4 Torics Varieties as Ambient Spaces

How should we work with subvarieties of a toric variety?

Correspondence. Today, the normal toric variety X is *smooth* and *projective.* Set $n := |\Sigma(1)|$. The Cox ring is $S := \mathbb{C}[x_0, x_1, \dots, x_{n-1}]$, where $\deg(x_i) := [D_i] \in \operatorname{Cl}(X)$ for all $0 \leqslant i < n$, and the irrelevant ideal is $B := \langle x^{\hat{\sigma}} \mid \sigma \in \Sigma \rangle = \bigcap_{\gamma} \langle x_{\gamma_0}, x_{\gamma_1}, \dots, x_{\gamma_k} \rangle$.

The *saturation* of an ideal I in S by the irrelevant ideal is

$$(I : B^\infty) := \{ f \in S \mid f B^m \subseteq I \text{ for some positive } m \in \mathbb{Z} \}.$$

Geometrically, this removes the components supported on the irrelevant set.

Proposition. *Every subscheme $Y \subseteq X$ is defined by a homogeneous ideal I in S. Moreover, the homogeneous ideals I and J in S determine the same ideal sheaf on X (and thereby the same closed subscheme in X) if and only if $(I : B^\infty) = (J : B^\infty)$.*

Example. Calabi–Yau hypersurfaces and complete intersections in Fano toric varieties.

Remark. Cox (1995) describes the data needed to specify a map from an arbitrary scheme to a smooth toric variety. It involves a collection of n line bundles (and a global section of each) that satisfy certain compatibility and nondegeneracy conditions.

Hilbert Polynomials. Set $d := \dim X$ and let $r := n - d$ denote the rank of the Picard group of X; $\operatorname{Pic}(X) \cong \mathbb{Z}^r$.

Proposition. *For each* $\operatorname{Pic}(X)$*-graded S-module F, there exists a polynomial $p \in \mathbb{Q}[t_0, t_1, \dots, t_{r-1}]$ such that $p(\mathbf{t}) = \dim_{\mathbb{C}} F_{\mathbf{t}}$ for all $\mathbf{t}$ sufficiently far into the interior of the nef cone. Moreover, we have*

$$p(\mathbf{t}) = \sum_{i=0}^{d} (-1)^i h^i\big(X, \widetilde{F}(\mathbf{t})\big).$$

Theorem (Maclagan–Smith 2005)**.** *For any $p(\mathbf{t}) \in \mathbb{Q}[t_0, t_1, \dots, t_{r-1}]$, there exists a projective scheme* $\operatorname{Hilb}^p(X)$ *that parameterizes all closed subschemes of X with Hilbert polynomial p.*

Unlike projective space, there isn't (yet?) a combinatorial characterization of the polynomials corresponding to nonempty $\operatorname{Hilb}^p(X)$.

Example. The lines on the smooth quadric surface $X := \mathbb{P}^1 \times \mathbb{P}^1$ embedded in $\mathbb{P}^3$ belong to two rulings. The closed subscheme of $\operatorname{Hilb}^{t+1}(\mathbb{P}^3)$ parameterizing subschemes of $\mathbb{P}^3$ with Hilbert polynomial $t+1$ lying on X is $\operatorname{Hilb}^{t_0+1}(X) \sqcup \operatorname{Hilb}^{t_1+1}(X) = \mathbb{P}^1 \sqcup \mathbb{P}^1$.

Virtual Resolutions. On $\mathbb{P}^d$, the saturated ideal $I = (I : B^\infty)$ is most often the "best" homogeneous ideal representing a subvariety. On a general toric variety, this is no longer true.

Example. A hyperelliptic curve C of genus 4 can be embedded as a curve of bidegree $(2, 8)$ in $\mathbb{P}^1 \times \mathbb{P}^2$. Its saturated ideal typically has a free resolution of the form

$$S^1 \leftarrow \begin{matrix} S(-3,-1)^1 \\ \oplus \\ S(-2,-2)^1 \\ \oplus \\ S(-2,-3)^2 \\ \oplus \\ S(-1,-5)^3 \\ \oplus \\ S(0,-8)^1 \end{matrix} \leftarrow \begin{matrix} S(-3,-3)^3 \\ \oplus \\ S(-2,-5)^6 \\ \oplus \\ S(-1,-7)^1 \\ \oplus \\ S(-1,-8)^2 \end{matrix} \leftarrow \begin{matrix} S(-3,-5)^3 \\ \oplus \\ S(-2,-7)^2 \\ \oplus \\ S(-2,-8)^1 \end{matrix} \leftarrow S(-3,-7)^1 \leftarrow 0\,.$$

However, the maximal minors of a (4×3)-matrix cut out the same curve and give

$$S^1 \leftarrow \begin{matrix} S(-3,-1)^1 \\ \oplus \\ S(-2,-2)^1 \\ \oplus \\ S(-2,-3)^2 \end{matrix} \leftarrow S(-3,-3)^3 \leftarrow 0\,.$$

Definition (Berkesch–Erman–Smith 2020). A free $\operatorname{Pic}(X)$-graded S-complex is a *virtual resolution* of a $\operatorname{Pic}(X)$-graded S-module F if the corresponding $\mathscr{O}_X$-complex is a locally free resolution of $\widetilde{F}$.

