# Fibered K3 Surfaces using Sage

Andrey Novoseltsev

Department of Mathematical and Statistical Sciences
University of Alberta

March 11, 2009

# Outline

1. Reflexive Polytopes in Toric Geometry
   - Reflexive Polytopes
   - Homogeneous Coordinates
   - Calabi-Yau Hypersurfaces
   - Tools and Projects

# Outline

1. Reflexive Polytopes in Toric Geometry
   - Reflexive Polytopes
   - Homogeneous Coordinates
   - Calabi-Yau Hypersurfaces
   - Tools and Projects
2. Elliptically Fibered K3 Surfaces
   - Fibrations from Polytopes
   - Slicing the Cube
   - Splitting

# Outline

# Outline

# Lattice and Reflexive Polytopes

Let $N \simeq \mathbb{Z}^d$ be a lattice, $N_\mathbb{R} = N \otimes_\mathbb{Z} \mathbb{R}$, $M = \mathrm{Hom}(N, \mathbb{Z})$.

**Definition**

A *lattice polytope* $\Delta$ is the convex hull in $N_\mathbb{R}$ of finitely many points $v_1, \ldots, v_n \in N$.

# Lattice and Reflexive Polytopes

Let $N \simeq \mathbb{Z}^d$ be a lattice, $N_\mathbb{R} = N \otimes_\mathbb{Z} \mathbb{R}$, $M = \mathrm{Hom}(N, \mathbb{Z})$.

**Definition**

A *lattice polytope* $\Delta$ is the convex hull in $N_\mathbb{R}$ of finitely many points $v_1, \dots, v_n \in N$.

**Definition**

$\Delta$ is *reflexive* if $\Delta^\circ = \{w : \langle w, v \rangle \geqslant -1 \ \forall v \in \Delta\} \subset M_\mathbb{R}$ is a lattice polytope in $M$.

By default we will be dealing with reflexive polytopes.

# Lattice and Reflexive Polytopes

Let $N \simeq \mathbb{Z}^d$ be a lattice, $N_\mathbb{R} = N \otimes_\mathbb{Z} \mathbb{R}$, $M = \mathrm{Hom}(N, \mathbb{Z})$.

## Definition

A *lattice polytope* $\Delta$ is the convex hull in $N_\mathbb{R}$ of finitely many points $v_1, \dots, v_n \in N$.

## Definition

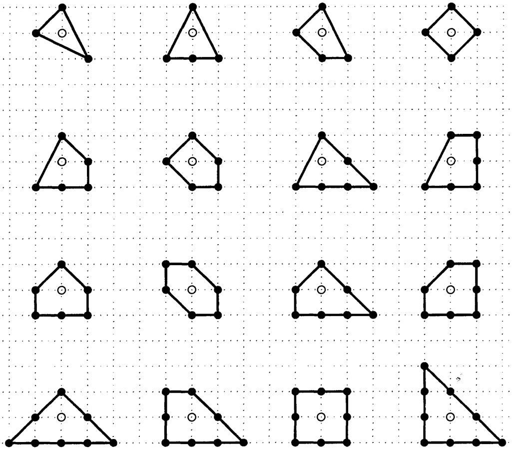
$\Delta$ is *reflexive* if $\Delta^\circ = \{w : \langle w, v \rangle \geqslant -1 \ \forall v \in \Delta\} \subset M_\mathbb{R}$ is a lattice polytope in $M$.

By default we will be dealing with reflexive polytopes.

- A necessary condition for $\Delta$ to be reflexive: the origin is the only *interior* lattice point (sufficient for $d = 2$).
- Vertices of $\Delta^\circ$ are $w_1, \dots, w_m$, where $\langle w_i, v \rangle + 1 = 0$ are equations of facets of $\Delta$ and $w_i$ are normalized inner normals to them.
- $(\Delta^\circ)^\circ = \Delta$.

# Low-Dimensional Reflexive Polytopes

Up to the action of $GL(\mathbb{Z}^d)$, there are finitely many $d$-dimensional reflexive polytopes and there is a classification algorithm.



There are 16 reflexive polygons.

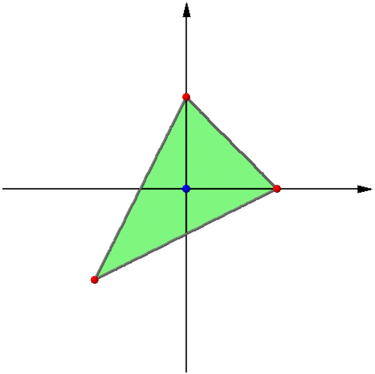
For $d = 3$: 4319.
For $d = 4$: 473,800,776.

The image is taken from Poonen, B. & Rodriguez-Villegas, F., Lattice polygons and the number 12, Amer. Math. Monthly 107 (2000), no. 3, 238–250.

# Outline

# Homogeneous Coordinates: Projective Space

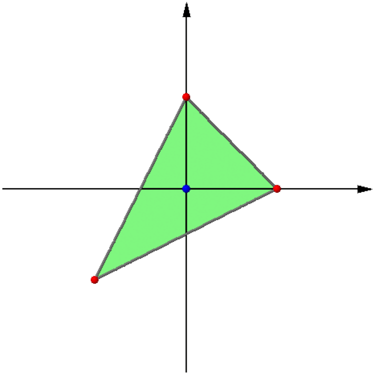
Example:



- Let $v_1 = (1,0)$, $v_2 = (0,1)$, $v_3 = (-1,-1)$ be vertices of $\Delta$.

# Homogeneous Coordinates: Projective Space

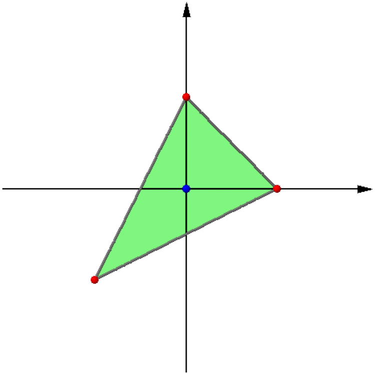
Example:



- Let $v_1 = (1, 0)$, $v_2 = (0, 1)$, $v_3 = (-1, -1)$ be vertices of $\Delta$.
- The only set of vertices which is not contained in a single face of $\Delta$ is $P = \{1, 2, 3\}$.

