\mathcal{O}_C) = 1, h^0(\mathcal{O}_C(1)) = 4$, and $h^1(\mathcal{O}_C(-5)) = 14$. Here are more cohomology groups:

```
i14 : apply(-3..4, i -> HH^0(OO_C(i)))

                      1    4    7    10    13
o14 = (0, 0, 0, QQ , QQ , QQ , QQ  , QQ  )

o14 : Sequence

i15 : apply(-6..4, i -> HH^1(OO_C(i)))

         17    14    11    8    5    2
o15 = (QQ  , QQ  , QQ  , QQ , QQ , QQ , 0, 0, 0, 0, 0)

o15 : Sequence
```

3.2. *Tate resolutions and cohomology of coherent sheaves*

What have we done so far? We start with a graded S-module M, or its associated sheaf $\widetilde{M}$ and obtain an exact complex $\mathbf{T}(M)$ of free E-modules, which is eventually linear. This is cute, and pretty, but so what? What good is it? Well, it turns out to be amazingly useful. We will see two completely different applications below. In addition, there are several other applications that we do not have the time or space to describe (see Eisenbud et al., 2001; Eisenbud and Schreyer, 2001; Decker and Eisenbud, 2001).

Let us compute the Tate resolution of the twisted cubic curve in projective 3-space. We'll use *Macaulay 2* to do the computations for us.

```
i16 : load "bgg.m2"; -- described in the appendix


i17 : E = QQ[w,x,y,z, SkewCommutative => true];


i18 : M = cokernel generators IC;
```

M is the homogeneous coordinate ring of the twisted cubic curve. The degree one and two parts have bases consisting of the following sets of monomials.

```
i19 : basis(1,M)

o19 = | W X Y Z |

o19 : Matrix


i20 : basis(2,M)

o20 = | W2 WX WY WZ XZ YZ Z2 |

o20 : Matrix
```

The routine bgg computes the map ϕ_d defined above.

```
i21 : phi1 = bgg(1,M,E)

o21 = {-2} | w 0 0 0 |
      {-2} | x w 0 0 |
      {-2} | y x w 0 |
      {-2} | z y x w |
      {-2} | 0 z y x |
      {-2} | 0 0 z y |
      {-2} | 0 0 0 z |

               7       4
o21 : Matrix E  <--- E


i22 : phi2 = bgg(2,M,E)

o22 = {-3} | w 0 0 0 0 0 0 |
      {-3} | x w 0 0 0 0 0 |
      {-3} | y x w 0 0 0 0 |
      {-3} | z y x w 0 0 0 |
      {-3} | 0 z y x w 0 0 |
      {-3} | 0 0 z y x w 0 |
      {-3} | 0 0 0 z y x w |
      {-3} | 0 0 0 0 z y x |
      {-3} | 0 0 0 0 0 z y |
      {-3} | 0 0 0 0 0 0 z |

               10       7
o22 : Matrix E   <--- E


i23 : phi2 * phi1
```

```
o23 = 0

               10       4
o23 : Matrix E   <--- E
```

Here is a *Macaulay 2* routine for computing (a part of) the Tate resolution. The routine first truncates M at the regularity of M, calls `bgg`, and then computes several steps of a free resolution. Notice that the second matrix from the left is $\phi_2 : E^7 \longrightarrow E^{10}$ (up to change of basis).

```
i24 : Ta = tateResolution(presentation M,E,-3,4)

        13       10       7       4       3       5       8       11
o24 = E   <-- E   <-- E  <-- E  <-- E  <-- E  <-- E  <-- E

      -4      -3      -2      -1      0       1       2       3

o24 : ChainComplex
```

Caution! Macaulay 2 displays maps from right to left, so the eventually linear part here is the leftmost displayed part of the Tate resolution. For example, $\psi_{-1} : E^5 \longrightarrow E^3$ is

```
i25 : Ta.dd_1

o25 = {0} | 0 0 -wz wy wx |
      {1} | z y x   -w 0  |
      {1} | 0 z y   -x -w |

               3       5
o25 : Matrix E  <--- E
```

The graded pieces of each module are displayed using the `betti` command.

```
i26 : betti Ta

o26 = total: 13 10 7 4 3 5 8 11
          0: 13 10 7 4 1 . . .
          1:  .  . . . 2 5 8 11
```

The entry in row d: and column c (where the first column displayed is $c = -4$) is the number of generators of degree $d + c$ in the cth free module, where each variable in the exterior algebra has degree 1. For example, ψ_{-1}, as a map of graded free modules, has the form $\psi_{-1} : E(-2)^5 \longrightarrow E(-1)^2 \oplus E$, where $E(-d)$ is the graded free module of rank one, having its generator in degree d.

