Fundamental and Advanced Algorithms in Singular

Janko Boehm, Yue Ren

University of Kaiserslautern

3C in G Workshop on Computational Algebra
Cambridge, April 18, 2017

Singular

- Computer algebra system for polynomial computations, over 30 development teams worldwide, over 140 libraries for advanced topics.

https://www.singular.uni-kl.de/

- Computer algebra system for polynomial computations, over 30 development teams worldwide, over 140 libraries for advanced topics.

https://www.singular.uni-kl.de/

- Special emphasis on algebraic geometry, commutative and non-commutative algebra, singularity theory, packages for convex and tropical geometry.

Outline

- Warm-up, Gröbner bases
- Normalization, Adjoint Curves, Classification of Singularities
- Parallel Computations
- Resolution of Singularities
- Modular Methods
- Massively Parallel Computations

Outline

- Warm-up, Gröbner bases
- Normalization, Adjoint Curves, Classification of Singularities
- Parallel Computations
- Resolution of Singularities
- Modular Methods
- Massively Parallel Computations

- Primary Decomposition
- Standard Bases and Associated Graded Ring
- Convex Geometry
- Computation of Tropical Varieties

Outline

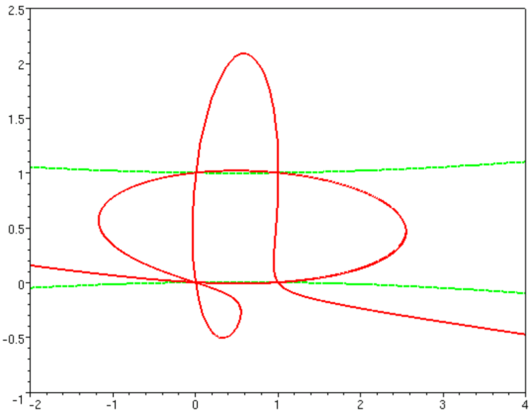
- Warm-up, Gröbner bases
- Normalization, Adjoint Curves, Classification of Singularities
- Parallel Computations
- Resolution of Singularities
- Modular Methods
- Massively Parallel Computations

- Primary Decomposition
- Standard Bases and Associated Graded Ring
- Convex Geometry
- Computation of Tropical Varieties

- Computing the GIT-Fan
- Feynman Integrals and Tropical Mirror Symmetry

Parametrizing Rational Curves

Example



Example

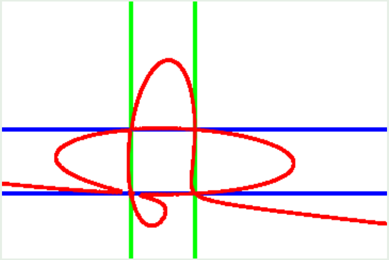
We consider the degree-5 curve with equation

$$x^5 + 10x^4y + 20x^3y^2 + 130x^2y^3 - 20xy^4 + 20y^5 - 2x^4z - 40x^3yz - 150x^2y^2z - 90xy^3z - 40y^4z + x^3z^2 + 30x^2yz^2 + 110xy^2z^2 + 20y^3z^2 = 0.$$

Example

We consider the degree-5 curve with equation

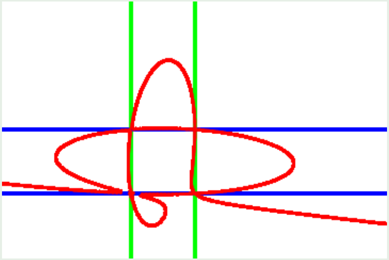
$$x^5 + 10x^4y + 20x^3y^2 + 130x^2y^3 - 20xy^4 + 20y^5 - 2x^4z - 40x^3yz - 150x^2y^2z - 90xy^3z - 40y^4z + x^3z^2 + 30x^2yz^2 + 110xy^2z^2 + 20y^3z^2 = 0.$$



Example

We consider the degree-5 curve with equation

$$x^5 + 10x^4y + 20x^3y^2 + 130x^2y^3 - 20xy^4 + 20y^5 - 2x^4z$$
$$- 40x^3yz - 150x^2y^2z - 90xy^3z - 40y^4z + x^3z^2 + 30x^2yz^2$$
$$+ 110xy^2z^2 + 20y^3z^2 = 0.$$



Genus Formula. $p_g(C) = p_a(C) - \delta(C) = p_a(C) - \sum_{P \in \mathrm{Sing}(\Gamma)} \delta_P(C)$

Parametrizing Rational Curves

Example

```
> ring R = 0, (x,y,z), dp;
> poly f = x5+10x4y+20x3y2+130x2y3-20xy4+20y5-2x4z-40x3yz-150x2y2z
           -90xy3z-40y4z+x3z2+30x2yz2+110xy2z2+20y3z2;
> LIB "paraplanecurves.lib";
> genus(f);
  0
> paraPlaneCurve(f);
```

Example

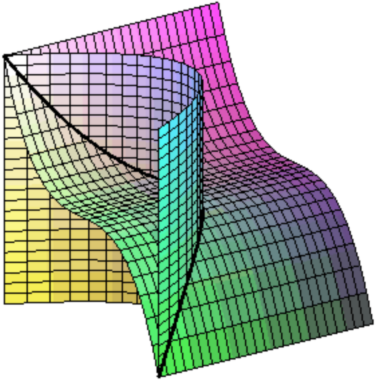
```
> ideal AI = adjointIdeal(f); // requires normalization, integral bases
> AI;
  _[1]=y3-y2z
  _[2]=xy2-xyz
  _[3]=x2y-xyz
  _[4]=x3-x2z
> def Rn = mapToRatNormCurve(f,AI);
> setring(Rn);
```

The Parametrization Algorithm

Example

```
> ideal AI = adjointIdeal(f); // requires normalization, integral bases
> AI;
_[1]=y3-y2z
_[2]=xy2-xyz
_[3]=x2y-xyz
_[4]=x3-x2z
> def Rn = mapToRatNormCurve(f,AI);
> setring(Rn);
> RNC;
RNC[1]=y(2)*y(3)-y(1)*y(4)
RNC[2]=20*y(1)*y(2)-20*y(2)^2+130*y(1)*y(4)
       +20*y(2)*y(4)+10*y(3)*y(4)+y(4)^2
RNC[3]=20*y(1)^2-20*y(1)*y(2)+130*y(1)*y(3)
       +10*y(3)^2+20*y(1)*y(4)+y(3)*y(4)
```

The Parametrization Algorithm



The Parametrization Algorithm

Example

```
> LIB "sing.lib";
> radical(slocus(RNC));
  _[1]=y(4)
  _[2]=y(3)
  _[2]=y(2)
  _[1]=y(1)
> rncAntiCanonicalMap(RNC);
  _[1]=2*y(2)+13*y(4)
  _[2]=y(4)
```

The Parametrization Algorithm

Example

```
> LIB "sing.lib";
> radical(slocus(RNC));
  _[1]=y(4)
  _[2]=y(3)
  _[2]=y(2)
  _[1]=y(1)
> rncAntiCanonicalMap(RNC);
  _[1]=2*y(2)+13*y(4)
  _[2]=y(4)
```

Remark

May require quadratic field extension in even-degree case.

The Main Computational Tool: Gröbner Bases

Division with remainder in one variable successively eliminates the highest power.

Division with remainder in one variable successively eliminates the highest power. In more than one variable we have to fix a monomial ordering (a total ordering compatible with multiplication).

Division with remainder in one variable successively eliminates the highest power. In more than one variable we have to fix a monomial ordering (a total ordering compatible with multiplication).

Divide $x^2 - y^2$ durch $x^2 + y$ und $xy + x$ with respect to lexicographic ordering.

$$\begin{array}{rl} x^2 - y^2 = & 1 \cdot (x^2 + y) + (-y^2 - y) \\ x^2 + y & \\ \hline -y^2 - y & \end{array}$$

The Main Computational Tool: Gröbner Bases

Division with remainder in one variable successively eliminates the highest power. In more than one variable we have to fix a monomial ordering (a total ordering compatible with multiplication).

Divide $x^2 - y^2$ durch $x^2 + y$ und $xy + x$ with respect to lexicographic ordering.

$$\begin{array}{rl} x^2 - y^2 = & 1 \cdot (x^2 + y) + (-y^2 - y) \\ x^2 + y & \\ \hline -y^2 - y & \end{array}$$

so remainder $\neq 0$, but

$$x^2 - y^2 = -y\left(x^2 + y\right) + x\left(xy + x\right) \in I := \left\langle x^2 + y, xy + x \right\rangle$$

Division with remainder in one variable successively eliminates the highest power. In more than one variable we have to fix a monomial ordering (a total ordering compatible with multiplication).
Divide $x^2 - y^2$ durch $x^2 + y$ und $xy + x$ with respect to lexicographic ordering.

$$\begin{array}{ll} x^2 - y^2 = & 1 \cdot (x^2 + y) + (-y^2 - y) \\ \underline{x^2 + y} & \\ -y^2 - y & \end{array}$$

so remainder $\neq 0$, but

$$x^2 - y^2 = -y\left(x^2 + y\right) + x\left(xy + x\right) \in I := \left\langle x^2 + y, xy + x \right\rangle$$

Problem: Lead terms cancel, division algorithm can't do that.

Division with remainder in one variable successively eliminates the highest power. In more than one variable we have to fix a monomial ordering (a total ordering compatible with multiplication).
Divide $x^2 - y^2$ durch $x^2 + y$ und $xy + x$ with respect to lexicographic ordering.

$$\begin{array}{ll} x^2 - y^2 = & 1 \cdot \left(x^2 + y\right) + \left(-y^2 - y\right) \\ \underline{x^2 + y} & \\ -y^2 - y & \end{array}$$

so remainder $\neq 0$, but

$$x^2 - y^2 = -y\left(x^2 + y\right) + x\left(xy + x\right) \in I := \left\langle x^2 + y, xy + x \right\rangle$$

Problem: Lead terms cancel, division algorithm can't do that.
Solution: Add $y^2 + y$ to the divisor set. The result is a **Gröbner basis** G of I. Then

Division with remainder in one variable successively eliminates the highest power. In more than one variable we have to fix a monomial ordering (a total ordering compatible with multiplication).
Divide $x^2 - y^2$ durch $x^2 + y$ und $xy + x$ with respect to lexicographic ordering.

$$\begin{array}{rl} x^2 - y^2 = & 1 \cdot \left(x^2 + y\right) + \left(-y^2 - y\right) \\ x^2 + y & \\ \hline -y^2 - y & \end{array}$$

so remainder $\neq 0$, but

$$x^2 - y^2 = -y\left(x^2 + y\right) + x\left(xy + x\right) \in I := \left\langle x^2 + y, xy + x \right\rangle$$

Problem: Lead terms cancel, division algorithm can't do that.
Solution: Add $y^2 + y$ to the divisor set. The result is a **Gröbner basis** G of I. Then

$$f \in I \Longleftrightarrow NF(f, G) = 0$$

The Main Computational Tool: Gröbner Bases

Example

Gröbner Bases can be used to:

- eliminate variables ($\rightarrow$ birational geometry),
- ideal intersections,
- compute ideal quotients

  $$(I : J) = \{a \in R \mid aJ \subset I\}$$

  for ideals $I, J \subset R$,
- saturations,
- syzygies ($\rightarrow$ homological algebra).

Greuel, G.-M., Pfister, G.: *A Singular Introduction to Commutative Algebra.* Springer.

Normalization

Setup: $A = K[X]/I$ domain.

Definition

The **normalization** $\overline{A}$ of A is the integral closure of A in its quotient field $Q(A)$.

Normalization

Setup: $A = K[X]/I$ domain.

Definition

The **normalization** $\overline{A}$ of A is the integral closure of A in its quotient field $Q(A)$. We call A **normal** if $A = \overline{A}$.

Normalization

Setup: $A = K[X]/I$ domain.

Definition

The **normalization** $\overline{A}$ of A is the integral closure of A in its quotient field $Q(A)$. We call A **normal** if $A = \overline{A}$.

Theorem (Noether)

$\overline{A}$ is a finitely generated A-module.

Normalization

Setup: $A = K[X]/I$ domain.

Definition

The **normalization** $\overline{A}$ of A is the integral closure of A in its quotient field $Q(A)$. We call A **normal** if $A = \overline{A}$.

Theorem (Noether)

$\overline{A}$ is a finitely generated A-module.

Example

Curve $I = \langle x^3 + x^2 - y^2 \rangle \subset K[x, y]$

$$\begin{array}{ccccc} A = K[x,y]/I & \cong & K[t^2-1, t^3-t] & \subset & K[t] \cong \overline{A} \\ \overline{x} & \mapsto & t^2 - 1 & & \\ \overline{y} & \mapsto & t^3 - t & & \end{array}$$

Normalization

Setup: $A = K[X]/I$ domain.

Definition

The **normalization** $\overline{A}$ of A is the integral closure of A in its quotient field $Q(A)$. We call A **normal** if $A = \overline{A}$.

Theorem (Noether)

$\overline{A}$ is a finitely generated A-module.

Example

Curve $I = \langle x^3 + x^2 - y^2 \rangle \subset K[x,y]$

$$\begin{array}{rcccc} A = K[x,y]/I & \cong & K[t^2-1, t^3-t] & \subset & K[t] \cong \overline{A} \\ \overline{x} & \mapsto & t^2-1 & & \\ \overline{y} & \mapsto & t^3-t & & \end{array}$$

As an A-module $\overline{A} = \left\langle 1, \frac{\overline{y}}{\overline{x}} \right\rangle$.