# Homogeneous Coordinates: Projective Space

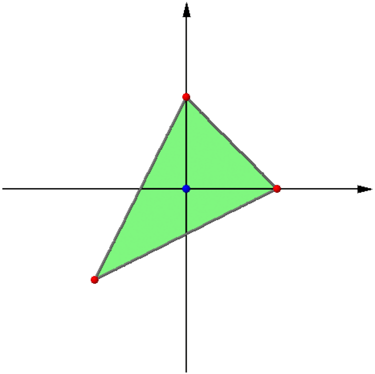
Example:



- Let $v_1 = (1, 0)$, $v_2 = (0, 1)$, $v_3 = (-1, -1)$ be vertices of $\Delta$.
- The only set of vertices which is not contained in a single face of $\Delta$ is $P = \{1, 2, 3\}$.
- We have $1 \cdot v_1 + 1 \cdot v_2 + 1 \cdot v_3 = 0$.

# Homogeneous Coordinates: Projective Space

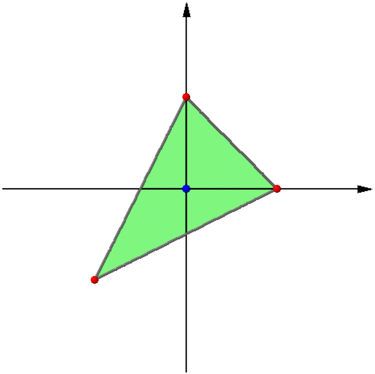
Example:



- Let $v_1 = (1,0)$, $v_2 = (0,1)$, $v_3 = (-1,-1)$ be vertices of $\Delta$.
- The only set of vertices which is not contained in a single face of $\Delta$ is $P = \{1,2,3\}$.
- We have $1 \cdot v_1 + 1 \cdot v_2 + 1 \cdot v_3 = 0$.
- Introduce coordinates $z = (z_1, z_2, z_3)$ on $\mathbb{C}^3$.

# Homogeneous Coordinates: Projective Space

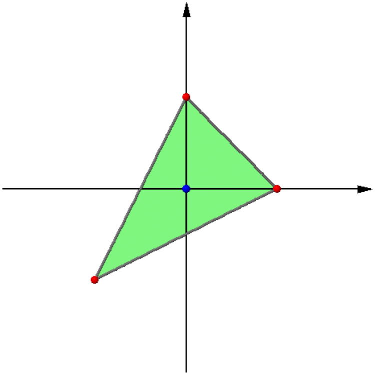
Example:



- Let $v_1 = (1, 0)$, $v_2 = (0, 1)$, $v_3 = (-1, -1)$ be vertices of $\Delta$.
- The only set of vertices which is not contained in a single face of $\Delta$ is $P = \{1, 2, 3\}$.
- We have $1 \cdot v_1 + 1 \cdot v_2 + 1 \cdot v_3 = 0$.
- Introduce coordinates $z = (z_1, z_2, z_3)$ on $\mathbb{C}^3$.
- Consider $\{z : z_i = 0 \ \forall i \in P\} = \{0\}$.

# Homogeneous Coordinates: Projective Space

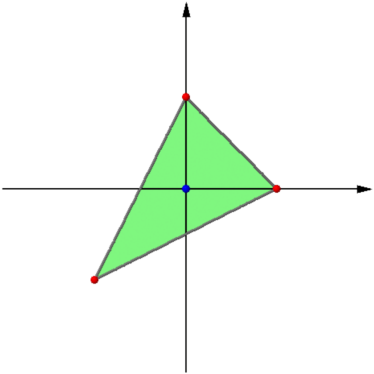
Example:



- Let $v_1 = (1, 0)$, $v_2 = (0, 1)$, $v_3 = (-1, -1)$ be vertices of $\Delta$.
- The only set of vertices which is not contained in a single face of $\Delta$ is $P = \{1, 2, 3\}$.
- We have $1 \cdot v_1 + 1 \cdot v_2 + 1 \cdot v_3 = 0$.
- Introduce coordinates $z = (z_1, z_2, z_3)$ on $\mathbb{C}^3$.
- Consider $\{z : z_i = 0 \ \forall i \in P\} = \{0\}$.
- Let $\mathbb{C}^\times$ act on $(z_1, z_2, z_3)$ via $(\lambda^1 z_1, \lambda^1 z_2, \lambda^1 z_3)$.

# Homogeneous Coordinates: Projective Space

Example:



- Let $v_1 = (1, 0)$, $v_2 = (0, 1)$, $v_3 = (-1, -1)$ be vertices of $\Delta$.
- The only set of vertices which is not contained in a single face of $\Delta$ is $P = \{1, 2, 3\}$.
- We have $1 \cdot v_1 + 1 \cdot v_2 + 1 \cdot v_3 = 0$.
- Introduce coordinates $z = (z_1, z_2, z_3)$ on $\mathbb{C}^3$.
- Consider $\{z : z_i = 0 \ \forall i \in P\} = \{0\}$.
- Let $\mathbb{C}^\times$ act on $(z_1, z_2, z_3)$ via $(\lambda^1 z_1, \lambda^1 z_2, \lambda^1 z_3)$.

This defines a toric variety $\mathbb{P}_\Delta$ as $(\mathbb{C}^3 \setminus \{0\})/\mathbb{C}^\times \simeq \mathbb{P}^2$.

# Homogeneous Coordinates

In general:

- Let $\Sigma$ be the fan spanned by faces of (a triangulation of) $\partial\Delta$.

# Homogeneous Coordinates

In general:

- Let $\Sigma$ be the fan spanned by faces of (a triangulation of) $\partial\Delta$.
- Let $v_1, \ldots, v_n$ be the generators of 1-dimensional cones of $\Sigma$.

# Homogeneous Coordinates

In general:

- Let $\Sigma$ be the fan spanned by faces of (a triangulation of) $\partial\Delta$.
- Let $v_1, \ldots, v_n$ be the generators of 1-dimensional cones of $\Sigma$.
- Introduce a coordinate $z_i$ for each $v_i$.

# Homogeneous Coordinates

In general:

- Let $\Sigma$ be the fan spanned by faces of (a triangulation of) $\partial\Delta$.
- Let $v_1, \ldots, v_n$ be the generators of 1-dimensional cones of $\Sigma$.
- Introduce a coordinate $z_i$ for each $v_i$.
- The exceptional set is $S = \cup_P \{z \in \mathbb{C}^n : z_i = 0\ \forall i \in P\}$ where $P$ runs over all primitive collections — sets of 1-dimensional cones not contained in a single cone of $\Sigma$, such that any proper subset is contained in a single cone.