Wait! These numbers in the betti diagram are the same numbers we encountered when computing the cohomology of $\mathcal{O}_C$ and its twists. Eisenbud, Fløystad, and Schreyer observed this, and then were able to prove in general that the graded pieces of the Tate resolution $\mathbf{T}(M)$ are exactly the cohomology modules of $\widetilde{M}$ and its twists. In terms of this `betti` diagram, the statement is:

Theorem 6 (Eisenbud et al., 2001). *Let M be a graded $S = k[x_1, \dots, x_n]$ module. Let $\widetilde{M}$ be the corresponding sheaf. Then: the betti diagram of the Tate resolution of M has the form*

$$\begin{array}{ccccc}
\cdots & h^0(\widetilde{M}(1)) & h^0(\widetilde{M}) & h^0(\widetilde{M}(-1)) & \cdots \\
\cdots & h^1(\widetilde{M}) & h^1(\widetilde{M}(-1)) & h^1(\widetilde{M}(-2)) & \cdots \\
 & \vdots & & \vdots & \\
\cdots & h^{n-1}(\widetilde{M}(-n+2)) & h^{n-1}(\widetilde{M}(-n+1)) & h^{n-1}(\widetilde{M}(-n)) & \cdots
\end{array}$$

4. Resultants, Chow forms, and the Tate resolution

There has been a great deal of interest in finding determinantal formulae for multivariate resultants (the *Macaulay* resultant), and for sparse resultants. For one such result from these proceedings, with pointers to the literature for others, see Khetan (2002).

Khetan (2002) discovered that their exterior algebra methods can be used to construct determinantal formulae for some of these resultants. In this section, we present a part of this work, leading up to an explicit Bezout formula for the Macaulay resultant of three quadratic forms in three variables.

Macaulay resultants are the Chow forms of Veronese varieties, and Eisenbud and Schreyer find formulae for Chow forms. Thus, our story starts with Chow forms.

4.1. *The Chow divisor and Chow form of V*

Let $X \subset \mathbb{P}^{n-1}$ be a projective variety of dimension d. Let G be the set of all codimension $d+1$ planes L in $\mathbb{P}^{n-1}$. This is a Grassmann variety, and has dimension $(d+1)(n-d-1)$.

The **Chow divisor** D_X of X is

$$D_X = \{L \in G \mid X \cap L \neq \emptyset\}.$$

It is an exercise in dimension theory to show that D_X has codimension one in G.

An element L of G is represented by a $(d+1) \times n$ matrix M such that if $H_i = M_{i1}x_1 + \cdots + M_{in}x_n$, then $L = (H_1 = 0) \cap \cdots \cap (H_{d+1} = 0)$. The Chow divisor D_X is defined by a single equation Ch_X (the **Chow form**) in the indeterminates M_{ij}. This polynomial may also be expressed as a polynomial in the Plücker coordinates

$$[i_1, \dots, i_{d+1}] = \det \begin{pmatrix} M_{1,i_1} & M_{1,i_2} & \dots & M_{1,i_{d+1}} \\ \vdots & & \dots & \vdots \\ M_{d+1,i_1} & M_{d+1,i_2} & \dots & M_{d+1,i_{d+1}} \end{pmatrix}.$$

The degree of the polynomial in the Plücker coordinates is the degree of X.

Let us now specialize to the Veronese surface $V \subset \mathbb{P}^5$. This surface is the image of the map

$$\mathbb{P}^2 \longrightarrow \mathbb{P}^5$$

given by

$$(r, s, t) \mapsto (A, B, C, D, E, F) = (r^2, rs, rt, s^2, st, t^2).$$

The variety V has dimension 2 and degree 4.

What is the Chow form Ch_V of the Veronese surface? In this situation, G is the Grassmannian of codimension 3 subspaces of $\mathbb{P}^5$. Each element $L \in G$ is determined by a 3×6 matrix

$$M = \begin{pmatrix} a_1 & \dots & a_6 \\ b_1 & \dots & b_6 \\ c_1 & \dots & c_6 \end{pmatrix}$$

where if

$$\begin{aligned} H_a &= a_1x_1 + \cdots + a_6x_6, \\ H_b &= b_1x_1 + \cdots + b_6x_6, \\ H_c &= c_1x_1 + \cdots + c_6x_6, \end{aligned}$$

then $L = (H_a = 0) \cap (H_b = 0) \cap (H_c = 0)$. The Plücker coordinates are

$$[i, j, k] := \det \begin{pmatrix} a_i & a_j & a_k \\ b_i & b_j & b_k \\ c_i & c_j & c_k \end{pmatrix}.$$

Ch_V is a polynomial in the 18 variables a, b, c. Ch_V can also be expressed in terms of the Plücker coordinates. Since the degree of the Veronese surface is four, general theory tells us that Ch_V is a polynomial of degree four in the Plücker coordinates (and so of degree 12 in the a, b, c variables).