Normalization

Lemma

If $J \subset A$ is an ideal and $0 \neq g \in J$, then

Lemma

If $J \subset A$ is an ideal and $0 \neq g \in J$, then

$$\begin{array}{ccccccc} A & \hookrightarrow & \mathrm{Hom}_A(J,J) & \cong & \frac{1}{g}(gJ :_A J) & \subset & \overline{A} \\ a & \mapsto & a\cdot & & & & \\ & & & \varphi & \mapsto & \frac{\varphi(g)}{g} & \end{array}$$

Lemma

If $J \subset A$ is an ideal and $0 \neq g \in J$, then

$$\begin{array}{ccccccc} A & \hookrightarrow & \mathrm{Hom}_A(J,J) & \cong & \frac{1}{g}(gJ :_A J) & \subset & \overline{A} \\ a & \mapsto & a\cdot & & & & \\ & & & \varphi & \mapsto & \frac{\varphi(g)}{g} & \end{array}$$

Algorithm

Starting from $A_0 = A$ and $J_0 = J$,

Lemma

If $J \subset A$ is an ideal and $0 \neq g \in J$, then

$$\begin{array}{ccccccc} A & \hookrightarrow & \mathrm{Hom}_A(J,J) & \cong & \frac{1}{g}(gJ :_A J) & \subset & \overline{A} \\ a & \mapsto & a\cdot & & & & \\ & & \varphi & \mapsto & \frac{\varphi(g)}{g} & & \end{array}$$

Algorithm

Starting from $A_0 = A$ and $J_0 = J$, setting

$$A_{i+1} = \frac{1}{g}(gJ_i :_{A_i} J_i) \qquad J_i = \sqrt{JA_i}$$

Normalization

Lemma

If $J \subset A$ is an ideal and $0 \neq g \in J$, then

$$\begin{array}{ccccccccc} A & \hookrightarrow & \mathrm{Hom}_A(J,J) & \cong & \frac{1}{g}(gJ :_A J) & \subset & \overline{A} \\ a & \mapsto & a\cdot & & & & \\ & & & \varphi & \mapsto & \frac{\varphi(g)}{g} & \end{array}$$

Algorithm

Starting from $A_0 = A$ and $J_0 = J$, setting

$$A_{i+1} = \frac{1}{g}(gJ_i :_{A_i} J_i) \qquad J_i = \sqrt{JA_i}$$

we get a chain of extensions of reduced Noetherian rings

$$A = A_0 \subset \cdots \subset A_i \subset \cdots \subset A_m = A_{m+1}.$$

Terminates since A is Noetherian.

Grauert-Remmert criterion

Non-normal locus $N(A)$ is contained in **singular locus** $\mathrm{Sing}(A)$.

Non-normal locus $N(A)$ is contained in **singular locus** $\mathrm{Sing}(A)$.

Theorem (Grauert-Remmert)

Let $0 \neq J \subset A$ *be an ideal with* $J = \sqrt{J}$

Grauert-Remmert criterion

Non-normal locus $N(A)$ is contained in **singular locus** $\operatorname{Sing}(A)$.

Theorem (Grauert-Remmert)

Let $0 \neq J \subset A$ *be an ideal with* $J = \sqrt{J}$ *and*

$$N(A) \subset V(J).$$

Non-normal locus $N(A)$ is contained in **singular locus** $\mathrm{Sing}(A)$.

Theorem (Grauert-Remmert)

Let $0 \neq J \subset A$ *be an ideal with* $J = \sqrt{J}$ *and*

$$N(A) \subset V(J).$$

Then A *is normal iff the inclusion*

$$\begin{array}{ccc} A & \hookrightarrow & \mathrm{Hom}_A(J, J) \\ a & \mapsto & a\cdot \end{array}$$

is an isomorphism.

Non-normal locus $N(A)$ is contained in **singular locus** $\operatorname{Sing}(A)$.

Theorem (Grauert-Remmert)

Let $0 \neq J \subset A$ be an ideal with $J = \sqrt{J}$ and

$$N(A) \subset V(J).$$

Then A is normal iff the inclusion

$$\begin{array}{ccc} A & \hookrightarrow & \operatorname{Hom}_A(J, J) \\ a & \mapsto & a\cdot \end{array}$$

is an isomorphism.

$\Longrightarrow$ For $J = \sqrt{\operatorname{Jac}(I)}$ algorithm terminates with $A_m = A_{m+1} = \overline{A}$,

Grauert-Remmert criterion

Non-normal locus $N(A)$ is contained in **singular locus** $\mathrm{Sing}(A)$.

Theorem (Grauert-Remmert)

Let $0 \neq J \subset A$ *be an ideal with* $J = \sqrt{J}$ *and*

$$N(A) \subset V(J).$$

Then A *is normal iff the inclusion*

$$\begin{array}{ccc} A & \hookrightarrow & \mathrm{Hom}_A(J, J) \\ a & \mapsto & a\cdot \end{array}$$

is an isomorphism.

$\Longrightarrow$ For $J = \sqrt{\mathrm{Jac}(I)}$ algorithm terminates with $A_m = A_{m+1} = \overline{A}$, since:

Lemma

$N(A_i) \subset V(\sqrt{JA_i})$

Theorem (BDLSS, 2011)

Suppose

$$\mathrm{Sing}(A) = \{P_1, \ldots, P_r\}$$

Theorem (BDLSS, 2011)

Suppose

$$\mathrm{Sing}(A) = \{P_1, \ldots, P_r\}$$

and

$$A \subset B_i \subset \overline{A}$$

is the ring given by the normalization algorithm applied to P_i *instead of* J

Theorem (BDLSS, 2011)

Suppose

$$\mathrm{Sing}(A) = \{P_1, \dots, P_r\}$$

and

$$A \subset B_i \subset \overline{A}$$

is the ring given by the normalization algorithm applied to P_i instead of J. Then

$$(B_i)_{P_i} = \overline{A_{P_i}}$$
$$(B_i)_Q = A_Q \text{ for all } P_i \neq Q \in \operatorname{Spec} A,$$

Theorem (BDLSS, 2011)

Suppose

$$\mathrm{Sing}(A) = \{P_1, \ldots, P_r\}$$

and

$$A \subset B_i \subset \overline{A}$$

is the ring given by the normalization algorithm applied to P_i *instead of* J. *Then*

$$(B_i)_{P_i} = \overline{A_{P_i}}$$
$$(B_i)_Q = A_Q \text{ for all } P_i \neq Q \in \mathrm{Spec}\, A,$$

and

$$\overline{A} = B_1 + \ldots + B_r.$$

We call B_i *the* **minimal local contribution** *to* $\overline{A}$ *at* P_i.

Adjoint ideals

Setup: $\Gamma \subset \mathbb{P}^r$ integral, non-degenerate projective curve, $\pi : \overline{\Gamma} \to \Gamma$ normalization map, $I(\Gamma) \subsetneq I \subset k[x_0, ..., x_r]$ saturated homogeneous ideal.

Adjoint ideals

Setup: $\Gamma \subset \mathbb{P}^r$ integral, non-degenerate projective curve, $\pi : \overline{\Gamma} \to \Gamma$ normalization map, $I(\Gamma) \subsetneq I \subset k[x_0, ..., x_r]$ saturated homogeneous ideal. Let H be pullback of hyperplane, $\Delta(I)$ pullback of $\mathrm{Proj}(S/I)$.

Adjoint ideals

Setup: $\Gamma \subset \mathbb{P}^r$ integral, non-degenerate projective curve, $\pi : \overline{\Gamma} \to \Gamma$ normalization map, $I(\Gamma) \subsetneq I \subset k[x_0, \ldots, x_r]$ saturated homogeneous ideal. Let H be pullback of hyperplane, $\Delta(I)$ pullback of $\mathrm{Proj}(S/I)$. Then

$$0 \to \widetilde{I}\mathcal{O}_\Gamma \to \pi_*(\widetilde{I}\mathcal{O}_{\overline{\Gamma}}) \to \mathcal{F} \to 0$$

Adjoint ideals

Setup: $\Gamma \subset \mathbb{P}^r$ integral, non-degenerate projective curve, $\pi : \overline{\Gamma} \to \Gamma$ normalization map, $I(\Gamma) \subsetneq I \subset k[x_0, ..., x_r]$ saturated homogeneous ideal. Let H be pullback of hyperplane, $\Delta(I)$ pullback of $\mathrm{Proj}(S/I)$. Then

$$0 \to \widetilde{I}\mathcal{O}_\Gamma \to \pi_*(\widetilde{I}\mathcal{O}_{\overline{\Gamma}}) \to \mathcal{F} \to 0$$

gives for $m \gg 0$ linear maps

$$0 \to I_m / I(\Gamma)_m \stackrel{\overline{\varrho_m}}{\to} H^0\left(\overline{\Gamma}, \mathcal{O}_{\overline{\Gamma}}\left(mH - \Delta(I)\right)\right) \to H^0\left(\Gamma, \mathcal{F}\right) \to 0$$

Definition

I is an **adjoint ideal** of Γ if $\overline{\varrho_m}$ surjective for $m \gg 0$.

Adjoint ideals

Setup: $\Gamma \subset \mathbb{P}^r$ integral, non-degenerate projective curve, $\pi : \overline{\Gamma} \to \Gamma$ normalization map, $I(\Gamma) \subsetneqq I \subset k[x_0, ..., x_r]$ saturated homogeneous ideal. Let H be pullback of hyperplane, $\Delta(I)$ pullback of $\mathrm{Proj}(S/I)$. Then

$$0 \to \widetilde{I}\mathcal{O}_\Gamma \to \pi_*(\widetilde{I}\mathcal{O}_{\overline{\Gamma}}) \to \mathcal{F} \to 0$$

gives for $m \gg 0$ linear maps

$$0 \to I_m / I(\Gamma)_m \overset{\overline{\varrho_m}}{\to} H^0\left(\overline{\Gamma}, \mathcal{O}_{\overline{\Gamma}}\left(mH - \Delta(I)\right)\right) \to H^0\left(\Gamma, \mathcal{F}\right) \to 0$$

Definition

I is an **adjoint ideal** of Γ if $\overline{\varrho_m}$ surjective for $m \gg 0$.

$$h^0\left(\Gamma, \mathcal{F}\right) = \sum\nolimits_{P \in \mathrm{Sing}(\Gamma)} \ell(I_P \overline{\mathcal{O}_{\Gamma,P}} / I_P) \qquad \Longrightarrow$$

Theorem

$$I \text{ adjoint} \iff I_P \overline{\mathcal{O}_{\Gamma,P}} = I_P \text{ for all } P \in \mathrm{Sing}(\Gamma).$$

Conductor is largest ideal with this property.

Definition

Gorenstein adjoint ideal is the unique largest homogeneous ideal $\mathfrak{G} \subset K[x_0, \ldots, x_r]$ with

$$\mathfrak{G}_P = \mathcal{C}_{\mathcal{O}_{\Gamma,P}} \text{ for all } P \in \operatorname{Sing}(\Gamma).$$

Adjoint ideals

Definition

Gorenstein adjoint ideal is the unique largest homogeneous ideal $\mathfrak{G} \subset K[x_0, \ldots, x_r]$ with

$$\mathfrak{G}_P = \mathcal{C}_{\mathcal{O}_{\Gamma,P}} \text{ for all } P \in \mathrm{Sing}(\Gamma).$$

Applications:

Example

If Γ is plane curve of degree n, then $\mathfrak{G}_{n-3}$ cuts out canonical linear series.

Adjoint ideals

Definition

Gorenstein adjoint ideal is the unique largest homogeneous ideal $\mathfrak{G} \subset K[x_0, \ldots, x_r]$ with

$$\mathfrak{G}_P = \mathcal{C}_{\mathcal{O}_{\Gamma,P}} \text{ for all } P \in \mathrm{Sing}(\Gamma).$$

Applications:

Example

If Γ is plane curve of degree n, then $\mathfrak{G}_{n-3}$ cuts out canonical linear series.

Example

If Γ is plane rational of degree n then $\mathfrak{G}_{n-2}$ maps Γ to rational normal curve of degree $n-2$ in $\mathbb{P}^{n-2}$.

Adjoint ideals

Definition

Gorenstein adjoint ideal is the unique largest homogeneous ideal $\mathfrak{G} \subset K[x_0, \ldots, x_r]$ with

$$\mathfrak{G}_P = \mathcal{C}_{\mathcal{O}_{\Gamma,P}} \text{ for all } P \in \mathrm{Sing}(\Gamma).$$

Applications:

Example

If Γ is plane curve of degree n, then $\mathfrak{G}_{n-3}$ cuts out canonical linear series.

Example

If Γ is plane rational of degree n then $\mathfrak{G}_{n-2}$ maps Γ to rational normal curve of degree $n-2$ in $\mathbb{P}^{n-2}$.

Example

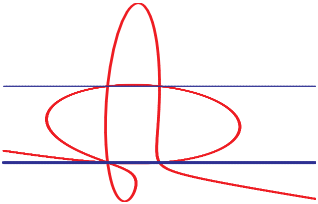
Brill-Noether-Algorithm for computing Riemann-Roch spaces.

Example

Minimal generators of $\mathfrak{G}$ for rational curve of degree 5:

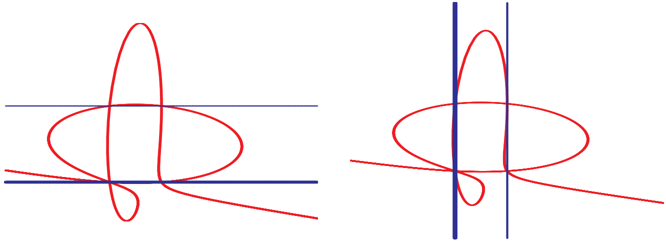
Example

Minimal generators of $\mathfrak{G}$ for rational curve of degree 5:

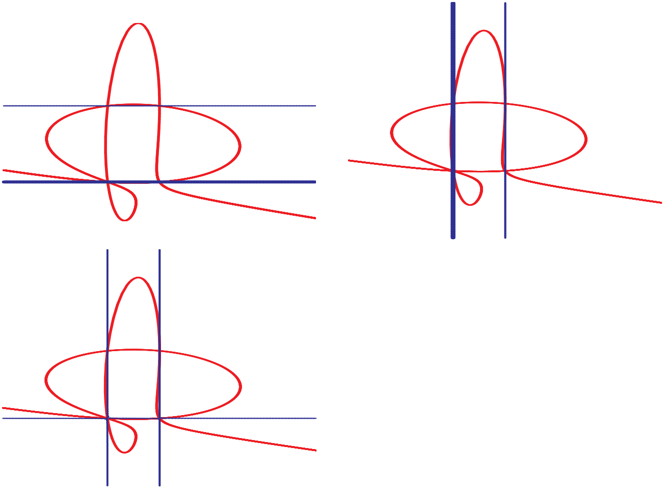


Example

Minimal generators of $\mathfrak{G}$ for rational curve of degree 5:

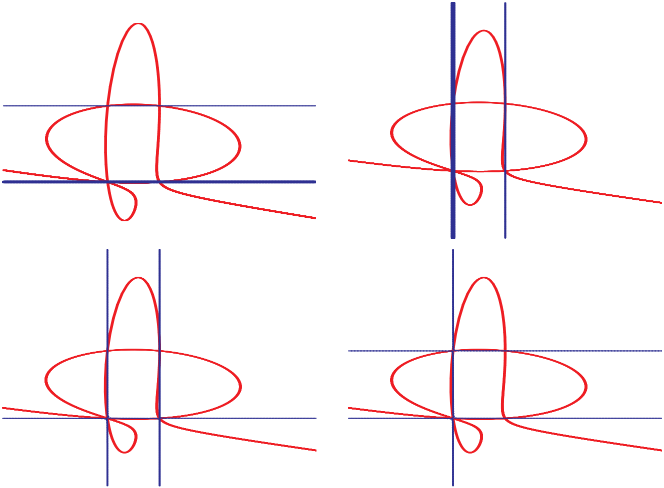


Minimal generators of $\mathfrak{G}$ for rational curve of degree 5:

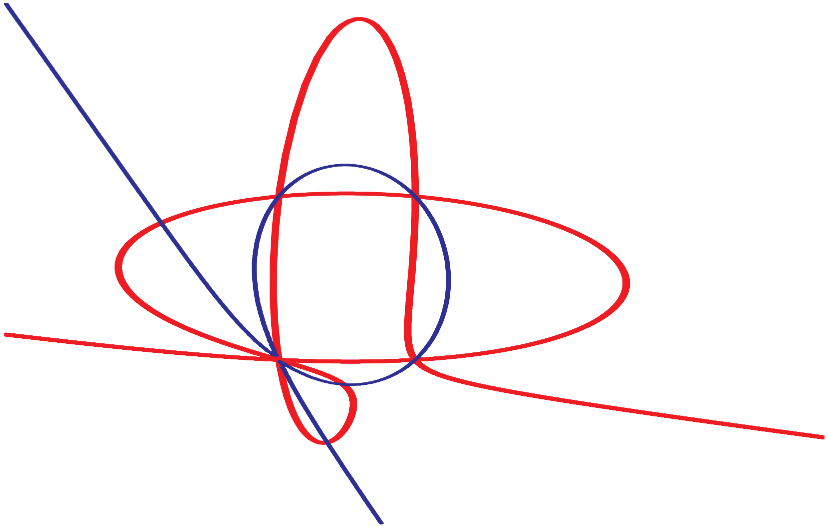


Example

Minimal generators of $\mathfrak{G}$ for rational curve of degree 5:



Example



Local-to-global algorithm

Definition

The **local adjoint ideal** of Γ at $P \in \operatorname{Sing}\Gamma$ is the largest homogeneous ideal $\mathfrak{G}(P) \subset k[x_0, \ldots, x_r]$ with

$$\mathfrak{G}(P)_P = \mathcal{C}_{\mathcal{O}_{\Gamma,P}}$$

Local-to-global algorithm

Definition

The **local adjoint ideal** of Γ at $P \in \operatorname{Sing}\Gamma$ is the largest homogeneous ideal $\mathfrak{G}(P) \subset k[x_0, \ldots, x_r]$ with

$$\mathfrak{G}(P)_P = \mathcal{C}_{\mathcal{O}_{\Gamma,P}}$$

Lemma (BDLP, 2015)

$$\mathfrak{G} = \bigcap_{P \in \operatorname{Sing}\Gamma} \mathfrak{G}(P)$$

The $\mathfrak{G}(P)$ can be computed in parallel via normalization.

Local-to-global algorithm

Definition

The **local adjoint ideal** of Γ at $P \in \operatorname{Sing}\Gamma$ is the largest homogeneous ideal $\mathfrak{G}(P) \subset k[x_0, \ldots, x_r]$ with

$$\mathfrak{G}(P)_P = \mathcal{C}_{\mathcal{O}_{\Gamma,P}}$$

Lemma (BDLP, 2015)

$$\mathfrak{G} = \bigcap_{P \in \operatorname{Sing}\Gamma} \mathfrak{G}(P)$$

The $\mathfrak{G}(P)$ can be computed in parallel via normalization.

Algorithm (BDLP, 2015)

If $\frac{1}{d}U$ is the minimal local contribution at P then

$$\mathfrak{G}(P) = (d : U)^h$$

If $\Gamma \subset \mathbb{P}^2$ has a singularity of type A_n at $P = (0:0:1)$, then given by

$$f = T^2 + W^{n+1} \quad \text{with} \quad T, W \in \mathbb{C}[[x, y]].$$

If $\Gamma \subset \mathbb{P}^2$ has a singularity of type A_n at $P = (0:0:1)$, then given by

$$f = T^2 + W^{n+1} \quad \text{with} \quad T, W \in \mathbb{C}[[x, y]].$$

Compute $T_j = T + O(j+1)$ inductively.

Special types of singularities

If $\Gamma \subset \mathbb{P}^2$ has a singularity of type A_n at $P = (0:0:1)$, then given by

$$f = T^2 + W^{n+1} \quad \text{with} \quad T, W \in \mathbb{C}[[x, y]].$$

Compute $T_j = T + O(j+1)$ inductively.

Lemma

If $P = (0,0)$ is of type A_n and $s = \left\lfloor \frac{n+1}{2} \right\rfloor$, then

$$\mathfrak{G}(P) = \langle x^s,\ T_{s-1},\ y^s \rangle^h \subset \mathbb{C}[x, y, z]$$

Special types of singularities

If $\Gamma \subset \mathbb{P}^2$ has a singularity of type A_n at $P = (0:0:1)$, then given by

$$f = T^2 + W^{n+1} \quad \text{with} \quad T, W \in \mathbb{C}[[x, y]].$$

Compute $T_j = T + O(j+1)$ inductively.

Lemma

If $P = (0,0)$ is of type A_n and $s = \left\lfloor \frac{n+1}{2} \right\rfloor$, then

$$\mathfrak{G}(P) = \langle x^s,\ T_{s-1},\ y^s \rangle^h \subset \mathbb{C}[x, y, z]$$

Similar results for D_n, E_n and other singularities in Arnold's list.

Example

$f = x^4 - y^2 + x^5$ with A_3 singularity. Then $\mathfrak{G}(P) = \langle x^2, y \rangle$.

Example

```
> LIB("parallel.lib","random.lib");
> ring R = 0,x(1..4),dp;
> ideal I = randomid(maxideal(3),3,100);
```

Parallel Computations in SINGULAR

Example

```
> LIB("parallel.lib","random.lib");
> ring R = 0,x(1..4),dp;
> ideal I = randomid(maxideal(3),3,100);
> proc sizeStd(ideal I, string monord){
      def R = basering; list RL = ringlist(R);
      RL[3][1][1] = monord; def S = ring(RL); setring(S);
      return(size(std(imap(R,I))));}
```

Example

```
> LIB("parallel.lib","random.lib");
> ring R = 0,x(1..4),dp;
> ideal I = randomid(maxideal(3),3,100);
> proc sizeStd(ideal I, string monord){
      def R = basering; list RL = ringlist(R);
      RL[3][1][1] = monord; def S = ring(RL); setring(S);
      return(size(std(imap(R,I))));}
> list commands = "sizeStd","sizeStd";
> list args = list(I,"lp"),list(I,"dp");
```

Example

```
> LIB("parallel.lib","random.lib");
> ring R = 0,x(1..4),dp;
> ideal I = randomid(maxideal(3),3,100);
> proc sizeStd(ideal I, string monord){
      def R = basering; list RL = ringlist(R);
      RL[3][1][1] = monord; def S = ring(RL); setring(S);
      return(size(std(imap(R,I))));}
> list commands = "sizeStd","sizeStd";
> list args = list(I,"lp"),list(I,"dp");
> parallelWaitFirst(commands, args);
  [1] empty list
  [2] 11
```

Parallel Computations in SINGULAR

Example

```
> LIB("parallel.lib","random.lib");
> ring R = 0,x(1..4),dp;
> ideal I = randomid(maxideal(3),3,100);
> proc sizeStd(ideal I, string monord){
      def R = basering; list RL = ringlist(R);
      RL[3][1][1] = monord; def S = ring(RL); setring(S);
      return(size(std(imap(R,I))));}
> list commands = "sizeStd","sizeStd";
> list args = list(I,"lp"),list(I,"dp");
> parallelWaitFirst(commands, args);
  [1] empty list
  [2] 11
> parallelWaitAll(commands, args);
  [1] 55
  [2] 11
```

Parallelization

There are algorithms whose basic strategy is inherently parallel, whereas others are sequential in nature.

Parallelization

There are algorithms whose basic strategy is inherently parallel, whereas others are sequential in nature.

Example

- Normalization is inherently sequential.
- Local-to-global algorithms for normalization and adjoint ideal are parallel, if the singular locus decomposes.
- Villamayor's constructive version of Hironaka's desingularization theorem is inherently parallel by the iterative use of blow-ups in charts.
- Modular methods can be used to turn sequential algorithms over $\mathbb{Q}$ into parallel ones.

Example: Resolution of Singularities

Theorem (Hironaka, 1964)

For every algebraic variety over a field K with $\operatorname{char} K = 0$ a desingularization can be obtained by a finite sequence of blow-ups along smooth centers.

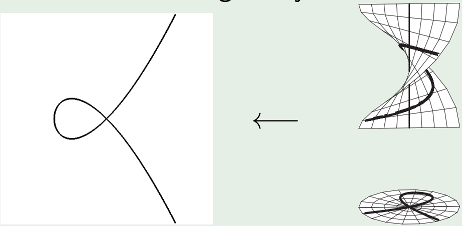
Example: Resolution of Singularities

Theorem (Hironaka, 1964)

For every algebraic variety over a field K with $\operatorname{char} K = 0$ *a desingularization can be obtained by a finite sequence of blow-ups along smooth centers.*

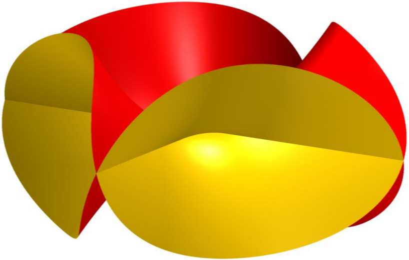
Example

Blow-up of the node resolves the singularity

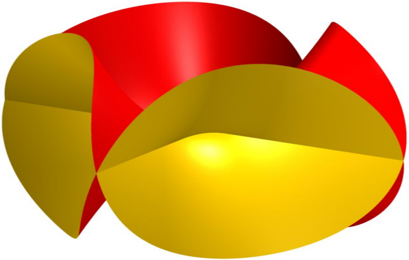


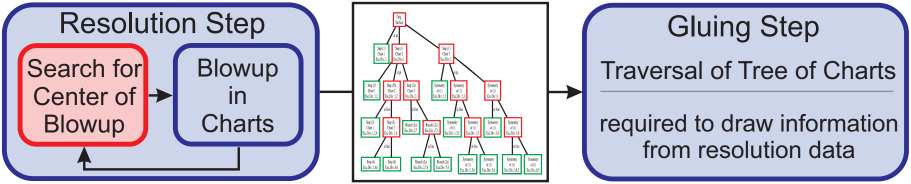
by replacing it by a line of points corresponding to its tangent directions, hence separating the two branches of the curve.

Example:
$x^2 - y^2z^2 = 0$



Example:
$x^2 - y^2 z^2 = 0$





Hironaka Resolution of Singularities

Example

```
> LIB "resolve.lib";
> ring R= 0,(x,y,z),dp;
> ideal I = x2-y2z2;
> list L = resolve(I);
> def S1 = L[1][1];
> setring S1;
> showBO(BO);
  ==== Ambient Space:
  _[1]=0
  ==== Ideal of Variety:
  _[1]=y(1)^2-1
  ==== Exceptional Divisors:
  ...
```

Modular computations

- Many exact computations in computer algebra are carried out over $\mathbb{Q}$ and extensions thereof.

Modular computations

- Many exact computations in computer algebra are carried out over $\mathbb{Q}$ and extensions thereof.
- Modular techniques are an important tool to improve performance of algorithms over $\mathbb{Q}$.

- Many exact computations in computer algebra are carried out over $\mathbb{Q}$ and extensions thereof.
- Modular techniques are an important tool to improve performance of algorithms over $\mathbb{Q}$.
- Fundamental approach:
  1. Compute modulo primes.

Modular computations

- Many exact computations in computer algebra are carried out over $\mathbb{Q}$ and extensions thereof.
- Modular techniques are an important tool to improve performance of algorithms over $\mathbb{Q}$.
- Fundamental approach:
  1. Compute modulo primes.
  2. Reconstruct result over $\mathbb{Q}$.

Modular computations

- Many exact computations in computer algebra are carried out over $\mathbb{Q}$ and extensions thereof.
- Modular techniques are an important tool to improve performance of algorithms over $\mathbb{Q}$.
- Fundamental approach:
  1. Compute modulo primes.
  2. Reconstruct result over $\mathbb{Q}$.
- Benefits:
  - Avoid intermediate coefficient growth.

Modular computations

- Many exact computations in computer algebra are carried out over $\mathbb{Q}$ and extensions thereof.
- Modular techniques are an important tool to improve performance of algorithms over $\mathbb{Q}$.
- Fundamental approach:
  1. Compute modulo primes.
  2. Reconstruct result over $\mathbb{Q}$.
- Benefits:
  - Avoid intermediate coefficient growth.
  - Obtain parallel version of the algorithm.

Modular computations

- Many exact computations in computer algebra are carried out over $\mathbb{Q}$ and extensions thereof.
- Modular techniques are an important tool to improve performance of algorithms over $\mathbb{Q}$.
- Fundamental approach:
  1. Compute modulo primes.
  2. Reconstruct result over $\mathbb{Q}$.
- Benefits:
  - Avoid intermediate coefficient growth.
  - Obtain parallel version of the algorithm.
- Goal:
  General reconstruction scheme for algorithms in commutative algebra, algebraic geometry, number theory.