# Homogeneous Coordinates

In general:

- Let $\Sigma$ be the fan spanned by faces of (a triangulation of) $\partial\Delta$.
- Let $v_1, \ldots, v_n$ be the generators of 1-dimensional cones of $\Sigma$.
- Introduce a coordinate $z_i$ for each $v_i$.
- The exceptional set is $S = \cup_P \{z \in \mathbb{C}^n : z_i = 0\ \forall i \in P\}$ where $P$ runs over all primitive collections — sets of 1-dimensional cones not contained in a single cone of $\Sigma$, such that any proper subset is contained in a single cone.
- On $\mathbb{C}^n \setminus S$ we have the action of $G = ker\left[(\mathbb{C}^\times)^n \to \mathbb{C}^d\right]$ for

$$(\lambda_1, \ldots, \lambda_n) \mapsto \left(\prod_{i=1}^n \lambda_i^{(v_i)_1}, \ldots, \prod_{i=1}^n \lambda_i^{(v_i)_d}\right).$$

# Homogeneous Coordinates

In general:

- Let $\Sigma$ be the fan spanned by faces of (a triangulation of) $\partial\Delta$.
- Let $v_1, \dots, v_n$ be the generators of 1-dimensional cones of $\Sigma$.
- Introduce a coordinate $z_i$ for each $v_i$.
- The exceptional set is $S = \cup_P \{z \in \mathbb{C}^n : z_i = 0 \ \forall i \in P\}$ where $P$ runs over all primitive collections — sets of 1-dimensional cones not contained in a single cone of $\Sigma$, such that any proper subset is contained in a single cone.
- On $\mathbb{C}^n \setminus S$ we have the action of $G = ker \left[(\mathbb{C}^\times)^n \to \mathbb{C}^d\right]$ for

$$(\lambda_1, \dots, \lambda_n) \mapsto \left(\prod_{i=1}^n \lambda_i^{(v_i)_1}, \dots, \prod_{i=1}^n \lambda_i^{(v_i)_d}\right).$$

- This defines the toric variety $\mathbb{P}_\Sigma$ as $(\mathbb{C}^n \setminus S)/G$, which is a good geometric quotient if $\Sigma$ is simplicial (e.g. if $\Sigma$ corresponds to a maximal lattice triangulation of $\partial\Delta$).

# Outline

# Anticanonical Hypersurfaces

- Batyrev's mirror pair of Calabi-Yau anticanonical hypersurfaces $X \subset \mathbb{P}_{\Sigma(\Delta)}$ and $X^\circ \subset \mathbb{P}_{\Sigma(\Delta^\circ)}$ is given by the equation

$$\sum_{w \in \Delta^\circ \cap M} a_w \prod_{i=1}^{n} z_i^{\langle w, v_i \rangle + 1} = 0$$

and its "polar" (may omit interior points of facets of $\Delta^\circ$).

# Anticanonical Hypersurfaces

- Batyrev's mirror pair of Calabi-Yau anticanonical hypersurfaces $X \subset \mathbb{P}_{\Sigma(\Delta)}$ and $X^\circ \subset \mathbb{P}_{\Sigma(\Delta^\circ)}$ is given by the equation

$$\sum_{w \in \Delta^\circ \cap M} a_w \prod_{i=1}^{n} z_i^{\langle w, v_i \rangle + 1} = 0$$

  and its "polar" (may omit interior points of facets of $\Delta^\circ$).
- Q: What are the Hodge numbers of $X$ and $X^\circ$?

# Anticanonical Hypersurfaces

- Batyrev's mirror pair of Calabi-Yau anticanonical hypersurfaces $X \subset \mathbb{P}_{\Sigma(\Delta)}$ and $X^\circ \subset \mathbb{P}_{\Sigma(\Delta^\circ)}$ is given by the equation

$$\sum_{w \in \Delta^\circ \cap M} a_w \prod_{i=1}^{n} z_i^{\langle w, v_i \rangle + 1} = 0$$

  and its "polar" (may omit interior points of facets of $\Delta^\circ$).
- Q: What are the Hodge numbers of $X$ and $X^\circ$?
- A: For $d \geqslant 4$ they can be computed "in terms of $\Delta$:"

$$h^{1,1}(X) = \ell(\Delta) - 1 - d - \sum_{\Gamma} \ell^*(\Gamma) + \sum_{F} \ell^*(F)\ell^*(F^*),$$

  where $\ell(\Delta) = |\Delta \cap N|$, $\Gamma$ runs over codimension-1 faces of $\Delta$, $\ell^*(\Gamma) = |\operatorname{int} \Gamma \cap N|$, $F$ runs over codimension-2 faces of $\Delta$, and $F^*$ is the dual to $F$ face of $\Delta^\circ$.

# Outline

# Geometry, Combinatorics, Computations

Things like

- ambient spaces
- equations of interesting (i.e. Calabi-Yau) hypersurfaces
- Hodge numbers

can be dealt with explicitly using polytope data — facial structure and lattice points on faces. Perfect for computer experimentation!

# Geometry, Combinatorics, Computations

Things like

- ambient spaces
- equations of interesting (i.e. Calabi-Yau) hypersurfaces
- Hodge numbers

can be dealt with explicitly using polytope data — facial structure and lattice points on faces. Perfect for computer experimentation!
Tools:

- PALP — Package for Analyzing Lattice Polytopes, a small package of C programs designed to work using pipes.

Things like

- ambient spaces
- equations of interesting (i.e. Calabi-Yau) hypersurfaces
- Hodge numbers

can be dealt with explicitly using polytope data — facial structure and lattice points on faces. Perfect for computer experimentation!

Tools:

- PALP — Package for Analyzing Lattice Polytopes, a small package of C programs designed to work using pipes.
- Sage — a big package of everything, including PALP.

# Geometry, Combinatorics, Computations

Things like

- ambient spaces
- equations of interesting (i.e. Calabi-Yau) hypersurfaces
- Hodge numbers

can be dealt with explicitly using polytope data — facial structure and lattice points on faces. Perfect for computer experimentation!

Tools:

- PALP — Package for Analyzing Lattice Polytopes, a small package of C programs designed to work using pipes.
- Sage — a big package of everything, including PALP.
- `lattice_polytope` — a module in Sage, uses PALP for some internal operations and gives direct access to it as well.