Let

$$\begin{aligned} F_a &= a_1r^2 + a_2rs + a_3rt + a_4s^2 + a_5st + a_6t^2 \\ F_b &= b_1r^2 + b_2rs + b_3rt + b_4s^2 + b_5st + b_6t^2 \\ F_c &= c_1r^2 + c_2rs + c_3rt + c_4s^2 + c_5st + c_6t^2. \end{aligned}$$

We now identify the Chow form of V.

$$\begin{aligned} & Ch_V(a, b, c) = 0 \\ & \quad \Longleftrightarrow V \cap (H_a = 0) \cap (H_b = 0) \cap (H_c = 0) \neq \emptyset \\ & \quad \Longleftrightarrow F_a(r, s, t) = F_b(r, s, t) = F_c(r, s, t) = 0 \qquad \text{for some } (r, s, t) \in \mathbb{P}^2 \\ & \quad \Longleftrightarrow Res_{2,2,2}(F_a, F_b, F_c) = 0. \end{aligned}$$

Therefore (since both are irreducible polynomials)

$$Ch_V(a, b, c) = Res_{2,2,2}(F_a, F_b, F_c)$$

is the Macaulay resultant of three ternary quadratic forms.

The goal is to find determinantal formulae for resultants such as this. Here is one example, which appears in the list in Gelfand et al. (1994). The Chow form Ch_V is the determinant of the 6×6 matrix

$$\begin{pmatrix} a_1 & b_1 & c_1 & [1,2,6] & 0 & [1,2,3] \\ a_2 & b_2 & c_2 & [1,4,6] & [1,4,5] & [1,2,5]-[1,3,4] \\ a_3 & b_3 & c_3 & [1,5,6]-[2,3,6] & [1,4,6] & [1,2,6] \\ a_4 & b_4 & c_4 & 0 & [2,4,5] & [1,4,5] \\ a_5 & b_5 & c_5 & [3,4,6] & [3,4,5]+[2,4,6] & [1,4,6] \\ a_6 & b_6 & c_6 & [3,5,6] & [3,4,6] & 0 \end{pmatrix}.$$

If we use a Laplace expansion with the first three columns, we see that the determinant is a polynomial of degree 4 in the cubics $[i, j, k]$.

Question: Can the polynomial Ch_V be expressed as the determinant of a 4×4 matrix whose entries are linear in the Plücker coordinates $[i, j, k]$? Or, if not, is there *any* nice formula involving only the $[i, j, k]$'s? In fact, there is no such 4×4 determinant, but Eisenbud and Schreyer construct an 8×8 skew symmetric matrix whose pfaffian is $Ch_V = Res_{2,2,2}$. In the rest of this section, we describe their construction, and at the end we obtain the 8×8 matrix explicitly.

4.2. *The Eisenbud–Schreyer construction*

For the general construction, see Eisenbud and Schreyer (2001). Here we present an important special case, which works for the Veronese and many other cases.

Start with a variety $X \subset \mathbb{P}^{n-1}$. Let M be a graded $S = k[x_1, \ldots, x_n]$ module, which is supported on X (i.e. $I_X \subset ann(M)$). Assume that the sheaf associated to M is locally free on X of rank r. There is an additional assumption on M (M is "Ulrich") for the formula below to work as nicely as it does, but we will not get into that here. See Eisenbud and Schreyer (2001) for the specific condition.

From this module M, find the Tate resolution of M. In the case when M is suitably nice, i.e. is "Ulrich", the resolution has the form

$$\cdots \longrightarrow E^{\gamma} \longrightarrow E^{\alpha} \xrightarrow{\psi} E^{\alpha} \longrightarrow E^{\beta} \longrightarrow \cdots.$$

The entries of all of the matrices except ψ are linear in $e_1, \ldots, e_n$, and the non-zero entries of the matrix ψ all have degree $d+1$. The final step of the construction is to create the $\alpha \times \alpha$ matrix $\mathbf{U}(\psi)$ whose entries are obtained from those of ψ by setting $T(e_i e_j e_k) = [i, j, k]$, and extending via k-linearity. For example, $T(e_1e_2e_3 + e_1e_2e_4) = [1, 2, 3] + [1, 2, 4]$.