Example

Compute

$$\frac{3}{4}+\frac{1}{3}=\frac{13}{12}$$

using modular techniques:

Example

Compute

$$\frac{3}{4}+\frac{1}{3}=\frac{13}{12}$$

using modular techniques:

$$\begin{array}{cccccccccc} & & \mathbb{Z}/5 & \times & \mathbb{Z}/7 & \times & \mathbb{Z}/11 & \times & \mathbb{Z}/101 & \cong & \mathbb{Z}/38885 \\ \frac{3}{4} & \mapsto & (\,\overline{2} & , & \overline{6} & , & \overline{9} & , & \overline{26}\,) & & \end{array}$$

Example

Compute

$$\frac{3}{4} + \frac{1}{3} = \frac{13}{12}$$

using modular techniques:

$$\begin{array}{ccccccccccc} & & \mathbb{Z}/5 & \times & \mathbb{Z}/7 & \times & \mathbb{Z}/11 & \times & \mathbb{Z}/101 & \cong & \mathbb{Z}/38885 \\ \frac{3}{4} & \mapsto & (\ \overline{2} & , & \overline{6} & , & \overline{9} & , & \overline{26}\ ) & & \\ & & & & & + & & & & & \\ \frac{1}{3} & \mapsto & (\ \overline{2} & , & \overline{5} & , & \overline{4} & , & \overline{34}\ ) & & \end{array}$$

Example

Compute

$$\frac{3}{4} + \frac{1}{3} = \frac{13}{12}$$

using modular techniques:

$$\begin{array}{ccccccccccc}
 & & \mathbb{Z}/5 & \times & \mathbb{Z}/7 & \times & \mathbb{Z}/11 & \times & \mathbb{Z}/101 & \cong & \mathbb{Z}/38885 \\
\frac{3}{4} & \mapsto & (\ \overline{2} & , & \overline{6} & , & \overline{9} & , & \overline{26}\ ) & & \\
 & & & & & + & & & & & \\
\frac{1}{3} & \mapsto & (\ \overline{2} & , & \overline{5} & , & \overline{4} & , & \overline{34}\ ) & & \\
 & & & & & \shortparallel & & & & & \\
 & & (\ \overline{4} & , & \overline{4} & , & \overline{2} & , & \overline{60}\ ) & &
\end{array}$$

Example

Compute

$$\frac{3}{4}+\frac{1}{3}=\frac{13}{12}$$

using modular techniques:

$$\begin{array}{ccccccccccccc}
 & & \mathbb{Z}/5 & \times & \mathbb{Z}/7 & \times & \mathbb{Z}/11 & \times & \mathbb{Z}/101 & & \cong & \mathbb{Z}/38885 \\
\frac{3}{4} & \mapsto & (\ \overline{2} & , & \overline{6} & , & \overline{9} & , & \overline{26} &) & & \\
 & & & & & + & & & & & & \\
\frac{1}{3} & \mapsto & (\ \overline{2} & , & \overline{5} & , & \overline{4} & , & \overline{34} &) & & \\
 & & & & & \shortparallel & & & & & & \\
 & & (\ \overline{4} & , & \overline{4} & , & \overline{2} & , & \overline{60} &) & \mapsto & \overline{22684}
\end{array}$$

Modular computations

Example

Compute

$$\frac{3}{4} + \frac{1}{3} = \frac{13}{12}$$

using modular techniques:

$$\begin{array}{ccccccccccc}
 & & \mathbb{Z}/5 & \times & \mathbb{Z}/7 & \times & \mathbb{Z}/11 & \times & \mathbb{Z}/101 & \cong & \mathbb{Z}/38885 \\
\frac{3}{4} & \mapsto & (\,\overline{2} & , & \overline{6} & , & \overline{9} & , & \overline{26}\,) & & \\
 & & & & & + & & & & & \\
\frac{1}{3} & \mapsto & (\,\overline{2} & , & \overline{5} & , & \overline{4} & , & \overline{34}\,) & & \\
 & & & & & \shortparallel & & & & & \\
 & & (\,\overline{4} & , & \overline{4} & , & \overline{2} & , & \overline{60}\,) & \mapsto & \overline{22684}
\end{array}$$

How to obtain a rational number from $\overline{22684}$?

Theorem (Kornerup, Gregory, 1983)

The **Farey map**

$$\left\{ \tfrac{a}{b} \in \mathbb{Q} \;\middle|\; \begin{array}{c} \gcd(a,b)=1 \\ \gcd(b,N)=1 \end{array} \quad |a|\,,|b| \le \sqrt{(N-1)/2} \right\} \longrightarrow \mathbb{Z}/N$$

$$\tfrac{a}{b} \longmapsto \overline{a}\cdot\overline{b}^{-1}$$

Theorem (Kornerup, Gregory, 1983)

The **Farey map**

$$\left\{ \tfrac{a}{b} \in \mathbb{Q} \;\middle|\; \begin{array}{c} \gcd(a,b)=1 \\ \gcd(b,N)=1 \end{array} \quad |a|\,,|b| \leq \sqrt{(N-1)/2} \right\} \longrightarrow \mathbb{Z}/N$$

$$\tfrac{a}{b} \longmapsto \overline{a} \cdot \overline{b}^{-1}$$

is injective.

Rational reconstruction

Theorem (Kornerup, Gregory, 1983)

The **Farey map**

$$\left\{ \tfrac{a}{b} \in \mathbb{Q} \;\middle|\; \begin{array}{c} \gcd(a,b)=1 \\ \gcd(b,N)=1 \end{array} \quad |a|\,,|b| \leq \sqrt{(N-1)/2} \right\} \longrightarrow \mathbb{Z}/N$$

$$\tfrac{a}{b} \longmapsto \overline{a} \cdot \overline{b}^{-1}$$

is injective. Efficient algorithm for preimage.

Rational reconstruction

Theorem (Kornerup, Gregory, 1983)

The **Farey map**

$$\left\{ \tfrac{a}{b} \in \mathbb{Q} \;\middle|\; \begin{array}{c} \gcd(a,b) = 1 \\ \gcd(b,N) = 1 \end{array} \quad |a|\,, |b| \leq \sqrt{(N-1)/2} \right\} \longrightarrow \mathbb{Z}/N$$

$$\tfrac{a}{b} \longmapsto \overline{a} \cdot \overline{b}^{-1}$$

is injective. Efficient algorithm for preimage.

Example

Indeed, in the above example

$$\left\{ \tfrac{a}{b} \in \mathbb{Q} \;\middle|\; \begin{array}{c} \gcd(a,b) = 1 \\ \gcd(b,38885) = 1 \end{array} \quad |a|\,, |b| \leq 139 \right\} \longrightarrow \mathbb{Z}/38885$$

Rational reconstruction

Theorem (Kornerup, Gregory, 1983)

The **Farey map**

$$\left\{ \tfrac{a}{b} \in \mathbb{Q} \;\middle|\; \begin{array}{c} \gcd(a,b)=1 \\ \gcd(b,N)=1 \end{array} \quad |a|\,,|b| \leq \sqrt{(N-1)/2} \right\} \longrightarrow \mathbb{Z}/N$$

$$\tfrac{a}{b} \longmapsto \overline{a}\cdot\overline{b}^{-1}$$

is injective. Efficient algorithm for preimage.

Example

Indeed, in the above example

$$\left\{ \tfrac{a}{b} \in \mathbb{Q} \;\middle|\; \begin{array}{c} \gcd(a,b)=1 \\ \gcd(b,38885)=1 \end{array} \quad |a|\,,|b| \leq 139 \right\} \longrightarrow \mathbb{Z}/38885$$

$$\tfrac{13}{12} \longmapsto \overline{22684}$$

Basic concept for modular computations

1. Compute result over $\mathbb{Z}/p_i$ for distinct primes $p_1, \ldots, p_r$.

Basic concept for modular computations

1. Compute result over $\mathbb{Z}/p_i$ for distinct primes $p_1, \ldots, p_r$.
2. For $N = p_1 \cdot \ldots \cdot p_r$ compute lift w.r.t Chinese remainder isomorphism

$$\mathbb{Z}/N \;\cong\; \mathbb{Z}/p_1 \times \ldots \times \mathbb{Z}/p_r$$

Basic concept for modular computations

1. Compute result over $\mathbb{Z}/p_i$ for distinct primes $p_1, \ldots, p_r$.
2. For $N = p_1 \cdot \ldots \cdot p_r$ compute lift w.r.t Chinese remainder isomorphism

$$\mathbb{Z}/N \;\cong\; \mathbb{Z}/p_1 \times \ldots \times \mathbb{Z}/p_r$$

3. If exists, compute preimage w.r.t injective Farey map.

Basic concept for modular computations

1. Compute result over $\mathbb{Z}/p_i$ for distinct primes $p_1, \ldots, p_r$.
2. For $N = p_1 \cdot \ldots \cdot p_r$ compute lift w.r.t Chinese remainder isomorphism

$$\mathbb{Z}/N \cong \mathbb{Z}/p_1 \times \ldots \times \mathbb{Z}/p_r$$

3. If exists, compute preimage w.r.t injective Farey map.
4. Verify correctness of lift.

Basic concept for modular computations

1. Compute result over $\mathbb{Z}/p_i$ for distinct primes $p_1, \ldots, p_r$.
2. For $N = p_1 \cdot \ldots \cdot p_r$ compute lift w.r.t Chinese remainder isomorphism

$$\mathbb{Z}/N \;\cong\; \mathbb{Z}/p_1 \times \ldots \times \mathbb{Z}/p_r$$

3. If exists, compute preimage w.r.t injective Farey map.
4. Verify correctness of lift.

This will yield correct result, provided

- N is large enough s.t. the $\mathbb{Q}$-result is in source of Farey map, and

Basic concept for modular computations

1. Compute result over $\mathbb{Z}/p_i$ for distinct primes $p_1, \ldots, p_r$.
2. For $N = p_1 \cdot \ldots \cdot p_r$ compute lift w.r.t Chinese remainder isomorphism

$$\mathbb{Z}/N \cong \mathbb{Z}/p_1 \times \ldots \times \mathbb{Z}/p_r$$

3. If exists, compute preimage w.r.t injective Farey map.
4. Verify correctness of lift.

This will yield correct result, provided

- N is large enough s.t. the $\mathbb{Q}$-result is in source of Farey map, and
- none of the p_i is bad.

Basic concept for modular computations

1. Compute result over $\mathbb{Z}/p_i$ for distinct primes $p_1, \ldots, p_r$.
2. For $N = p_1 \cdot \ldots \cdot p_r$ compute lift w.r.t Chinese remainder isomorphism

$$\mathbb{Z}/N \cong \mathbb{Z}/p_1 \times \ldots \times \mathbb{Z}/p_r$$

3. If exists, compute preimage w.r.t injective Farey map.
4. Verify correctness of lift.

This will yield correct result, provided

- N is large enough s.t. the $\mathbb{Q}$-result is in source of Farey map, and
- none of the p_i is bad.

Definition

A prime p is called **bad** if the result over $\mathbb{Q}$ does not reduce modulo p to the result over $\mathbb{Z}/p$.

Bad primes in Gröbner basis computations

For $G \subset K[X] = K[x_1, \ldots, x_n]$ and a monomial ordering $>$, let $\mathrm{LM}(G)$ be the set of lead monomials of G.

Bad primes in Gröbner basis computations

For $G \subset K[X] = K[x_1, \ldots, x_n]$ and a monomial ordering $>$, let $\mathrm{LM}(G)$ be the set of lead monomials of G. For $G \subset \mathbb{Z}[X]$ define

$$G_p := \overline{G} \subset \mathbb{Z}/p\,[X].$$

Bad primes in Gröbner basis computations

For $G \subset K[X] = K[x_1, \ldots, x_n]$ and a monomial ordering $>$, let $\mathrm{LM}(G)$ be the set of lead monomials of G. For $G \subset \mathbb{Z}[X]$ define

$$G_p := \overline{G} \subset \mathbb{Z}/p\,[X].$$

Theorem (Arnold, 2003)

Suppose $F = \{f_1, ..., f_r\} \subset \mathbb{Z}[X]$ *with* f_i *primitve,*

Bad primes in Gröbner basis computations

For $G \subset K[X] = K[x_1, \ldots, x_n]$ and a monomial ordering $>$, let $\mathrm{LM}(G)$ be the set of lead monomials of G. For $G \subset \mathbb{Z}[X]$ define

$$G_p := \overline{G} \subset \mathbb{Z}/p\,[X].$$

Theorem (Arnold, 2003)

Suppose $F = \{f_1, ..., f_r\} \subset \mathbb{Z}[X]$ with f_i primitve, and

- *G is the reduced Gröbner basis of $\langle F \rangle \subset \mathbb{Q}[X]$,* For $G \subset K[X] = K[x_1, \ldots, x_n]$ and a monomial ordering $>$, let $\mathrm{LM}(G)$ be the set of lead monomials of G. For $G \subset \mathbb{Z}[X]$ define

$$G_p := \overline{G} \subset \mathbb{Z}/p\,[X].$$

Theorem (Arnold, 2003)

Suppose $F = \{f_1, ..., f_r\} \subset \mathbb{Z}[X]$ with f_i primitve, and

- *G is the reduced Gröbner basis of $\langle F \rangle \subset \mathbb{Q}[X]$,*
- *$G(p)$ is the reduced Gröbner basis of $\langle F_p \rangle$, and*

Bad primes in Gröbner basis computations

For $G \subset K[X] = K[x_1, \ldots, x_n]$ and a monomial ordering $>$, let $\mathrm{LM}(G)$ be the set of lead monomials of G. For $G \subset \mathbb{Z}[X]$ define

$$G_p := \overline{G} \subset \mathbb{Z}/p\,[X].$$

Theorem (Arnold, 2003)

Suppose $F = \{f_1, ..., f_r\} \subset \mathbb{Z}[X]$ with f_i primitve, and

- *G is the reduced Gröbner basis of $\langle F \rangle \subset \mathbb{Q}[X]$,*
- *$G(p)$ is the reduced Gröbner basis of $\langle F_p \rangle$, and*
- *$G_{\mathbb{Z}}$ a minimal strong Gröbnerbasis of $\langle F \rangle \subset \mathbb{Z}[X]$.*

Bad primes in Gröbner basis computations

For $G \subset K[X] = K[x_1, \ldots, x_n]$ and a monomial ordering $>$, let $\mathrm{LM}(G)$ be the set of lead monomials of G. For $G \subset \mathbb{Z}[X]$ define

$$G_p := \overline{G} \subset \mathbb{Z}/p\,[X].$$

Theorem (Arnold, 2003)

Suppose $F = \{f_1, ..., f_r\} \subset \mathbb{Z}[X]$ with f_i primitve, and

- *G is the reduced Gröbner basis of $\langle F \rangle \subset \mathbb{Q}[X]$,*
- *$G(p)$ is the reduced Gröbner basis of $\langle F_p \rangle$, and*
- *$G_{\mathbb{Z}}$ a minimal strong Gröbnerbasis of $\langle F \rangle \subset \mathbb{Z}[X]$. Then*

$$p \text{ does not divide any lead coefficient in } G_{\mathbb{Z}} \iff \mathrm{LM}\, G = \mathrm{LM}\, G(p) \iff G_p = G(p)$$

Bad primes in Gröbner basis computations

For $G \subset K[X] = K[x_1, \ldots, x_n]$ and a monomial ordering $>$, let $\mathrm{LM}(G)$ be the set of lead monomials of G. For $G \subset \mathbb{Z}[X]$ define

$$G_p := \overline{G} \subset \mathbb{Z}/p\,[X].$$

Theorem (Arnold, 2003)

Suppose $F = \{f_1, ..., f_r\} \subset \mathbb{Z}[X]$ with f_i primitve, and

- *G is the reduced Gröbner basis of $\langle F \rangle \subset \mathbb{Q}[X]$,*
- *$G(p)$ is the reduced Gröbner basis of $\langle F_p \rangle$, and*
- *$G_{\mathbb{Z}}$ a minimal strong Gröbnerbasis of $\langle F \rangle \subset \mathbb{Z}[X]$. Then*

$$p \text{ does not divide any lead coefficient in } G_{\mathbb{Z}} \iff \mathrm{LM}\, G = \mathrm{LM}\, G(p) \iff G_p = G(p)$$

that is, p is not bad.