# Geometry, Combinatorics, Computations

Things like

- ambient spaces
- equations of interesting (i.e. Calabi-Yau) hypersurfaces
- Hodge numbers

can be dealt with explicitly using polytope data — facial structure and lattice points on faces. Perfect for computer experimentation!
Tools:

- PALP — Package for Analyzing Lattice Polytopes, a small package of C programs designed to work using pipes.
- Sage — a big package of everything, including PALP.
- `lattice_polytope` — a module in Sage, uses PALP for some internal operations and gives direct access to it as well.
- Some extra code, which may be added to Sage once it is clean, if there will be general interest.

# Sample Projects

The above tools were used for a number of projects including

- Search for reflexive polytopes with facial structure of a particular form (for “Algebraic K-theory of toric hypersurfaces” by C.F. Doran & M. Kerr)

# Sample Projects

The above tools were used for a number of projects including

- Search for reflexive polytopes with facial structure of a particular form (for “Algebraic K-theory of toric hypersurfaces” by C.F. Doran & M. Kerr)
- Search for “nice” compactifications of a given Calabi-Yau variety.

# Sample Projects

The above tools were used for a number of projects including

- Search for reflexive polytopes with facial structure of a particular form (for “Algebraic K-theory of toric hypersurfaces” by C.F. Doran & M. Kerr)
- Search for “nice” compactifications of a given Calabi-Yau variety.
- Experiments and verification of results on Hodge numbers of complete intersections.

# Sample Projects

The above tools were used for a number of projects including

- Search for reflexive polytopes with facial structure of a particular form (for “Algebraic K-theory of toric hypersurfaces” by C.F. Doran & M. Kerr)
- Search for “nice” compactifications of a given Calabi-Yau variety.
- Experiments and verification of results on Hodge numbers of complete intersections.
- Computation of the Picard lattice of K3 surfaces.

# Sample Projects

The above tools were used for a number of projects including

- Search for reflexive polytopes with facial structure of a particular form (for “Algebraic K-theory of toric hypersurfaces” by C.F. Doran & M. Kerr)
- Search for “nice” compactifications of a given Calabi-Yau variety.
- Experiments and verification of results on Hodge numbers of complete intersections.
- Computation of the Picard lattice of K3 surfaces.
- Exploration of properties of elliptically fibered K3 surfaces.

# Sample Projects

The above tools were used for a number of projects including

- Search for reflexive polytopes with facial structure of a particular form (for “Algebraic K-theory of toric hypersurfaces” by C.F. Doran & M. Kerr)
- Search for “nice” compactifications of a given Calabi-Yau variety.
- Experiments and verification of results on Hodge numbers of complete intersections.
- Computation of the Picard lattice of K3 surfaces.
- Exploration of properties of elliptically fibered K3 surfaces.

Let’s look at elliptic fibrations closer.

# Outline

# Torically Induced Fibrations

(Following works of E. Perevalov & H. Skarke, F. Rohsiepe, and others)

- Elliptic fibrations of a K3 surface $X \subset \mathbb{P}_{\Sigma(\Delta)}$ (where $\dim \Delta = 3$) can be obtained from fibrations of $\mathbb{P}_{\Sigma(\Delta)}$ itself.

# Torically Induced Fibrations

(Following works of E. Perevalov & H. Skarke, F. Rohsiepe, and others)

- Elliptic fibrations of a K3 surface $X \subset \mathbb{P}_{\Sigma(\Delta)}$ (where $\dim \Delta = 3$) can be obtained from fibrations of $\mathbb{P}_{\Sigma(\Delta)}$ itself.
- Morphisms of toric varieties correspond to morphisms of related lattices.

# Torically Induced Fibrations

(Following works of E. Perevalov & H. Skarke, F. Rohsiepe, and others)

- Elliptic fibrations of a K3 surface $X \subset \mathbb{P}_{\Sigma(\Delta)}$ (where $\dim \Delta = 3$) can be obtained from fibrations of $\mathbb{P}_{\Sigma(\Delta)}$ itself.
- Morphisms of toric varieties correspond to morphisms of related lattices.
- We may look for a hyperplane through the origin which intersects all edges of $\Delta$ in integral points (or for a projection of $\Delta^\circ$ in some direction onto a plane).

# Torically Induced Fibrations

(Following works of E. Perevalov & H. Skarke, F. Rohsiepe, and others)

- Elliptic fibrations of a K3 surface $X \subset \mathbb{P}_{\Sigma(\Delta)}$ (where $\dim \Delta = 3$) can be obtained from fibrations of $\mathbb{P}_{\Sigma(\Delta)}$ itself.
- Morphisms of toric varieties correspond to morphisms of related lattices.
- We may look for a hyperplane through the origin which intersects all edges of $\Delta$ in integral points (or for a projection of $\Delta^\circ$ in some direction onto a plane).
- If the intersection (resp. projection) is a reflexive polygon, then we get a fibration of $\mathbb{P}_{\Sigma(\Delta)}$ which induces an elliptic fibration of $X$.

# Torically Induced Fibrations

(Following works of E. Perevalov & H. Skarke, F. Rohsiepe, and others)

- Elliptic fibrations of a K3 surface $X \subset \mathbb{P}_{\Sigma(\Delta)}$ (where $\dim \Delta = 3$) can be obtained from fibrations of $\mathbb{P}_{\Sigma(\Delta)}$ itself.
- Morphisms of toric varieties correspond to morphisms of related lattices.
- We may look for a hyperplane through the origin which intersects all edges of $\Delta$ in integral points (or for a projection of $\Delta^\circ$ in some direction onto a plane).
- If the intersection (resp. projection) is a reflexive polygon, then we get a fibration of $\mathbb{P}_{\Sigma(\Delta)}$ which induces an elliptic fibration of $X$.
- Moreover, exceptional fibers over zero and infinity can be directly seen from the edges of $\Delta$!

# Torically Induced Fibrations

(Following works of E. Perevalov & H. Skarke, F. Rohsiepe, and others)

- Elliptic fibrations of a K3 surface $X \subset \mathbb{P}_{\Sigma(\Delta)}$ (where $\dim \Delta = 3$) can be obtained from fibrations of $\mathbb{P}_{\Sigma(\Delta)}$ itself.
- Morphisms of toric varieties correspond to morphisms of related lattices.
- We may look for a hyperplane through the origin which intersects all edges of $\Delta$ in integral points (or for a projection of $\Delta^\circ$ in some direction onto a plane).
- If the intersection (resp. projection) is a reflexive polygon, then we get a fibration of $\mathbb{P}_{\Sigma(\Delta)}$ which induces an elliptic fibration of $X$.
- Moreover, exceptional fibers over zero and infinity can be directly seen from the edges of $\Delta$!
- With some care the method works for higher dimensions.