Theorem 7. *If $\widetilde{M}$ is locally free on X of rank r, and M is "Ulrich", then*

$$\det \mathbf{U}(\psi) = (Ch_X)^r.$$

In particular, if $\widetilde{M}$ is a line bundle ($r = 1$), then the construction provides a determinantal formula for Ch_X.

To summarize,

Construction 8 (A Power of the Chow Form).
input: An S-module M, such that the sheaf $\widetilde{M}$ is locally free of rank r on a variety $X \subset \mathbb{P}^{n-1}$, and M is "Ulrich".
output: A square matrix $\mathbf{U}(\psi)$ whose entries are linear forms in the Plücker coordinates, such that $(Ch_X)^r = \det \mathbf{U}(\psi)$.
begin
  Compute the matrix $\psi = \psi_{-1}$ in the Tate resolution.
  if the matrix ψ is not square or has entries not of degree $d+1$
    then error M is not Ulrich.
  Return $\mathbf{U}(\psi)$.
end.

In practice, such an Ulrich module M may not exist, or at least might be difficult to find. Eisenbud and Schreyer (2001) give criteria for when such a module exists, and how to find one. In particular, with their construction they can reproduce all of the known determinantal formulae (at least the ones that appear in Gelfand et al., 1994 for Macaulay resultants).

4.3. The resultant $Res_{2,2,2}(F_a, F_b, F_c)$

In our example of the Veronese surface $V \subset \mathbb{P}^5$, we choose $M = TV$ to be a graded S-module which corresponds to the (rank two) tangent bundle of V. In this case, the module M is "Ulrich". When we apply the above construction, we obtain an 8×8 *skew symmetric* matrix whose determinant is the square of $Ch_V = Res_{2,2,2}(F_a, F_b, F_c)$. Recall that the determinant of a skew symmetric matrix (even a matrix of polynomials) is a square. Its square root is called the *pfaffian* of the matrix. Therefore the resultant $Res_{2,2,2}$ is the pfaffian of an 8×8 matrix of linear forms in the Plücker coordinates $[i, j, k]$. We will construct this matrix using *Macaulay 2*.

```
i27 : S3= QQ[r,s,t];
```

We use the variables $A, \ldots, F$, and $a, \ldots, f$ instead of x_i and e_i, to improve the readability of the *Macaulay 2* output.

```
i28 : S6 = QQ[A..F,Degrees=>{2,2,2,2,2,2}];


i29 : E6 = QQ[symbol a..symbol f,SkewCommutative=>true]

o29 = E6

o29 : PolynomialRing


i30 : FV = map(S3,S6,{r^2, r*s, r*t, s^2, s*t, t^2})

                   2             2        2
o30 = map(S3,S6,{r , r*s, r*t, s , s*t, t })

o30 : RingMap S3 <--- S6
```

The following three lines of *Macaulay 2* code is one way to compute `TV`, a module in `S6` corresponding to the tangent sheaf of V. This method starts with the tangent bundle of $\mathbb{P}^2$, truncates it so that all of the generators are in degree zero, and pushes it forward to a bundle on the image V.

```
i31 : TP2 = coker transpose vars S3

o31 = cokernel {-1} | r |
               {-1} | s |
               {-1} | t |

                              3
o31 : S3-module, quotient of S3


i32 : M0 = prune truncate(0, TP2)

o32 = cokernel | 0  0  s  0  0  0 0  r  t |
               | 0  0  -t 0  r  t 0  0  0 |
               | 0  0  0  0  -s 0 0  -t 0 |
               | 0  s  0  0  0  0 r  0  0 |
               | 0  -t 0  r  0  s 0  0  t |
               | 0  0  0  -s 0  0 -t 0  0 |
               | s  0  0  0  0  0 0  0  r |
               | -t 0  0  0  0  r 0  0  0 |

                              8
o32 : S3-module, quotient of S3


i33 : TV = prune coker pushForward1(FV,M0)

o33 = cokernel | -F 0  0  -E 0  0  C 0 0  0  0  0  B 0 0  0  0  0  A 0 0   ...
               | E  0  0  D  0  0  0 C 0  0  0  0  0 B 0  0  0  0  0 A 0   ...
               | 0  0  0  0  0  0  0 0 -E -F 0  0  0 0 -D -E 0  0  D E -B  ...
               | 0  -F 0  0  -E 0  E F C  0  0  0  D E B  0  0  0  0 0 A   ...
               | 0  E  0  0  D  0  0 0 0  C  0  0  0 0 0  B  0  0  0 0 0   ...
               | 0  0  0  0  0  0  0 0 0  0  -E -F 0 0 0  0  -D -E E F 0   ...
               | 0  0  -F 0  0  -E F 0 0  0  C  0  E F 0  0  B  0  0 0 0   ...
               | 0  0  E  0  0  D  0 F 0  0  0  C  0 0 0  0  0  B  0 0 0   ...

                              8
o33 : S6-module, quotient of S6
```