Bad primes in Gröbner basis computations

Example

```
> option("redSB");
> ring R = integer,(x, y, z),lp;
> poly f = x7y5 + x2yz9 + xz11 + y3z9;
> ideal I = groebner(ideal(diff(f, x), diff(f, y), diff(f,z)));
> apply(list(I[1..size(I)]),leadcoef);
```

Example

```
> option("redSB");
> ring R = integer,(x, y, z),lp;
> poly f = x7y5 + x2yz9 + xz11 + y3z9;
> ideal I = groebner(ideal(diff(f, x), diff(f, y), diff(f,z)));
> apply(list(I[1..size(I)]),leadcoef);
13781115527868730344777310464613260 835219122901135172410746088764444 60
12 4 12 12 45349632 12 1473863040 12 22674816 12 3888 12 12 12 13608 12
108 54 6 2 27 3 1 4 2 2 1 216 1 2 3 1 540 12 108 27 3 1 9 3 1 1 1 1 1 7
1 5 1 1
```

Bad primes in Gröbner basis computations

Example

```
> option("redSB");
> ring R = integer,(x, y, z),lp;
> poly f = x7y5 + x2yz9 + xz11 + y3z9;
> ideal I = groebner(ideal(diff(f, x), diff(f, y), diff(f,z)));
> apply(list(I[1..size(I)]),leadcoef);
1378111552786873034477731046461326O 835219122901135172410746O8876444 60
12 4 12 12 45349632 12 1473863040 12 22674816 12 3888 12 12 12 13608 12
108 54 6 2 27 3 1 4 2 2 1 216 1 2 3 1 540 12 108 27 3 1 9 3 1 1 1 1 1 7
1 5 1 1
```

and the bad primes are the prime factors

$$p = 2, 3, 5, 7, 11, 13, 257, 247072949, 3288380889935506827$$

Bad primes in Gröbner basis computations

Example

```
> option("redSB");
> ring R = integer,(x, y, z),lp;
> poly f = x7y5 + x2yz9 + xz11 + y3z9;
> ideal I = groebner(ideal(diff(f, x), diff(f, y), diff(f,z)));
> apply(list(I[1..size(I)]),leadcoef);
137811552786873034477310464613260 83521912290113517241074608876444 60
12 4 12 12 45349632 12 1473863040 12 22674816 12 3888 12 12 12 13608 12
108 54 6 2 27 3 1 4 2 2 1 216 1 2 3 1 540 12 108 27 3 1 9 3 1 1 1 1 1 7
1 5 1 1
```

and the bad primes are the prime factors

$$p = 2, 3, 5, 7, 11, 13, 257, 247072949, 328838088993550682027$$

Note: The lead coefficients of the Gröbner basis over $\mathbb{Q}$ involve only the prime factors $2, 3, 5, 7, 13$.

Bad primes

Classification of bad primes:

- Type 1: Input modulo p not valid (no problem)

Bad primes

Classification of bad primes:

- Type 1: Input modulo p not valid (no problem)
- Type 2: Failure in the course of the algorithm (e.g. matrix not invertible modulo p, wastes computation time if happens)

Bad primes

Classification of bad primes:

- Type 1: Input modulo p not valid (no problem)
- Type 2: Failure in the course of the algorithm (e.g. matrix not invertible modulo p, wastes computation time if happens)
- Type 3: Computable invariant with known expected value (e.g. dimension) is wrong (have to do expensive test for each prime, although set of bad primes usually is finite)

Bad primes

Classification of bad primes:

- Type 1: Input modulo p not valid (no problem)
- Type 2: Failure in the course of the algorithm (e.g. matrix not invertible modulo p, wastes computation time if happens)
- Type 3: Computable invariant with known expected value (e.g. dimension) is wrong (have to do expensive test for each prime, although set of bad primes usually is finite)
- Type 4: Computable invariant with unknown expected value (e.g. lead ideal in Gröbner basis computations) is wrong (to detect by a majority vote, have to compute invariant for each modular result and store modular results)

Bad primes

Classification of bad primes:

- Type 1: Input modulo p not valid (no problem)
- Type 2: Failure in the course of the algorithm (e.g. matrix not invertible modulo p, wastes computation time if happens)
- Type 3: Computable invariant with known expected value (e.g. dimension) is wrong (have to do expensive test for each prime, although set of bad primes usually is finite)
- Type 4: Computable invariant with unknown expected value (e.g. lead ideal in Gröbner basis computations) is wrong (to detect by a majority vote, have to compute invariant for each modular result and store modular results)
- Type 5: otherwise.

Example of type 5 bad prime

For ideal $I \subset \mathbb{Q}[X]$ and prime p define $I_p = (I \cap \mathbb{Z}[X])_p$.

Example

Consider the algorithm $I \mapsto \sqrt{I + \mathrm{Jac}(I)}$ for

$$I = \langle x^6 + y^6 + 7x^5z + x^3y^2z - 31x^4z^2 - 224x^3z^3 + 244x^2z^4 + 1632xz^5 + 576z^6 \rangle$$

Example of type 5 bad prime

For ideal $I \subset \mathbb{Q}[X]$ and prime p define $I_p = (I \cap \mathbb{Z}[X])_p$.

Example

Consider the algorithm $I \mapsto \sqrt{I + \text{Jac}(I)}$ for

$$I = \langle x^6 + y^6 + 7x^5z + x^3y^2z - 31x^4z^2 - 224x^3z^3 + 244x^2z^4 + 1632xz^5 + 576z^6 \rangle$$

Then w.r.t dp $\qquad \text{LM}(I) = \langle x^6 \rangle = \text{LM}(I_5)$

Example of type 5 bad prime

For ideal $I \subset \mathbb{Q}[X]$ and prime p define $I_p = (I \cap \mathbb{Z}[X])_p$.

Example

Consider the algorithm $I \mapsto \sqrt{I + \mathrm{Jac}(I)}$ for

$$I = \langle x^6 + y^6 + 7x^5z + x^3y^2z - 31x^4z^2 - 224x^3z^3 + 244x^2z^4 + 1632xz^5 + 576z^6 \rangle$$

Then w.r.t dp $\qquad \mathrm{LM}(I) = \langle x^6 \rangle = \mathrm{LM}(I_5)$

$$U(0) = \sqrt{I + \mathrm{Jac}(I)} = \langle y, x - 4z \rangle \cap \langle y, x + 6z \rangle$$

$$U(5) = \sqrt{I_5 + \mathrm{Jac}(I_5)} = \langle y, x^2 - z^2 \rangle = \langle y, x - z \rangle \cap \langle y, \; x + z \rangle$$

Example of type 5 bad prime

For ideal $I \subset \mathbb{Q}[X]$ and prime p define $I_p = (I \cap \mathbb{Z}[X])_p$.

Example

Consider the algorithm $I \mapsto \sqrt{I + \mathrm{Jac}(I)}$ for

$$I = \langle x^6 + y^6 + 7x^5z + x^3y^2z - 31x^4z^2 - 224x^3z^3 + 244x^2z^4 + 1632xz^5 + 576z^6 \rangle$$

Then w.r.t dp $\qquad \mathrm{LM}(I) = \langle x^6 \rangle = \mathrm{LM}(I_5)$

$$U(0) = \sqrt{I + \mathrm{Jac}(I)} = \langle y, x - 4z \rangle \cap \langle y, x + 6z \rangle$$

$$U(5) = \sqrt{I_5 + \mathrm{Jac}(I_5)} = \langle y, x^2 - z^2 \rangle = \langle y, x - z \rangle \cap \langle y, \ x + z \rangle$$

$$U(0)_5 = \langle y, (x+z)^2 \rangle$$

Example of type 5 bad prime

For ideal $I \subset \mathbb{Q}[X]$ and prime p define $I_p = (I \cap \mathbb{Z}[X])_p$.

Example

Consider the algorithm $I \mapsto \sqrt{I + \mathrm{Jac}(I)}$ for

$$I = \langle x^6 + y^6 + 7x^5z + x^3y^2z - 31x^4z^2 - 224x^3z^3 + 244x^2z^4 + 1632xz^5 + 576z^6 \rangle$$

Then w.r.t dp $\qquad \mathrm{LM}(I) = \langle x^6 \rangle = \mathrm{LM}(I_5)$

$$U(0) = \sqrt{I + \mathrm{Jac}(I)} = \langle y, x - 4z \rangle \cap \langle y, x + 6z \rangle$$

$$U(5) = \sqrt{I_5 + \mathrm{Jac}(I_5)} = \langle y, x^2 - z^2 \rangle = \langle y, x - z \rangle \cap \langle y, \ x + z \rangle$$

$$U(0)_5 = \langle y, (x+z)^2 \rangle$$

Hence

$$U(0)_5 \neq U(5)$$

$$\mathrm{LM}(U(0)) = \langle y, x^2 \rangle = \mathrm{LM}(U(5))$$

Error tolerant reconstruction

Goal: Reconstruct $\frac{a}{b}$ from $\overline{r} \in \mathbb{Z}/N$ in the presence of bad primes.

Error tolerant reconstruction

Goal: Reconstruct $\frac{a}{b}$ from $\overline{r} \in \mathbb{Z}/N$ in the presence of bad primes.

Algorithm: Find (x, y) with $\frac{x}{y} = \frac{a}{b}$ in the lattice

$$\Lambda = \langle (N, 0), (r, 1) \rangle \subset \mathbb{Z}^2$$

Error tolerant reconstruction

Goal: Reconstruct $\frac{a}{b}$ from $\overline{r} \in \mathbb{Z}/N$ in the presence of bad primes.

Algorithm: Find (x, y) with $\frac{x}{y} = \frac{a}{b}$ in the lattice

$$\Lambda = \langle (N, 0), (r, 1) \rangle \subset \mathbb{Z}^2$$

Lemma (BDFP, 2015)

All $(x, y) \in \Lambda$ with $x^2 + y^2 < N$ are collinear.

Error tolerant reconstruction

Goal: Reconstruct $\frac{a}{b}$ from $\overline{r} \in \mathbb{Z}/N$ in the presence of bad primes.

Algorithm: Find (x, y) with $\frac{x}{y} = \frac{a}{b}$ in the lattice

$$\Lambda = \langle (N, 0), (r, 1) \rangle \subset \mathbb{Z}^2$$

Lemma (BDFP, 2015)

All $(x, y) \in \Lambda$ *with* $x^2 + y^2 < N$ *are collinear.*

Proof.

Let $\lambda = (x, y)$, $\mu = (c, d) \in \Lambda$ with $x^2 + y^2, c^2 + d^2 < N$. Then $y\mu - d\lambda = (yc - xd, 0) \in \Lambda$, so $N | (yc - xd)$. By Cauchy–Schwarz $|yc - xd| < N$, hence $yc = xd$. □

Error tolerant reconstruction

Goal: Reconstruct $\frac{a}{b}$ from $\overline{r} \in \mathbb{Z}/N$ in the presence of bad primes.

Algorithm: Find (x, y) with $\frac{x}{y} = \frac{a}{b}$ in the lattice

$$\Lambda = \langle (N, 0), (r, 1) \rangle \subset \mathbb{Z}^2$$

Lemma (BDFP, 2015)

All $(x, y) \in \Lambda$ *with* $x^2 + y^2 < N$ *are collinear.*

Proof.

Let $\lambda = (x, y)$, $\mu = (c, d) \in \Lambda$ with $x^2 + y^2, c^2 + d^2 < N$. Then $y\mu - d\lambda = (yc - xd, 0) \in \Lambda$, so $N | (yc - xd)$. By Cauchy–Schwarz $|yc - xd| < N$, hence $yc = xd$. □

Now suppose

$$N = N' \cdot M$$

with $\gcd(N', M) = 1$.

Error tolerant reconstruction

Think of N' as the product of the good primes with correct result $\overline{s}$, and of M as the product of the bad primes with wrong result $\overline{t}$.

Error tolerant reconstruction

Think of N' as the product of the good primes with correct result $\overline{s}$, and of M as the product of the bad primes with wrong result $\overline{t}$.

Theorem (BDFP, 2015)

If

$$\overline{r} \mapsto (\overline{s}, \overline{t}) \quad \textit{with respect to} \quad \mathbb{Z}/N \cong \mathbb{Z}/N' \times \mathbb{Z}/M$$

and

$$\frac{a}{b} \equiv s \bmod N'$$

Error tolerant reconstruction

Think of N' as the product of the good primes with correct result $\overline{s}$, and of M as the product of the bad primes with wrong result $\overline{t}$.

Theorem (BDFP, 2015)

If

$$\overline{r} \mapsto (\overline{s}, \overline{t}) \quad \textit{with respect to} \quad \mathbb{Z}/N \cong \mathbb{Z}/N' \times \mathbb{Z}/M$$

and

$$\frac{a}{b} \equiv s \bmod N'$$

then $(aM, bM) \in \Lambda$.

Error tolerant reconstruction

Think of N' as the product of the good primes with correct result $\overline{s}$, and of M as the product of the bad primes with wrong result $\overline{t}$.

Theorem (BDFP, 2015)

If

$$\overline{r} \mapsto (\overline{s}, \overline{t}) \quad \textit{with respect to} \quad \mathbb{Z}/N \cong \mathbb{Z}/N' \times \mathbb{Z}/M$$

and

$$\frac{a}{b} \equiv s \bmod N'$$

then $(aM, bM) \in \Lambda$. *So if*

$$(a^2 + b^2)M < N',$$

then (by the lemma)

$$\frac{x}{y} = \frac{a}{b} \quad \textit{for all } (x, y) \in \Lambda \textit{ with } (x^2 + y^2) < N$$

and such vectors exist.