# Toric Fibrations: Details

- Let $m_f \in M$ be a primitive vector specifying the slicing hyperplane in $N$ or the direction of projection in $M$.

# Toric Fibrations: Details

- Let $m_f \in M$ be a primitive vector specifying the slicing hyperplane in $N$ or the direction of projection in $M$.
- The projection map $\mathbb{P}_{\Sigma(\Delta)} \to \mathbb{P}^1$ is given by

$$\left( \prod_{\langle m_f, v_i \rangle > 0} z_i^{\langle m_f, v_i \rangle} : \prod_{\langle m_f, v_i \rangle < 0} z_i^{-\langle m_f, v_i \rangle} \right) \in \mathbb{P}^1.$$

# Toric Fibrations: Details

- Let $m_f \in M$ be a primitive vector specifying the slicing hyperplane in $N$ or the direction of projection in $M$.
- The projection map $\mathbb{P}_{\Sigma(\Delta)} \to \mathbb{P}^1$ is given by

$$\left( \prod_{\langle m_f, v_i \rangle > 0} z_i^{\langle m_f, v_i \rangle} : \prod_{\langle m_f, v_i \rangle < 0} z_i^{-\langle m_f, v_i \rangle} \right) \in \mathbb{P}^1.$$

- In a chart we may set all but one of the involved $z_i$'s to 1.
- The remaining one (or its power) serves as an affine coordinate on the base.

# Toric Fibrations: Details

- Let $m_f \in M$ be a primitive vector specifying the slicing hyperplane in $N$ or the direction of projection in $M$.
- The projection map $\mathbb{P}_{\Sigma(\Delta)} \to \mathbb{P}^1$ is given by

$$\left( \prod_{\langle m_f, v_i \rangle > 0} z_i^{\langle m_f, v_i \rangle} : \prod_{\langle m_f, v_i \rangle < 0} z_i^{-\langle m_f, v_i \rangle} \right) \in \mathbb{P}^1 .$$

- In a chart we may set all but one of the involved $z_i$'s to 1.
- The remaining one (or its power) serves as an affine coordinate on the base.
- The divisor class of the fiber is given by

$$\sum_{\langle m_f, v_i \rangle > 0} \langle m_f, v_i \rangle \{z_i = 0\} \sim - \sum_{\langle m_f, v_i \rangle < 0} \langle m_f, v_i \rangle \{z_i = 0\} .$$

# Toric Fibrations: Details

- Let $m_f \in M$ be a primitive vector specifying the slicing hyperplane in $N$ or the direction of projection in $M$.
- The projection map $\mathbb{P}_{\Sigma(\Delta)} \to \mathbb{P}^1$ is given by

$$\left( \prod_{\langle m_f, v_i \rangle > 0} z_i^{\langle m_f, v_i \rangle} : \prod_{\langle m_f, v_i \rangle < 0} z_i^{-\langle m_f, v_i \rangle} \right) \in \mathbb{P}^1.$$

- In a chart we may set all but one of the involved $z_i$'s to 1.
- The remaining one (or its power) serves as an affine coordinate on the base.
- The divisor class of the fiber is given by

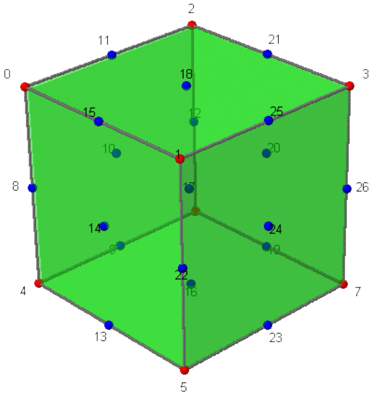
$$\sum_{\langle m_f, v_i \rangle > 0} \langle m_f, v_i \rangle \{z_i = 0\} \sim - \sum_{\langle m_f, v_i \rangle < 0} \langle m_f, v_i \rangle \{z_i = 0\}.$$

- Let's see it in action in 3D using Sage!

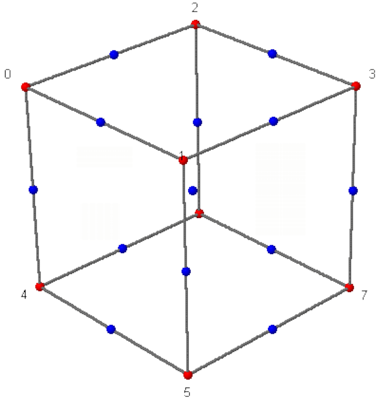
# Outline

# Slicing the Cube

Take a cube:

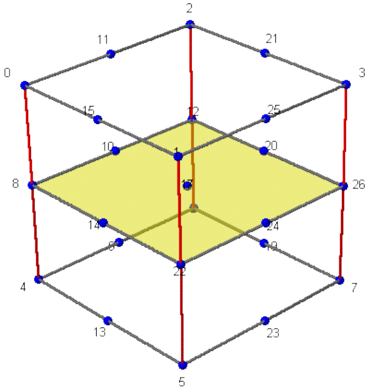



Consider its skeleton:

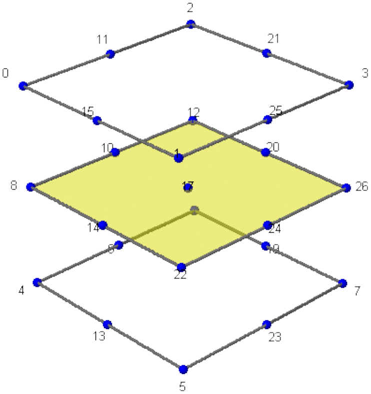


# Slicing the Cube

Slice it:

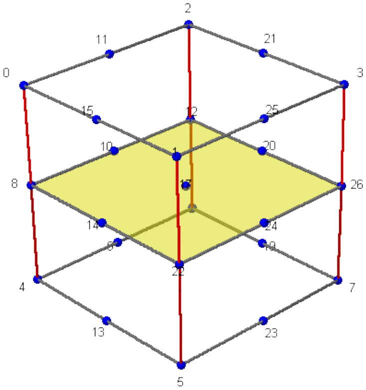



Remove “touching” parts:

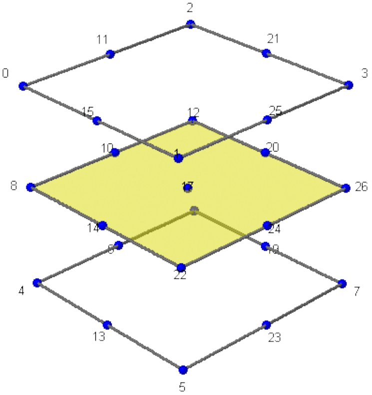


# Slicing the Cube

Slice it:




Remove “touching” parts:




Graphs on top and bottom are extended Dynkin diagrams corresponding to the exceptional fibers over zero and infinity of the induced fibration of $X$!