Theorem (Hanlon–Hicks–Lazarev 2024). *Every coherent sheaf on X has a virtual resolution of length at most* $\dim(X)$.

Idea. Construct a short virtual resolution of the diagonal $X \to X \times X$. The original proof was inspired by homological mirror symmetry. There are already several distict proofs including a combinatorial commutative algebra construction by Berkesch–Heller–Smith–Yang (2026) that relates them. □

Computational Examples in *Macaulay2*.

```
Macaulay2, version 1.26.06
Type "help" to see useful commands
i1 : needsPackage "NormalToricVarieties";
i2 : kk = ZZ/101;
i3 : X0 = toricProjectiveSpace(4, CoefficientRing => kk);
i4 : S0 = ring X0;
i5 : coefficientRing S0 === kk
o5 = true
i6 : I0 = ideal random(5, S0);
o6 : Ideal of S0
i7 : I0 == saturate(I0, ideal X0)
o7 = true
i8 : codim I0
o8 = 1
i9 : p0 = hilbertPolynomial(X0, I0)
     5 3    25
o9 = -i   + --i
     6 0     6 0
o9 : QQ[i ]
         0
```

```
i10 : freeResolution I0

        1       1
o10 = S0  <-- S0
                 
      0       1

o10 : Complex

i11 : all(2..10, i -> hilbertFunction({i},I0) == p0(i))

o11 = true

i12 : X1 = smoothFanoToricVariety(6, 8, CoefficientRing => kk);

i13 : S1 = ring X1;

i14 : isNef(X1_0+X1_1+X1_2+X1_8+X1_9)

o14 = true

i15 : isNef(X1_4)

o15 = true

i16 : isNef(X1_3+X1_5+X1_6+X1_7)

o16 = true

i17 : gens S1

o17 = {x , x , x , x , x , x , x , x , x , x }
        0   1   2   3   4   5   6   7   8   9

o17 : List

i18 : I1 = ideal(random(degree(X1_0+X1_1+X1_2+X1_8+X1_9), S1),
          random(degree(X1_4), S1),
          random(degree(X1_3+X1_5+X1_6+X1_7), S1));

o18 : Ideal of S1

i19 : freeResolution I1

        1       3       3       1
o19 = S1  <-- S1  <-- S1  <-- S1
                                 
      0       1       2       3

o19 : Complex

i20 : p1 = hilbertPolynomial(X1, I1)

      5 3    25
o20 = -i   + --i
      6 3     6 3

o20 : QQ[i ..i ]
          0   3

i21 : p0

      5 3    25
o21 = -i   + --i
      6 0     6 0

o21 : QQ[i ]
          0

i22 : all((1,1,1,1)..(3,3,3,3), i -> hilbertFunction(toList i,I1) == p1 i)

o22 = true

i23 : hilbertFunction({0,-1,1,4}, I1) =!= p1(0,-1,1,4)

o23 = true

i24 : J1 = saturate(I1, ideal X1);

o24 : Ideal of S1

i25 : I1 != J1

o25 = true
```

```
i26 : freeResolution J1

        1       3       3       1
o26 = S1  <-- S1  <-- S1  <-- S1

      0       1       2       3

o26 : Complex
i27 : degree \ J1_*
o27 = {{1, 1, 0, 0}, {0, 0, 1, 4}, {0, 0, 0, 5}}
o27 : List
i28 : degree \ I1_*
o28 = {{0, 0, 1, 4}, {1, 1, 0, 0}, {1, 0, 1, 5}}
o28 : List
i29 : degree (- toricDivisor X1)
o29 = {2, 1, 2, 9}
o29 : List
i30 : X2 = toricProjectiveSpace(1) ** toricProjectiveSpace(2);
i31 : S2 = ring X2;
i32 : I2 = ideal(S2_0^2*S2_2^2+S2_1^2*S2_3^2+S2_0*S2_1*S2_4^2,
          S2_1^3*S2_2-S2_1^3*S2_3+S2_0^3*S2_4,
          S2_1^2*S2_2^3-S2_1^2*S2_2^2*S2_3-
            S2_0*S2_1*S2_3^2*S2_4-S2_0^2*S2_4^3,
          S2_0*S2_1*S2_2^3-S2_0*S2_1*S2_2^2*S2_3-
            S2_0^2*S2_3^2*S2_4+S2_1^2*S2_2*S2_4^2-
              S2_1^2*S2_3*S2_4^2,
          S2_1*S2_2^3*S2_3^2-S2_1*S2_2^2*S2_3^3-
            S2_0*S2_3^4*S2_4-S2_0*S2_2^3*S2_4^2+
              S2_0*S2_2^2*S2_3*S2_4^2-S2_1*S2_2*S2_4^4+
                S2_1*S2_3*S2_4^4,
          S2_1*S2_2^5-S2_1*S2_2^4*S2_3-