Error tolerant reconstruction

Think of N' as the product of the good primes with correct result $\overline{s}$, and of M as the product of the bad primes with wrong result $\overline{t}$.

Theorem (BDFP, 2015)

If

$$\overline{r} \mapsto (\overline{s}, \overline{t}) \quad \textit{with respect to} \quad \mathbb{Z}/N \cong \mathbb{Z}/N' \times \mathbb{Z}/M$$

and

$$\frac{a}{b} \equiv s \bmod N'$$

then $(aM, bM) \in \Lambda$. *So if*

$$(a^2 + b^2)M < N',$$

then (by the lemma)

$$\frac{x}{y} = \frac{a}{b} \quad \textit{for all } (x, y) \in \Lambda \textit{ with } (x^2 + y^2) < N$$

and such vectors exist. Moreover, if $\gcd(a, b) = 1$ *and* (x, y) *is a shortest vector* $\neq 0$ *in* Λ, *we also have* $\gcd(x, y) | M$.

Error tolerant reconstruction via Gauss-Lagrange

Hence, if $N' \gg M$, the Gauss-Lagrange-Algorithm for finding a shortest vector $(x, y) \in \Lambda$ gives $\frac{a}{b}$ independently of t, provided $x^2 + y^2 < N$.

Error tolerant reconstruction via Gauss-Lagrange

Hence, if $N' \gg M$, the Gauss-Lagrange-Algorithm for finding a shortest vector $(x, y) \in \Lambda$ gives $\frac{a}{b}$ independently of t, provided $x^2 + y^2 < N$.

Algorithm (Error tolerant reconstruction)

```
function ErrorTolerantReconstruction(r::Integer, N::Integer)
a1 = [N, 0]
a2 = [r, 1]
while dot(a1, a1) > dot(a2, a2)
q = dot(a1, a2)//dot(a2, a2)
a1, a2 = a2, a1 - Integer(round(q))*a2
end
if dot(a1, a1) < N
return a1[1]//a1[2]
else
return false
end
end
```

Error tolerant reconstruction via Gauss-Lagrange

Hence, if $N' \gg M$, the Gauss-Lagrange-Algorithm for finding a shortest vector $(x, y) \in \Lambda$ gives $\frac{a}{b}$ independently of t, provided $x^2 + y^2 < N$.

Algorithm (Error tolerant reconstruction)

```
function ErrorTolerantReconstruction(r::Integer, N::Integer)
a1 = [N, 0]
a2 = [r, 1]
while dot(a1, a1) > dot(a2, a2)
q = dot(a1, a2)//dot(a2, a2)
a1, a2 = a2, a1 - Integer(round(q))*a2
end
if dot(a1, a1) < N
return a1[1]//a1[2]
else
return false
end
end
```

Singular-kernel	Julia	Singular-interpreter
0.001	0.005	0.055

(in seconds, bitlength 500)

Reconstruction via Gauss-Lagrange

Example

We reconstruct $\frac{13}{12}$ from

$$\overline{22684} \in \mathbb{Z}/38885$$

by determining a shortest vector in the lattice

$$\langle(38885, 0), (22684, 1)\rangle \subset \mathbb{Z}^2$$

via Gauss-Lagrange

Reconstruction via Gauss-Lagrange

Example

We reconstruct $\frac{13}{12}$ from

$$\overline{22684} \in \mathbb{Z}/38885$$

by determining a shortest vector in the lattice

$$\langle (38885, 0), (22684, 1) \rangle \subset \mathbb{Z}^2$$

via Gauss-Lagrange

$$\begin{aligned}
(38885, 0) &= 2 \cdot (22684, 1) + (-6483, -2), \\
(22684, 1) &= -3 \cdot (-6483, -2) + (3235, -5), \\
(-6483, -2) &= 2 \cdot (3235, -5) + (-13, -12), \\
(3235, -5) &= -134 \cdot (-13, -12) + (1493, -1613).
\end{aligned}$$

Example

Now introduce an error in the modular results:

$$\begin{array}{ccccccccc} \mathbb{Z}/5 & \times & \mathbb{Z}/7 & \times & \mathbb{Z}/11 & \times & \mathbb{Z}/101 & \cong & \mathbb{Z}/38885 \\ (\ \overline{4} & , & \overline{4} & , & \overline{2} & , & \overline{60}\ ) & \mapsto & \overline{22684} \end{array}$$

Example

Now introduce an error in the modular results:

$$\begin{array}{ccccccccccc} \mathbb{Z}/5 & \times & \mathbb{Z}/7 & \times & \mathbb{Z}/11 & \times & \mathbb{Z}/101 & \cong & \mathbb{Z}/38885 \\ (\,\overline{4} & , & \overline{4} & , & \overline{2} & , & \overline{60}\,) & \mapsto & \overline{22684} \\ (\,\overline{4} & , & {\color{red}\overline{2}} & , & \overline{2} & & \overline{60}\,) & \mapsto & \overline{464} \end{array}$$

Example

Now introduce an error in the modular results:

$$\begin{array}{ccccccccccc}
\mathbb{Z}/5 & \times & \mathbb{Z}/7 & \times & \mathbb{Z}/11 & \times & \mathbb{Z}/101 & & \cong & \mathbb{Z}/38885 \\
(\ \overline{4} & , & \overline{4} & , & \overline{2} & , & \overline{60} &) & \mapsto & \overline{22684} \\
(\ \overline{4} & , & \color{red}{\overline{2}} & , & \overline{2} & & \overline{60} &) & \mapsto & \overline{464}
\end{array}$$

Error tolerant reconstruction computes

$$(38885, 0) = 84 \cdot (464, 1) + (-91, -84),$$
$$(464, 1) = -3 \cdot (-91, -84) + (191, -251)$$

Example

Now introduce an error in the modular results:

$$\begin{array}{ccccccccccc} \mathbb{Z}/5 & \times & \mathbb{Z}/7 & \times & \mathbb{Z}/11 & \times & \mathbb{Z}/101 & \cong & \mathbb{Z}/38885 \\ (\,\overline{4} & , & \overline{4} & , & \overline{2} & , & \overline{60}\,) & \mapsto & \overline{22684} \\ (\,\overline{4} & , & \color{red}{\overline{2}} & , & \overline{2} & & \overline{60}\,) & \mapsto & \overline{464} \end{array}$$

Error tolerant reconstruction computes

$$\begin{aligned} (38885, 0) &= 84 \cdot (464, 1) + (-91, -84), \\ (464, 1) &= -3 \cdot (-91, -84) + (191, -251) \end{aligned}$$

hence yields

$$\frac{91}{84} = \frac{7 \cdot 13}{7 \cdot 12} = \frac{13}{12}.$$

Reconstruction via Gauss-Lagrange

Example

Now introduce an error in the modular results:

$$\begin{array}{ccccccccccc} \mathbb{Z}/5 & \times & \mathbb{Z}/7 & \times & \mathbb{Z}/11 & \times & \mathbb{Z}/101 & \cong & \mathbb{Z}/38885 \\ (\ \overline{4} & , & \overline{4} & , & \overline{2} & , & \overline{60}\ ) & \mapsto & \overline{22684} \\ (\ \overline{4} & , & \color{red}{\overline{2}} & , & \overline{2} & & \overline{60}\ ) & \mapsto & \overline{464} \end{array}$$

Error tolerant reconstruction computes

$$\begin{aligned} (38885, 0) &= 84 \cdot (464, 1) + (-91, -84), \\ (464, 1) &= -3 \cdot (-91, -84) + (191, -251) \end{aligned}$$

hence yields

$$\frac{91}{84} = \frac{7 \cdot 13}{7 \cdot 12} = \frac{13}{12}.$$

Note that

$$(13^2 + 12^2) \cdot 7 = 2191 < 5555 = 5 \cdot 11 \cdot 101.$$

General reconstruction scheme

Setup: For ideal $I \subset \mathbb{Q}[X]$ compute ideal (or module) $U(0)$ associated to I by deterministic algorithm.

Algorithm

- *For I_p compute result $U(p)$ over $\mathbb{Z}/p$ for p in finite set of primes $\mathcal{P}$.*

General reconstruction scheme

Setup: For ideal $I \subset \mathbb{Q}[X]$ compute ideal (or module) $U(0)$ associated to I by deterministic algorithm.

Algorithm

- *For I_p compute result $U(p)$ over $\mathbb{Z}/p$ for p in finite set of primes $\mathcal{P}$.*
- *Reduce $\mathcal{P}$ according to majority vote on $\mathrm{LM}(U(p))$.*

General reconstruction scheme

Setup: For ideal $I \subset \mathbb{Q}[X]$ compute ideal (or module) $U(0)$ associated to I by deterministic algorithm.

Algorithm

- *For I_p compute result $U(p)$ over $\mathbb{Z}/p$ for p in finite set of primes $\mathcal{P}$.*
- *Reduce $\mathcal{P}$ according to majority vote on $\mathrm{LM}(U(p))$.*
- *For $N = \prod_{p \in \mathcal{P}} p$ compute termwise CRT–lift $U(N)$ to $\mathbb{Z}/N$.*

General reconstruction scheme

Setup: For ideal $I \subset \mathbb{Q}[X]$ compute ideal (or module) $U(0)$ associated to I by deterministic algorithm.

Algorithm

- *For I_p compute result $U(p)$ over $\mathbb{Z}/p$ for p in finite set of primes $\mathcal{P}$.*
- *Reduce $\mathcal{P}$ according to majority vote on $\mathrm{LM}(U(p))$.*
- *For $N = \prod_{p\in\mathcal{P}} p$ compute termwise CRT–lift $U(N)$ to $\mathbb{Z}/N$.*
- *Lift $U(N)$ by error tolerant rational reconstruction to U.*

General reconstruction scheme

Setup: For ideal $I \subset \mathbb{Q}[X]$ compute ideal (or module) $U(0)$ associated to I by deterministic algorithm.

Algorithm

- *For I_p compute result $U(p)$ over $\mathbb{Z}/p$ for p in finite set of primes $\mathcal{P}$.*
- *Reduce $\mathcal{P}$ according to majority vote on $\mathrm{LM}(U(p))$.*
- *For $N = \prod_{p\in\mathcal{P}} p$ compute termwise CRT–lift $U(N)$ to $\mathbb{Z}/N$.*
- *Lift $U(N)$ by error tolerant rational reconstruction to U.*
- *Test $U_p = U(p)$ for random prime p.*

General reconstruction scheme

Setup: For ideal $I \subset \mathbb{Q}[X]$ compute ideal (or module) $U(0)$ associated to I by deterministic algorithm.

Algorithm

- *For I_p compute result $U(p)$ over $\mathbb{Z}/p$ for p in finite set of primes $\mathcal{P}$.*
- *Reduce $\mathcal{P}$ according to majority vote on $\mathrm{LM}(U(p))$.*
- *For $N = \prod_{p\in\mathcal{P}} p$ compute termwise CRT–lift $U(N)$ to $\mathbb{Z}/N$.*
- *Lift $U(N)$ by error tolerant rational reconstruction to U.*
- *Test $U_p = U(p)$ for random prime p.*
- *Verify $U = U(0)$.*

General reconstruction scheme

Setup: For ideal $I \subset \mathbb{Q}[X]$ compute ideal (or module) $U(0)$ associated to I by deterministic algorithm.

Algorithm

- *For I_p compute result $U(p)$ over $\mathbb{Z}/p$ for p in finite set of primes $\mathcal{P}$.*
- *Reduce $\mathcal{P}$ according to majority vote on* $\mathrm{LM}(U(p))$.
- *For $N = \prod_{p \in \mathcal{P}} p$ compute termwise CRT–lift $U(N)$ to $\mathbb{Z}/N$.*
- *Lift $U(N)$ by error tolerant rational reconstruction to U.*
- *Test $U_p = U(p)$ for random prime p.*
- *Verify $U = U(0)$.*
- *If lift, test or verification fails, then enlarge $\mathcal{P}$.*

General reconstruction scheme

Setup: For ideal $I \subset \mathbb{Q}[X]$ compute ideal (or module) $U(0)$ associated to I by deterministic algorithm.

Algorithm

- *For I_p compute result $U(p)$ over $\mathbb{Z}/p$ for p in finite set of primes $\mathcal{P}$.*
- *Reduce $\mathcal{P}$ according to majority vote on $\mathrm{LM}(U(p))$.*
- *For $N = \prod_{p \in \mathcal{P}} p$ compute termwise CRT–lift $U(N)$ to $\mathbb{Z}/N$.*
- *Lift $U(N)$ by error tolerant rational reconstruction to U.*
- *Test $U_p = U(p)$ for random prime p.*
- *Verify $U = U(0)$.*
- *If lift, test or verification fails, then enlarge $\mathcal{P}$.*

Theorem (BDFP, 2015)

If the set of bad primes for computing $U(0)$ from I is finite, then this algorithm terminates with the correct result.

Timings in Singular for Adjoint Ideal

Plane curve f_n of degree n with $\binom{n-1}{2}$ singularities of type A_1.

Timings in SINGULAR for Adjoint Ideal

Plane curve f_n of degree n with $\binom{n-1}{2}$ singularities of type A_1.

	parallel	probablisitic	f_5		f_6		f_7	
`locNormal`			2.1		56		-	
`Maple-IB`			5.1		47		318	
`LA`			98		4400		-	
`IQ`			1.3		54		3800	
`locIQ`	■		1.3	(1)	54	(1)	3800	(1)
`ADE`	■		.18	(1)	1.2	(1)	49	(1)
`modLocIQ`			6.4	[33]	19	[53]	150	[75]
		■	6.2	[33]	18	[53]	104	[75]
	■		.36	(74)	1.6	(153)	51	(230)
	■	■	.21	(74)	0.48	(153)	5.2	(230)

[primes] (cores)

References

- J. Boehm, W. Decker, C. Fieker, G. Pfister. *The use of bad primes in rational reconstruction*, Math. Comp. 84 (2015).
- J. Boehm, W. Decker, S. Laplagne, G. Pfister, A. Steenpaß, S. Steidel. *Parallel algorithms for normalization*, J. Symb. Comp. 51 (2013).
- J. Boehm, W. Decker, G. Pfister, S. Laplagne. *Local to global algorithms for the Gorenstein adjoint ideal of a curve*, arXiv:1505.05040.