The degrees of the ring `S6` were chosen so that `FV` would be homogeneous. We must adjust the degrees of the ring to be all of degree 1, before computing the Tate resolution.

```
i34 : R6 = QQ[A..F];


i35 : TV = coker substitute(presentation TV, R6);


i36 : Ta = tateResolution(presentation TV,E6,-3,4)

        120        80        48        24        8        8        24        48
o36 = E6    <-- E6    <-- E6    <-- E6    <-- E6   <-- E6   <-- E6    <-- E6

      -4         -3        -2        -1        0        1        2         3

o36 : ChainComplex


i37 : betti Ta
```

```
o37 = total: 120 80 48 24 8 8 24 48
          0: 120 80 48 24 8 .  .  .
          1:   .  .  .  . . .  .  .
          2:   .  .  .  . . 8 24 48


i38 : Ta.dd_1

o38 = | -aef -acf     -adf -ace     -ace+abf 0        -acd     abc     |
      | -bef -bcf     -bdf -bce+adf -bce+adf abf      -bcd+ade -acd+abe |
      | -cef 0        -cdf aef      -bcf+aef acf      adf      abf     |
      | adf  abf      ade  abe      acd      abc      abd      0       |
      | bdf  adf      bde  ade      bcd      -acd     0        -abd    |
      | cdf  -bcf+aef cde  -bce+adf 0        -ace+abf -bcd     -acd    |
      | -def cdf      0    cde+bdf  cde      adf      bde      ade     |
      | 0    cef      def  cdf+bef  cdf      aef      bdf      adf     |

              8        8
o38 : Matrix E6  <--- E6
```

This is the map ψ_{-1}. This matrix is almost the desired 8×8 skew symmetric matrix, except for one problem: it is not skew symmetric! This is because of the choices made by *Macaulay 2* in computing the resolution. By row and column operations over k, it is straightforward to produce the desired skew-symmetric matrix. The following lines of *Macaulay 2* code perform these row and column operations.

```
i39 : load "sparsemat.m2";


i40 : (m = sparseMutableMatrix Ta.dd_1;
          rflip(m,0,7);rflip(m,1,2);rflip(m,6,2);cflip(m,3,6);
          rflip(m,3,4);rscale(m,-1_E6,3);cflip(m,4,5);rflip(m,7,4);
          cflip(m,5,7);rflip(m,7,5);rscale(m,-1_E6,5);caxy(m,-1_E6,7,6);
          rscale(m,-1_E6,7);
          matrix m)

o40 = | 0    cef     def  bdf      aef     adf      bef     cdf      |
      | -cef 0       -cdf adf      acf     abf      bcf     -bcf+aef |
      | -def cdf     0    bde      adf     ade      bdf     cde      |
      | -bdf -adf    -bde 0        acd     abd      bcd-ade -bcd     |
      | -aef -acf    -adf -acd     0       abc      -abf    -ace+abf |
      | -adf -abf    -ade -abd     -abc    0        acd-abe -acd     |
      | -bef -bcf    -bdf -bcd+ade abf     -acd+abe 0       -bce+adf |
      | -cdf bcf-aef -cde bcd      ace-abf acd      bce-adf 0        |

              8        8
o40 : Matrix E6  <--- E6
```

Theorem 9 (Eisenbud and Schreyer, 2001). *The Macaulay resultant*

$$Res_{2,2,2}(F_a, F_b, F_c)$$

is the pfaffian of the 8×8 *matrix* $\mathbf{U}(m)$ *where* m *is the skew symmetric* 8×8 *matrix in the Macaulay 2 code above.*

Acknowledgement

The author was partially supported by NSF grant DMS-9979348.