            S2_0*S2_2^2*S2_3^2*S2_4+S2_1*S2_3^2*S2_4^3+
              S2_0*S2_4^5, S2_0*S2_2^5-S2_0*S2_2^4*S2_3+
                S2_1*S2_3^4*S2_4+S2_1*S2_2^3*S2_4^2-
                  S2_1*S2_2^2*S2_3*S2_4^2+S2_0*S2_3^2*S2_4^3,
          S2_2^8-2*S2_2^7*S2_3+S2_2^6*S2_3^2+
            S2_3^6*S2_4^2+3*S2_2^3*S2_3^2*S2_4^3-
              3*S2_2^2*S2_3^3*S2_4^3-S2_2*S2_4^7+S2_3*S2_4^7);

o32 : Ideal of S2
i33 : gens S2
o33 = {x , x , x , x , x }
        0   1   2   3   4
o33 : List
i34 : I2 == saturate(I2, ideal X2)
o34 = true
i35 : p2 = hilbertPolynomial(X2, I2)
o35 = 2i  + 8i  - 3
        0     1
o35 : QQ[i ..i ]
          0   1

i36 : CI2 = freeResolution I2

        1       8       12       6       1
o36 = S2  <-- S2  <-- S2   <-- S2  <-- S2

      0       1       2        3       4

o36 : Complex
```

```
i37 : netList apply(1+length CI2, i -> degrees CI2_i)

      +-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
o37 = |{0,0}|     |     |     |     |     |     |     |     |     |     |     |
      +-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
      |{3,1}|{2,2}|{2,3}|{2,3}|{1,5}|{1,5}|{1,5}|{0,8}|     |     |     |     |
      +-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
      |{3,3}|{3,3}|{3,3}|{2,5}|{2,5}|{2,5}|{2,5}|{2,5}|{2,5}|{1,7}|{1,8}|{1,8}|
      +-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
      |{3,5}|{3,5}|{3,5}|{2,7}|{2,7}|{2,8}|     |     |     |     |     |     |
      +-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
      |{3,7}|     |     |     |     |     |     |     |     |     |     |     |
      +-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

i38 : multigraded betti CI2

          0             1          2          3       4
o38 =  0: 1             .          .          .       .
       4: . a^3*b+a^2*b^2          .          .       .
       5: .      2*a^2*b^3         .          .       .
       6: .         3*a*b^5 3*a^3*b^3         .       .
       7: .             . 6*a^2*b^5           .       .
       8: .            b^8     a*b^7 3*a^3*b^5        .
       9: .             .    2*a*b^8 2*a^2*b^7        .
      10: .             .          .   a^2*b^8 a^3*b^7

o38 : MultigradedBettiTally

i39 : J2 = minors(3, matrix{
          {             S2_3^2, S2_4^2,   -S2_2^2},
          {-S2_1*(S2_2-S2_3),       0, S2_0*S2_4},
          {                S2_0,  -S2_1,         0},
          {                   0,   S2_0,      S2_1}});

o39 : Ideal of S2

i40 : CJ2 = freeResolution J2

         1       4       3
o40 = S2  <-- S2  <-- S2

      0       1       2

o40 : Complex

i41 : netList apply(1+length CJ2, i -> degrees CJ2_i)

      +------+------+------+------+
o41 = |{0, 0}|      |      |      |
      +------+------+------+------+
      |{3, 1}|{2, 2}|{2, 3}|{2, 3}|
      +------+------+------+------+
      |{3, 3}|{3, 3}|{3, 3}|      |
      +------+------+------+------+

i42 : multigraded betti CJ2

         0             1         2
o42 = 0: 1             .         .
      4: . a^3*b+a^2*b^2         .
      5: .     2*a^2*b^3         .
      6: .             . 3*a^3*b^3

o42 : MultigradedBettiTally

i43 : I2 == saturate(J2, ideal X2)

o43 = true

i44 : p2 == hilbertPolynomial(X2, J2)

o44 = true
```