- P. Kornerup, R. T. Gregory, *Mapping integers and Hensel codes onto Farey fractions*, BIT 23 (1983).
- E. Arnold, *Modular algorithms for computing Gröbner bases*, J. Symb. Comp. 35 (2003).
- G.-M. Greuel, S. Laplagne, S. Seelisch, *Normalization of rings*, J. Symb. Comp. (2010).

Singular and GPI-Space

Features of GPI-Space developed by Fraunhofer Institute for Industrial Mathematics ITWM, Kaiserslautern:

- Distributed run-time system suitable for massively parallel computations.

Singular and GPI-Space

Features of GPI-Space developed by Fraunhofer Institute for Industrial Mathematics ITWM, Kaiserslautern:

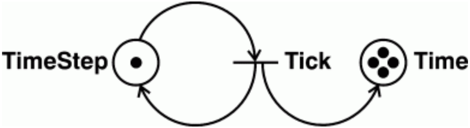
- Distributed run-time system suitable for massively parallel computations.
- Virtual memory layer.

Features of GPI-Space developed by Fraunhofer Institute for Industrial Mathematics ITWM, Kaiserslautern:

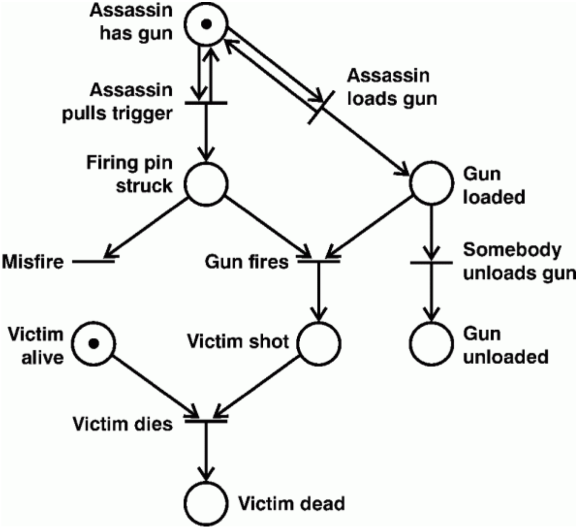
- Distributed run-time system suitable for massively parallel computations.
- Virtual memory layer.
- Modeling with Petri nets.

GPI-Space: A Petri net

Clock at time $t = 4$:



GPI-Space: A Petri net



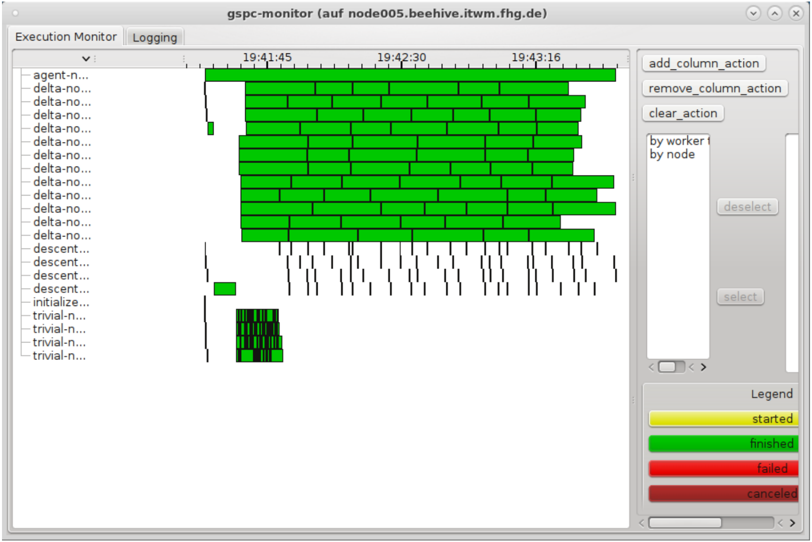
Features of GPI-Space developed by Fraunhofer Institute for Industrial Mathematics ITWM, Kaiserslautern:

- Distributed run-time system suitable for massively parallel computations.
- Virtual memory layer.
- Modeling with Petri nets.

Features of GPI-Space developed by Fraunhofer Institute for Industrial Mathematics ITWM, Kaiserslautern:

- Distributed run-time system suitable for massively parallel computations.
- Virtual memory layer.
- Modeling with Petri nets.
- Auto-parallelization engine.

GPI-Space: Scheduler



Singular and GPI-Space

Features of GPI-Space developed by Fraunhofer Institute for Industrial Mathematics ITWM, Kaiserslautern:

- Distributed run-time system suitable for massively parallel computations.
- Virtual memory layer.
- Modeling with Petri nets.
- Auto-parallelization engine.

Features of GPI-Space developed by Fraunhofer Institute for Industrial Mathematics ITWM, Kaiserslautern:

- Distributed run-time system suitable for massively parallel computations.
- Virtual memory layer.
- Modeling with Petri nets.
- Auto-parallelization engine.

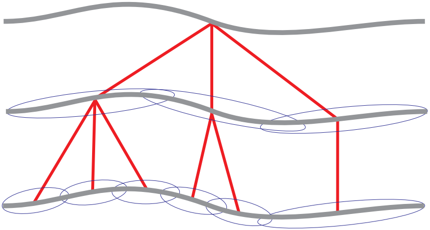
Integration of SINGULAR in GPI-Space.
Cluster at ITWM with $\approx 10^4$ nodes.

Example

Algorithm for determining smoothness by local descent in codimension relative to a smooth complete intersection (as in Hironaka's resolution of singularities). Descent to any desired size of minors in Jacobian criterion.

Example

Algorithm for determining smoothness by local descent in codimension relative to a smooth complete intersection (as in Hironaka's resolution of singularities). Descent to any desired size of minors in Jacobian criterion.



Boehm, J., Frühbis-Krüger: *A smoothness test for higher codimensions.* arXiv:1603.09241 JSC (to appear).

Good Quotients

The goal of Geometric Invariant Theory (GIT) is to assign to a given algebraic variety X with action of a reductive group G a reasonable quotient space $X /\!/ G$.

Good Quotients

The goal of Geometric Invariant Theory (GIT) is to assign to a given algebraic variety X with action of a reductive group G a reasonable quotient space $X /\!/ G$. Two main problems:

Good Quotients

The goal of Geometric Invariant Theory (GIT) is to assign to a given algebraic variety X with action of a reductive group G a reasonable quotient space $X /\!/ G$. Two main problems:

1. The orbit space X / G is not a good candidate for $X /\!/ G$:

$$\mathbb{C}^* \times \mathbb{C} \to \mathbb{C}, \qquad t \cdot x = tx$$

Good Quotients

The goal of Geometric Invariant Theory (GIT) is to assign to a given algebraic variety X with action of a reductive group G a reasonable quotient space $X /\!/ G$. Two main problems:

1. The orbit space X / G is not a good candidate for $X /\!/ G$:

$$\mathbb{C}^* \times \mathbb{C} \to \mathbb{C}, \qquad t \cdot x = tx$$

Instead, for X affine define

$$X /\!/ G = \operatorname{Spec} K[X]^G$$

as the spectrum of the (finitely generated) invariant ring of the functions on X. For general X, glue the quotients of an affine covering.

The goal of Geometric Invariant Theory (GIT) is to assign to a given algebraic variety X with action of a reductive group G a reasonable quotient space $X /\!/ G$. Two main problems:

1. The orbit space X/G is not a good candidate for $X /\!/ G$:

$$\mathbb{C}^* \times \mathbb{C} \to \mathbb{C}, \qquad t \cdot x = tx$$

   Instead, for X affine define

$$X /\!/ G = \operatorname{Spec} K[X]^G$$

   as the spectrum of the (finitely generated) invariant ring of the functions on X. For general X, glue the quotients of an affine covering.

2. The quotient $X /\!/ G$ may not carry much information.

Good Quotients

The goal of Geometric Invariant Theory (GIT) is to assign to a given algebraic variety X with action of a reductive group G a reasonable quotient space $X /\!/ G$. Two main problems:

1. The orbit space X/G is not a good candidate for $X /\!/ G$:

$$\mathbb{C}^* \times \mathbb{C} \to \mathbb{C}, \qquad t \cdot x = tx$$

   Instead, for X affine define

$$X /\!/ G = \operatorname{Spec} K[X]^G$$

   as the spectrum of the (finitely generated) invariant ring of the functions on X. For general X, glue the quotients of an affine covering.

2. The quotient $X /\!/ G$ may not carry much information.

   Hence pass to open subset $U \subset X$.

Good Quotients

Example

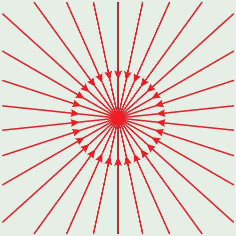
$$\mathbb{C}^* \times \mathbb{C}^2 \to \mathbb{C}^2, \qquad t \cdot (x, y) = (tx, ty)$$

Good Quotients

Example

$$\mathbb{C}^* \times \mathbb{C}^2 \to \mathbb{C}^2, \qquad t \cdot (x, y) = (tx, ty)$$

$$U = \mathbb{C}^2$$

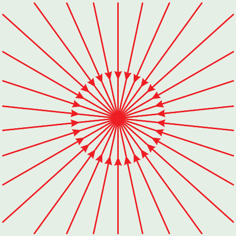


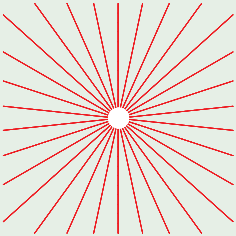
$$U /\!/ \mathbb{C}^* = \{pt\}$$

Example

$$\mathbb{C}^* \times \mathbb{C}^2 \to \mathbb{C}^2, \qquad t \cdot (x, y) = (tx, ty)$$

$U = \mathbb{C}^2$ | $U = \mathbb{C}^2 \setminus \{0\}$





$U//\mathbb{C}^* = \{pt\}$ | $U//\mathbb{C}^* = \mathbb{P}^1$

GIT-Fan

In general, there are many choices for these open subsets $U \subset X$ leading to different quotients. To describe this behaviour, Dolgachev and Hu introduced the **GIT-fan**, a polyhedral fan describing the variation of GIT-quotients.

GIT-Fan

In general, there are many choices for these open subsets $U \subset X$ leading to different quotients. To describe this behaviour, Dolgachev and Hu introduced the **GIT-fan**, a polyhedral fan describing the variation of GIT-quotients. We focus on the action of an algebraic torus $G = (\mathbb{C}^*)^k$ on an affine variety $X \subset \mathbb{C}^r$.

GIT-Fan

In general, there are many choices for these open subsets $U \subset X$ leading to different quotients. To describe this behaviour, Dolgachev and Hu introduced the **GIT-fan**, a polyhedral fan describing the variation of GIT-quotients. We focus on the action of an algebraic torus $G = (\mathbb{C}^*)^k$ on an affine variety $X \subset \mathbb{C}^r$.

Setup:

- ideal $\mathfrak{a} \subset \mathbb{C}[T_1, \ldots, T_r]$ defining X,
- matrix $Q = (q_1, \ldots, q_r) \in \mathbb{Z}^{k \times r}$ such that $\mathfrak{a}$ is homogeneous w.r.t. grading $\deg(T_i) = q_i \in \mathbb{Z}^k$.

GIT-Fan

In general, there are many choices for these open subsets $U \subset X$ leading to different quotients. To describe this behaviour, Dolgachev and Hu introduced the **GIT-fan**, a polyhedral fan describing the variation of GIT-quotients. We focus on the action of an algebraic torus $G = (\mathbb{C}^*)^k$ on an affine variety $X \subset \mathbb{C}^r$.

Setup:

- ideal $\mathfrak{a} \subset \mathbb{C}[T_1, \ldots, T_r]$ defining X,
- matrix $Q = (q_1, \ldots, q_r) \in \mathbb{Z}^{k \times r}$ such that $\mathfrak{a}$ is homogeneous w.r.t. grading $\deg(T_i) = q_i \in \mathbb{Z}^k$.

Example

For $\mathbb{C}^* \times \mathbb{C}^2 \to \mathbb{C}^2, \quad t \cdot (x, y) = (tx, ty)$

$$U_1 = \mathbb{C}^2 \qquad \Lambda(\langle 0 \rangle, (1,1)) =$$
$$U_2 = \mathbb{C}^2 \backslash \{0\}$$

Computing GIT-Fans

In case of a torus acting on an affine variety, Berchthold/Hausen and Keicher have developed a method for computing the GIT-fan.

In case of a torus acting on an affine variety, Berchthold/Hausen and Keicher have developed a method for computing the GIT-fan.
Decomposition into torus orbits corresponding to faces $\gamma \prec \mathbb{Q}^r_{\geq 0}$:

$$\mathbb{C}^r = \bigcup_\gamma O(\gamma)$$

$$O(\gamma) = (\mathbb{C}^*)^r \cdot \sum_{e_i \in \gamma} e_i = \{(z_1, \ldots, z_r) \in \mathbb{C}^r \mid z_i \neq 0 \Leftrightarrow e_i \in \gamma\}$$

Computing GIT-Fans

In case of a torus acting on an affine variety, Berchthold/Hausen and Keicher have developed a method for computing the GIT-fan.

Decomposition into torus orbits corresponding to faces $\gamma \prec \mathbb{Q}^r_{\geq 0}$:

$$\mathbb{C}^r = \bigcup_\gamma O(\gamma)$$

$$O(\gamma) = (\mathbb{C}^*)^r \cdot \sum_{e_i \in \gamma} e_i = \{(z_1, \ldots, z_r) \in \mathbb{C}^r \mid z_i \neq 0 \Leftrightarrow e_i \in \gamma\}$$

Proposition

*Face $\gamma \prec \mathbb{Q}^r_{\geq 0}$ is called an $\mathfrak{a}$-**face** if the following equivalent conditions are satisfied:*

Computing GIT-Fans

In case of a torus acting on an affine variety, Berchthold/Hausen and Keicher have developed a method for computing the GIT-fan.

Decomposition into torus orbits corresponding to faces $\gamma \prec \mathbb{Q}^r_{\geq 0}$:

$$\mathbb{C}^r = \bigcup_\gamma O(\gamma)$$

$$O(\gamma) = (\mathbb{C}^*)^r \cdot \sum_{e_i \in \gamma} e_i = \{(z_1, \ldots, z_r) \in \mathbb{C}^r \mid z_i \neq 0 \Leftrightarrow e_i \in \gamma\}$$

Proposition

*Face $\gamma \prec \mathbb{Q}^r_{\geq 0}$ is called an $\mathfrak{a}$-**face** if the following equivalent conditions are satisfied:*

1. $X \cap O(\gamma) \neq \emptyset$

Computing GIT-Fans

In case of a torus acting on an affine variety, Berchthold/Hausen and Keicher have developed a method for computing the GIT-fan.