Appendix A. *Macaulay 2* code used in this paper

The code that we use here appeared in Decker and Eisenbud (2001). They also include *Macaulay 2* code to compute the Beilinson monad, which is another interesting and important application of these exterior algebra techniques. All of these make up the file "`bgg.m2`".

The routine `bgg` computes the matrix ϕ_i corresponding to multiplication from M_i to M_{i+1}. The "BGG" stands for Bernstein–Gelfand–Gelfand.

```
i41 : code bgg

o41 = -- bgg.m2:10-20
      bgg = (i,M,E) ->(
           S :=ring(M);
           numvarsE := rank source vars E;
           ev:=map(E,S,vars E);
           f0:=basis(i,M);
           f1:=basis(i+1,M);
           g :=((vars S)**f0)//f1;
           b:=(ev g)*((transpose vars E)**(ev source f0));
           --correct the degrees (which are otherwise
           --wrong in the transpose)
           map(E^{(rank target b):i+1},E^{(rank source b):i}, b));
```

The routine `symExt` is a subroutine of `tateResolution`. The input is a presentation matrix for the module M above, and it is a method to obtain `bgg(coker m, 0, E)` with less computation.

```
i42 : code symExt

o42 = -- bgg.m2:1-9
      symExt = (m,E) ->(
           ev := map(E,ring m,vars E);
           mt := transpose jacobian m;
           jn := gens kernel mt;
           q  := vars(ring m)**id_(target m);
           ans:= transpose ev(q*jn);
           --now correct the degrees:
           map(E^{(rank target ans):1}, E^{(rank source ans):0},
               ans));
```

The routine `tateResolution` takes as input a presentation matrix for the module M, the corresponding exterior algebra E, and a low and high degree `loDeg` and `hiDeg` which determines the part of the Tate resolution to return. If r is the regularity of the module M, the piece that is returned is

$$\mathbf{T}^{\max(r+2,hi\mathrm{Deg})}(M) \longleftarrow \cdots \longleftarrow \mathbf{T}^{lo\mathrm{Deg}}(M).$$

We have modified the code slightly from Decker and Eisenbud (2001) by shifting the cohomological degrees of the result, so that if `T` is the result, then `T_(-d)` is $T^d(M)$.

```
i43 : code tateResolution

o43 = -- bgg.m2:21-32
      tateResolution = (m,E,loDeg,hiDeg)->(
           M := coker m;
           reg := regularity M;
           bnd := max(reg+1,hiDeg-1);
           mt  := presentation truncate(bnd,M);
           o   := symExt(mt,E);
           --adjust degrees, since symExt forgets them
           ofixed   :=  map(E^{(rank target o):bnd+1},
                      E^{(rank source o):bnd},
                      o);
           C := res(coker ofixed, LengthLimit=>max(1,bnd-loDeg+1));
           C[bnd+1])
```

References

Decker, W., Eisenbud, D., 2001. Sheaf algorithms using the exterior algebra. In: Eisenbud, D., Grayson, D.R., Stillman, M.E., Sturmfels, B. (Eds.), Computations in Algebraic Geometry with *Macaulay 2*. Algorithms and Computations in Mathematics, vol. 8. Springer-Verlag, Berlin, New York, pp. 131–178. ISBN 3-540-42230-7.

Eisenbud, D., 2003. Geometry of Syzygies. Graduate Texts in Mathematics. Springer-Verlag (in preparation).

Eisenbud, D., Fløystad, G., Schreyer, F.-O., 2001. Sheaf Cohomology and Free Resolutions over Exterior Algebras. Available from http://www.arxiv.org/abs/math.AG/0104203/.

Eisenbud, D., Schreyer, F.-O., 2001. Resultants and Chow Forms Via Exterior Syzygies. Available from http://www.arxiv.org/abs/math.AG/0111040/.

Gelfand, I.M., Kapranov, M., Zelevinsky, A., 1994. Discriminants, Resultants, and Multidimensional Determinants. Birkhäuser, Boston.

Grayson, D.R., Stillman, M.E., 1993–2003. Macaulay 2, A Software System for Research in Algebraic Geometry. Available from http://www.math.uiuc.edu/Macaulay2/.

Khetan, A., 2002. The Resultant of an Unmixed Bivariate System. Available from http://www.arxiv.org/abs/math.AG/0209115/.

Serre, J.P., 1955. Faisceaux algébriques cohérents. Ann. Math. 61, 197–278.

Vasconcelos, W., 1998. Computational Methods in Commutative Algebra and Algebraic Geometry. Algorithms and Computations in Mathematics, vol. 2. Springer-Verlag, Berlin, New York.