# Slicing the Cube — not by hands!!!

Using Sage, after some preparation we get

```
...
sage: ef.show()
A fibration of:
A polytope polar to An octahedron: 3-dimensional, 8 vertices.
Corresponding to:
...
Equation of a hypersurface:
```

$$a_3 z_0^2 z_2^2 z_4^2 z_6^2 z_8^2 z_9^2 + a_4 z_0^2 z_1^2 z_4^2 z_5^2 z_8^2 z_{10}^2 + a_2 z_0^2 z_1^2 z_2^2 z_3^2 z_8 z_9 z_{10} z_{11} + a_6 z_0 z_1 z_2 z_3 z_4 z_5 z_6 z_7 z_8 z_9 z_{10} z_{11} + a_5 z_4^2 z_5^2 z_6^2 z_7^2 z_8 z_9 z_{10} z_{11} + a_1 z_2^2 z_3^2 z_6^2 z_7^2 z_9^2 z_{11}^2 + a_0 z_1^2 z_3^2 z_5^2 z_7^2 z_{10}^2 z_{11}^2$$

```
...
Fiber over (t^(1),1):
```

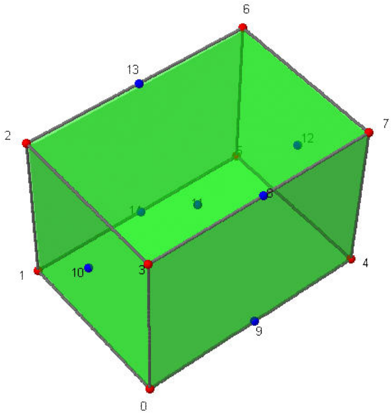
$$(a_3 t^2) z_8^2 z_9^2 + (a_4 t^2) z_8^2 z_{10}^2 + (a_2 t^2 + a_6 t + a_5) z_8 z_9 z_{10} z_{11} + a_1 z_9^2 z_{11}^2 + a_0 z_{10}^2 z_{11}^2$$

```
Top: ExtA7.
Bottom: ExtA7.
F_0: ('I', 8).
F_infinity: ('I', 8).
```

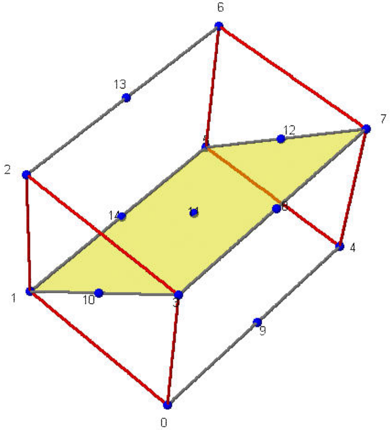
# Outline

# Another Example

Start with:

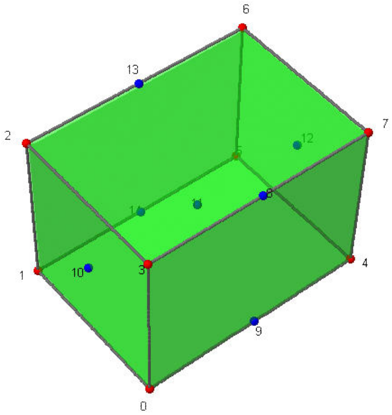



Fibration diagram:

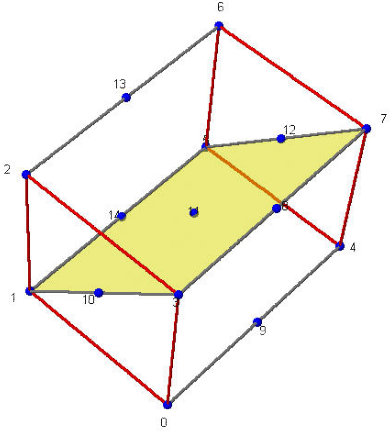


# Another Example

Start with:




Fibration diagram:




Not as satisfying as before — graphs of “top” and “bottom” are not extended Dynkin diagrams.

# Picard Lattice

- Up to linear equivalence, all divisors on $\mathbb{P}_{\Sigma(\Delta)}$ are toric-invariant, i.e. correspond to 1-dimensional cones of $\Sigma(\Delta)$.

# Picard Lattice

- Up to linear equivalence, all divisors on $\mathbb{P}_{\Sigma(\Delta)}$ are toric-invariant, i.e. correspond to 1-dimensional cones of $\Sigma(\Delta)$.
- Intersections of these divisors with $X$ give rise to divisors on $X$.

# Picard Lattice

- Up to linear equivalence, all divisors on $\mathbb{P}_{\Sigma(\Delta)}$ are toric-invariant, i.e. correspond to 1-dimensional cones of $\Sigma(\Delta)$.
- Intersections of these divisors with $X$ give rise to divisors on $X$.
- Some of them may split. Namely, interior points of an edge $E$ of $\Delta$ “split” into $\ell^*(E^*) + 1$ components and the Picard rank of $X$ is, generically,

$$p(X) = \ell(\Delta) - 1 - 3 - \sum_{F,\text{ facets of }\Delta} \ell^*(F) + \sum_{E,\text{ edges of }\Delta} \ell^*(E)\ell^*(E^*).$$

(Same expression as for $h^{1,1}$ in higher dimensions.)

# Picard Lattice

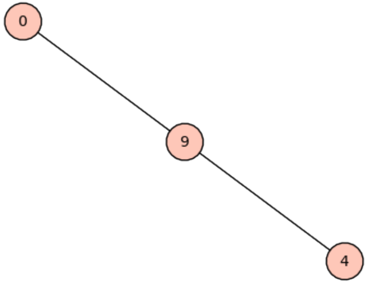
- Up to linear equivalence, all divisors on $\mathbb{P}_{\Sigma(\Delta)}$ are toric-invariant, i.e. correspond to 1-dimensional cones of $\Sigma(\Delta)$.
- Intersections of these divisors with $X$ give rise to divisors on $X$.
- Some of them may split. Namely, interior points of an edge $E$ of $\Delta$ "split" into $\ell^*(E^*) + 1$ components and the Picard rank of $X$ is, generically,

$$p(X) = \ell(\Delta) - 1 - 3 - \sum_{F\text{, facets of }\Delta} \ell^*(F) + \sum_{E\text{, edges of }\Delta} \ell^*(E)\ell^*(E^*).$$

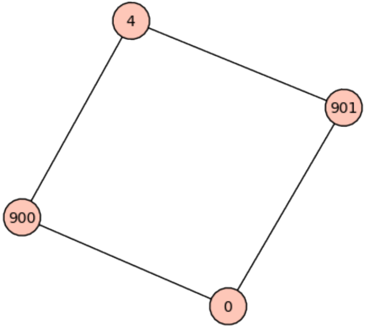
  (Same expression as for $h^{1,1}$ in higher dimensions.)
- After this "splitting" all points on graphs of the "top" and "bottom" correspond to divisors with self-intersection -2. "Edges split" as well and each line represents an intersection of divisors (without multiplicities except for the case o===o).

# Another Example — Splitting

Before splitting:

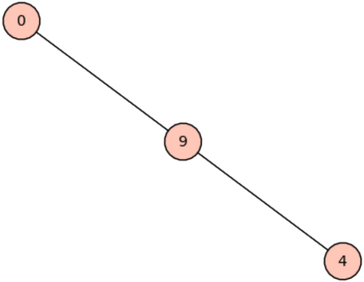



After splitting:

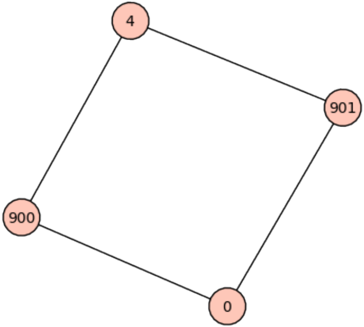


# Another Example — Splitting

Before splitting:




After splitting:




Now we do get an extended Dynkin diagram, as expected for an exceptional fiber!

# Splitting: Details

Intersection of a generic K3 surface and a torus-invariant divisor $\{z_j = 0\}$, where $v_j$ is an interior point of an edge $E$ gives

$$\sum_{w \in \Delta^\circ \cap M} a_w \prod_{i=1}^{n} z_i^{\langle w, v_i \rangle + 1} = \sum_{w \in E^* \cap M} a_w \prod_{i=1,\ldots,\hat{j},\ldots n} z_i^{\langle w, v_i \rangle + 1},$$

since $\langle w, v_j \rangle = -1$ precisely for $w \in E^*$.

# Splitting: Details

Intersection of a generic K3 surface and a torus-invariant divisor $\{z_j = 0\}$, where $v_j$ is an interior point of an edge $E$ gives

$$\sum_{w \in \Delta^\circ \cap M} a_w \prod_{i=1}^{n} z_i^{\langle w, v_i \rangle + 1} = \sum_{w \in E^* \cap M} a_w \prod_{i=1,\dots,\hat{j},\dots n} z_i^{\langle w, v_i \rangle + 1},$$

since $\langle w, v_j \rangle = -1$ precisely for $w \in E^*$.
Let $w_0, \dots, w_s$ be consecutive points along $E^*$. Since $w_k = w_0 + k(w_1 - w_0)$, we can rewrite the above as

$$\prod_{i=1}^{n} z_i^{\langle w_0, v_i \rangle + 1} \sum_{k=0}^{s} a_{w_k} \left( \prod_{t=1}^{n} z_t^{\langle w_1 - w_0, v_t \rangle} \right)^k$$

and the intersection splits into $s = \ell^*(E^*) + 1$ components.

# Outline

# Used Features of Sage

- Interfaces with other programs: integration with PALP, using MAGMA and Maple for (temporarily) missing features of Sage.

# Used Features of Sage

- Interfaces with other programs: integration with PALP, using MAGMA and Maple for (temporarily) missing features of Sage.
- Linear algebra.

# Used Features of Sage

- Interfaces with other programs: integration with PALP, using MAGMA and Maple for (temporarily) missing features of Sage.
- Linear algebra.
- Polynomial rings: working in homogeneous coordinates and charts, factoring, fraction fields, finding roots with given precision.

# Used Features of Sage

- Interfaces with other programs: integration with PALP, using MAGMA and Maple for (temporarily) missing features of Sage.
- Linear algebra.
- Polynomial rings: working in homogeneous coordinates and charts, factoring, fraction fields, finding roots with given precision.
- Graphs: "tops" and "bottoms," splitting, representing Eulerian posets (currently Sage some built-in support for posets).

# Used Features of Sage

- Interfaces with other programs: integration with PALP, using MAGMA and Maple for (temporarily) missing features of Sage.
- Linear algebra.
- Polynomial rings: working in homogeneous coordinates and charts, factoring, fraction fields, finding roots with given precision.
- Graphs: “tops” and “bottoms,” splitting, representing Eulerian posets (currently Sage some built-in support for posets).
- 3D-graphics.

# Used Features of Sage

- Interfaces with other programs: integration with PALP, using MAGMA and Maple for (temporarily) missing features of Sage.
- Linear algebra.
- Polynomial rings: working in homogeneous coordinates and charts, factoring, fraction fields, finding roots with given precision.
- Graphs: “tops” and “bottoms,” splitting, representing Eulerian posets (currently Sage some built-in support for posets).
- 3D-graphics.
- Python: making everything work together.

# Used Features of Sage

- Interfaces with other programs: integration with PALP, using MAGMA and Maple for (temporarily) missing features of Sage.
- Linear algebra.
- Polynomial rings: working in homogeneous coordinates and charts, factoring, fraction fields, finding roots with given precision.
- Graphs: “tops” and “bottoms,” splitting, representing Eulerian posets (currently Sage some built-in support for posets).
- 3D-graphics.
- Python: making everything work together.
- Groups: working with discriminant groups of lattices.

# Outline

# Interfaces with Other Software

- It was quite easy to make an interface to PALP for computing convex hulls, facial structure, polars, and nef partitions.

# Interfaces with Other Software

- It was quite easy to make an interface to PALP for computing convex hulls, facial structure, polars, and nef partitions.
- Interfaces to commercial software are built-in (but you have to have these programs, of course), if a particular function is not supported in the sense of returning a Sage object, at least it is possible to get a string output for any command.

# Interfaces with Other Software

- It was quite easy to make an interface to PALP for computing convex hulls, facial structure, polars, and nef partitions.
- Interfaces to commercial software are built-in (but you have to have these programs, of course), if a particular function is not supported in the sense of returning a Sage object, at least it is possible to get a string output for any command.
- This is convenient to get fast access to features that are not implemented in Sage and wait for someone to implement it.
- E.g. MAGMA was used for graph classification, but now there is no need for this, thanks to Robert Miller!

# Polynomial Rings

- It is convenient to work with hypersurfaces defined by elements of

$$\mathbb{Q}[a_1, \ldots, a_m][z_1, \ldots, z_n].$$

# Polynomial Rings

- It is convenient to work with hypersurfaces defined by elements of

$$\mathbb{Q}[a_1, \ldots, a_m][z_1, \ldots, z_n].$$

- This allows
  - precise arithmetics,
  - convenient monomial manipulation,
  - gathering of monomials when passing to the fiber,
  - even pretty printing (as well as LaTeX code)!

# Polynomial Rings

- It is convenient to work with hypersurfaces defined by elements of

$$\mathbb{Q}[a_1, \ldots, a_m][z_1, \ldots, z_n].$$

- This allows
  - precise arithmetics,
  - convenient monomial manipulation,
  - gathering of monomials when passing to the fiber,
  - even pretty printing (as well as LaTeX code)!
- For classifying all exceptional fibers numerically it was important to find roots with specified precision to make sure that multiple roots are determined as multiple.

# Polynomial Rings

- It is convenient to work with hypersurfaces defined by elements of

$$\mathbb{Q}[a_1, \dots, a_m][z_1, \dots, z_n].$$

- This allows
  - precise arithmetics,
  - convenient monomial manipulation,
  - gathering of monomials when passing to the fiber,
  - even pretty printing (as well as LaTeX code)!
- For classifying all exceptional fibers numerically it was important to find roots with specified precision to make sure that multiple roots are determined as multiple.
- For one of the projects it was important to have fast arithmetics for polynomials over $\mathbb{Z}$ and Sage is the fastest for the moment.

# 3D-Graphics

- Sage (via Jmol) allows working with interactive 3D graphs.

# 3D-Graphics

- Sage (via Jmol) allows working with interactive 3D graphs.
- It is really easy to create your own 3D objects!

# 3D-Graphics

- Sage (via Jmol) allows working with interactive 3D graphs.
- It is really easy to create your own 3D objects!

```
def top3d(self,
        edge_thickness=3, edge_color=(0.5,0.5,0.5),
        point_size=10, point_color=(0,0,1),
        label_color=(0,0,0), label_shift=1.1):
    r"""
    Return the 3d plot of the top edge diagram.
    """
    g = self._s.top
    points = self.polytope().points().columns(copy=False)
    plot = None
    for v in g.vertices():
        pt = points[v]
        plot += point3d([pt], size=point_size, rgbcolor=point_color)
        plot += text3d(str(v), label_shift*pt, rgbcolor=label_color)
    for e in g.edges():
        plot += line3d([points[e[0]], points[e[1]]],
                    thickness=edge_thickness, rgbcolor=edge_color)
    return plot
```

# Outline

# Personal Perspective on Sage

- Functionality is comparable to commercial software packages.

# Personal Perspective on Sage

- Functionality is comparable to commercial software packages.
- It has some bugs, but others have them as well (for a fee).

# Personal Perspective on Sage

- Functionality is comparable to commercial software packages.
- It has some bugs, but others have them as well (for a fee).
- Sage community is very responsive and active.

# Personal Perspective on Sage

- Functionality is comparable to commercial software packages.
- It has some bugs, but others have them as well (for a fee).
- Sage community is very responsive and active.
  - You can get answers in a few minutes even in the night.

# Personal Perspective on Sage

- Functionality is comparable to commercial software packages.
- It has some bugs, but others have them as well (for a fee).
- Sage community is very responsive and active.
  - You can get answers in a few minutes even in the night.
  - Reported zero vector in a basis

# Personal Perspective on Sage

- Functionality is comparable to commercial software packages.
- It has some bugs, but others have them as well (for a fee).
- Sage community is very responsive and active.
  - You can get answers in a few minutes even in the night.
  - Reported zero vector in a basis — patch next day.

# Personal Perspective on Sage

- Functionality is comparable to commercial software packages.
- It has some bugs, but others have them as well (for a fee).
- Sage community is very responsive and active.
  - You can get answers in a few minutes even in the night.
  - Reported zero vector in a basis — patch next day.
  - Asked for easy breaking/combining of notebook cells

# Personal Perspective on Sage

- Functionality is comparable to commercial software packages.
- It has some bugs, but others have them as well (for a fee).
- Sage community is very responsive and active.
  - You can get answers in a few minutes even in the night.
  - Reported zero vector in a basis — patch next day.
  - Asked for easy breaking/combining of notebook cells — got it in the next release.

# Personal Perspective on Sage

- Functionality is comparable to commercial software packages.
- It has some bugs, but others have them as well (for a fee).
- Sage community is very responsive and active.
  - You can get answers in a few minutes even in the night.
  - Reported zero vector in a basis — patch next day.
  - Asked for easy breaking/combining of notebook cells — got it in the next release.
- Python is more convenient for writing code than exclusive languages of other programs.

# Personal Perspective on Sage

- Functionality is comparable to commercial software packages.
- It has some bugs, but others have them as well (for a fee).
- Sage community is very responsive and active.
  - You can get answers in a few minutes even in the night.
  - Reported zero vector in a basis — patch next day.
  - Asked for easy breaking/combining of notebook cells — got it in the next release.
- Python is more convenient for writing code than exclusive languages of other programs.
- Sage server/notebook is a really convenient feature for working on fast computers from slow laptops or random machines.

# Personal Perspective on Sage

- Functionality is comparable to commercial software packages.
- It has some bugs, but others have them as well (for a fee).
- Sage community is very responsive and active.
  - You can get answers in a few minutes even in the night.
  - Reported zero vector in a basis — patch next day.
  - Asked for easy breaking/combining of notebook cells — got it in the next release.
- Python is more convenient for writing code than exclusive languages of other programs.
- Sage server/notebook is a really convenient feature for working on fast computers from slow laptops or random machines.
- If you do want to have a personal installation, you download the archive, follow simple instructions, and it works. If you have problems, it is clear where to get help.
- From personal experience, this can be more complicated (thus annoying) for much simpler free software.