Decomposition into torus orbits corresponding to faces $\gamma \prec \mathbb{Q}^r_{\geq 0}$:

$$\mathbb{C}^r = \bigcup_\gamma O(\gamma)$$

$$O(\gamma) = (\mathbb{C}^*)^r \cdot \sum_{e_i \in \gamma} e_i = \{(z_1, \ldots, z_r) \in \mathbb{C}^r \mid z_i \neq 0 \Leftrightarrow e_i \in \gamma\}$$

Proposition

Face $\gamma \prec \mathbb{Q}^r_{\geq 0}$ is called an $\mathfrak{a}$**-face** *if the following equivalent conditions are satisfied:*

1. $X \cap O(\gamma) \neq \emptyset$
2. *There is* $x \in X$ *with* $x_i \neq 0 \Leftrightarrow e_i \in \gamma$
3. $\left(\mathfrak{a}|_{T_i = 0 \text{ for } e_i \notin \gamma}\right) : \langle T_1 \cdots T_r \rangle^\infty \neq \langle 1 \rangle$

In case of a torus acting on an affine variety, Berchthold/Hausen and Keicher have developed a method for computing the GIT-fan.
Decomposition into torus orbits corresponding to faces $\gamma \prec \mathbb{Q}^r_{\geq 0}$:

$$\mathbb{C}^r = \bigcup_\gamma O(\gamma)$$

$$O(\gamma) = (\mathbb{C}^*)^r \cdot \sum_{e_i \in \gamma} e_i = \{(z_1, \ldots, z_r) \in \mathbb{C}^r \mid z_i \neq 0 \Leftrightarrow e_i \in \gamma\}$$

Proposition

Face $\gamma \prec \mathbb{Q}^r_{\geq 0}$ is called an $\mathfrak{a}$-**face** *if the following equivalent conditions are satisfied:*

1. $X \cap O(\gamma) \neq \emptyset$
2. *There is* $x \in X$ *with* $x_i \neq 0 \Leftrightarrow e_i \in \gamma$
3. $\left(\mathfrak{a}|_{T_i = 0 \text{ for } e_i \notin \gamma}\right) : \langle T_1 \cdots T_r \rangle^\infty \neq \langle 1 \rangle$

The **orbit cones** are the $Q(\gamma) = \operatorname{cone}(q_i \mid e_i \in \gamma)$ with γ an $\mathfrak{a}$-face.

1. Determine $\mathfrak{a}$-faces.

1. Determine $\mathfrak{a}$-faces.
2. Compute set of orbit cones

$$\Omega = \{Q(\gamma) \mid \gamma \text{ an } \mathfrak{a}\text{-face}\}$$

where

$$Q(\gamma) = \mathrm{cone}(q_i \mid e_i \in \gamma) \subset \Gamma = Q(\mathbb{Q}^r_{\geq 0}) = \mathrm{cone}(q_1, \ldots, q_r) \subset \mathbb{Q}^k$$

is projection of γ with respect to Q.

Computing GIT-Fans

1. Determine $\mathfrak{a}$-faces.
2. Compute set of orbit cones

$$\Omega = \{Q(\gamma) \mid \gamma \text{ an } \mathfrak{a}\text{-face}\}$$

where

$$Q(\gamma) = \text{cone}(q_i \mid e_i \in \gamma) \subset \Gamma = Q(\mathbb{Q}^r_{\geq 0}) = \text{cone}(q_1, \ldots, q_r) \subset \mathbb{Q}^k$$

is projection of γ with respect to Q.

3. Determine GIT-fan:

$$\Lambda(\mathfrak{a}, Q) = \{\lambda_\Omega(w) \mid w \in \Gamma\} \quad \text{where} \quad \lambda_\Omega(w) = \bigcap_{w \in \eta \in \Omega} \eta$$

Algorithm

Input: *Ideal* $\mathfrak{a} \subset \mathbb{C}[T_1, \ldots, T_r]$ *and matrix* $Q \in \mathbb{Z}^{k \times r}$ *of full rank such that* $\mathfrak{a}$ *is homogeneous w.r.t. multigrading by* Q.

Output: *The set of maximal cones of* $\Lambda(\mathfrak{a}, Q)$.

Algorithm

Input: *Ideal* $\mathfrak{a} \subset \mathbb{C}[T_1, \ldots, T_r]$ *and matrix* $Q \in \mathbb{Z}^{k \times r}$ *of full rank such that* $\mathfrak{a}$ *is homogeneous w.r.t. multigrading by* Q.

Output: *The set of maximal cones of* $\Lambda(\mathfrak{a}, Q)$.

1: $\mathcal{A} := \{\gamma \prec \mathbb{Q}^r_{\geq 0} \mid \gamma \text{ is an } \mathfrak{a}\text{-face}\}$

GIT-Algorithm

Algorithm

Input: *Ideal* $\mathfrak{a} \subset \mathbb{C}[T_1, \ldots, T_r]$ *and matrix* $Q \in \mathbb{Z}^{k \times r}$ *of full rank such that* $\mathfrak{a}$ *is homogeneous w.r.t. multigrading by* Q.

Output: *The set of maximal cones of* $\Lambda(\mathfrak{a}, Q)$.

1: $\mathcal{A} := \{\gamma \prec \mathbb{Q}^r_{\geq 0} \mid \gamma \text{ is an } \mathfrak{a}\text{-face}\}$
2: $\Omega := \{Q(\gamma) \mid \gamma \in \mathcal{A}\}$

GIT-Algorithm

Algorithm

Input: *Ideal* $\mathfrak{a} \subset \mathbb{C}[T_1, \ldots, T_r]$ *and matrix* $Q \in \mathbb{Z}^{k \times r}$ *of full rank such that* $\mathfrak{a}$ *is homogeneous w.r.t. multigrading by* Q.

Output: *The set of maximal cones of* $\Lambda(\mathfrak{a}, Q)$.

1: $\mathcal{A} := \{\gamma \prec \mathbb{Q}^r_{\geq 0} \mid \gamma$ *is an* $\mathfrak{a}$*-face*$\}$

2: $\Omega := \{Q(\gamma) \mid \gamma \in \mathcal{A}\}$

3: *Choose a vector* $w_0 \in Q(\mathbb{Q}^r_{\geq 0})$ *such that* $\dim(\lambda_\Omega(w_0)) = k$.

Algorithm

Input: *Ideal* $\mathfrak{a} \subset \mathbb{C}[T_1, \ldots, T_r]$ *and matrix* $Q \in \mathbb{Z}^{k \times r}$ *of full rank such that* $\mathfrak{a}$ *is homogeneous w.r.t. multigrading by* Q.

Output: *The set of maximal cones of* $\Lambda(\mathfrak{a}, Q)$.

1: $\mathcal{A} := \{\gamma \prec \mathbb{Q}^r_{\geq 0} \mid \gamma \text{ is an } \mathfrak{a}\text{-face}\}$

2: $\Omega := \{Q(\gamma) \mid \gamma \in \mathcal{A}\}$

3: *Choose a vector* $w_0 \in Q(\mathbb{Q}^r_{\geq 0})$ *such that* $\dim(\lambda_\Omega(w_0)) = k$.

4: $\mathcal{C} := \{\lambda_\Omega(w_0)\}$

Algorithm

Input: *Ideal* $\mathfrak{a} \subset \mathbb{C}[T_1, \ldots, T_r]$ *and matrix* $Q \in \mathbb{Z}^{k \times r}$ *of full rank such that* $\mathfrak{a}$ *is homogeneous w.r.t. multigrading by* Q.

Output: *The set of maximal cones of* $\Lambda(\mathfrak{a}, Q)$.

1: $\mathcal{A} := \{\gamma \prec \mathbb{Q}^r_{\geq 0} \mid \gamma$ *is an* $\mathfrak{a}$*-face*$\}$
2: $\Omega := \{Q(\gamma) \mid \gamma \in \mathcal{A}\}$
3: *Choose a vector* $w_0 \in Q(\mathbb{Q}^r_{\geq 0})$ *such that* $\dim(\lambda_\Omega(w_0)) = k$.
4: $\mathcal{C} := \{\lambda_\Omega(w_0)\}$
5: $\mathcal{F} := \{(\eta, \lambda_\Omega(w_0)) \mid \eta \prec \lambda_\Omega(w_0)$ *interior facet*$\}$.

GIT-Algorithm

Algorithm

Input: *Ideal* $\mathfrak{a} \subset \mathbb{C}[T_1, \ldots, T_r]$ *and matrix* $Q \in \mathbb{Z}^{k \times r}$ *of full rank such that* $\mathfrak{a}$ *is homogeneous w.r.t. multigrading by* Q.

Output: *The set of maximal cones of* $\Lambda(\mathfrak{a}, Q)$.

$\mathcal{A} := \{\gamma \prec \mathbb{Q}^r_{\geq 0} \mid \gamma \text{ is an } \mathfrak{a}\text{-face}\}$

$\Omega := \{Q(\gamma) \mid \gamma \in \mathcal{A}\}$

Choose a vector $w_0 \in Q(\mathbb{Q}^r_{\geq 0})$ *such that* $\dim(\lambda_\Omega(w_0)) = k$.

$\mathcal{C} := \{\lambda_\Omega(w_0)\}$

$\mathcal{F} := \{(\eta, \lambda_\Omega(w_0)) \mid \eta \prec \lambda_\Omega(w_0) \text{ interior facet}\}$.

while *there is* $(\eta, \lambda) \in \mathcal{F}$ **do**

GIT-Algorithm

Algorithm

Input: *Ideal $\mathfrak{a} \subset \mathbb{C}[T_1, \dots, T_r]$ and matrix $Q \in \mathbb{Z}^{k \times r}$ of full rank such that $\mathfrak{a}$ is homogeneous w.r.t. multigrading by Q.*

Output: *The set of maximal cones of $\Lambda(\mathfrak{a}, Q)$.*

$\mathcal{A} := \{\gamma \prec \mathbb{Q}^r_{\geq 0} \mid \gamma \text{ is an } \mathfrak{a}\text{-face}\}$
$\Omega := \{Q(\gamma) \mid \gamma \in \mathcal{A}\}$
Choose a vector $w_0 \in Q(\mathbb{Q}^r_{\geq 0})$ such that $\dim(\lambda_\Omega(w_0)) = k$.
$\mathcal{C} := \{\lambda_\Omega(w_0)\}$
$\mathcal{F} := \{(\eta, \lambda_\Omega(w_0)) \mid \eta \prec \lambda_\Omega(w_0) \text{ interior facet}\}$.
while *there is $(\eta, \lambda) \in \mathcal{F}$* **do**
    Find $w \in Q(\gamma)$ such that $w \notin \lambda$ and $\lambda_\Omega(w) \cap \lambda = \eta$.

GIT-Algorithm

Algorithm

Input: *Ideal* $\mathfrak{a} \subset \mathbb{C}[T_1, \dots, T_r]$ *and matrix* $Q \in \mathbb{Z}^{k \times r}$ *of full rank such that* $\mathfrak{a}$ *is homogeneous w.r.t. multigrading by* Q.

Output: *The set of maximal cones of* $\Lambda(\mathfrak{a}, Q)$.

1: $\mathcal{A} := \{\gamma \prec \mathbb{Q}^r_{\geq 0} \mid \gamma$ *is an* $\mathfrak{a}$*-face*$\}$
2: $\Omega := \{Q(\gamma) \mid \gamma \in \mathcal{A}\}$
3: *Choose a vector* $w_0 \in Q(\mathbb{Q}^r_{\geq 0})$ *such that* $\dim(\lambda_\Omega(w_0)) = k$.
4: $\mathcal{C} := \{\lambda_\Omega(w_0)\}$
5: $\mathcal{F} := \{(\eta, \lambda_\Omega(w_0)) \mid \eta \prec \lambda_\Omega(w_0)$ *interior facet*$\}$.
6: **while** *there is* $(\eta, \lambda) \in \mathcal{F}$ **do**
7: *Find* $w \in Q(\gamma)$ *such that* $w \notin \lambda$ *and* $\lambda_\Omega(w) \cap \lambda = \eta$.
8: $\mathcal{C} := \mathcal{C} \cup \{\lambda_\Omega(w)\}$

GIT-Algorithm

Algorithm

Input: *Ideal* $\mathfrak{a} \subset \mathbb{C}[T_1, \ldots, T_r]$ *and matrix* $Q \in \mathbb{Z}^{k \times r}$ *of full rank such that* $\mathfrak{a}$ *is homogeneous w.r.t. multigrading by* Q.

Output: *The set of maximal cones of* $\Lambda(\mathfrak{a}, Q)$.

1: $\mathcal{A} := \{\gamma \prec \mathbb{Q}^r_{\geq 0} \mid \gamma$ *is an* $\mathfrak{a}$*-face*$\}$
2: $\Omega := \{Q(\gamma) \mid \gamma \in \mathcal{A}\}$
3: *Choose a vector* $w_0 \in Q(\mathbb{Q}^r_{\geq 0})$ *such that* $\dim(\lambda_\Omega(w_0)) = k$.
4: $\mathcal{C} := \{\lambda_\Omega(w_0)\}$
5: $\mathcal{F} := \{(\eta, \lambda_\Omega(w_0)) \mid \eta \prec \lambda_\Omega(w_0)$ *interior facet*$\}$.
6: **while** *there is* $(\eta, \lambda) \in \mathcal{F}$ **do**
7: *Find* $w \in Q(\gamma)$ *such that* $w \notin \lambda$ *and* $\lambda_\Omega(w) \cap \lambda = \eta$.
8: $\mathcal{C} := \mathcal{C} \cup \{\lambda_\Omega(w)\}$
9: $\mathcal{F} := \mathcal{F} \ominus \{(\tau, \lambda_\Omega(w)) \mid \tau \subset \lambda_\Omega(w)$ *interior facet*$\}$

GIT-Algorithm

Algorithm

Input: *Ideal* $\mathfrak{a} \subset \mathbb{C}[T_1, \dots, T_r]$ *and matrix* $Q \in \mathbb{Z}^{k \times r}$ *of full rank such that* $\mathfrak{a}$ *is homogeneous w.r.t. multigrading by* Q.

Output: *The set of maximal cones of* $\Lambda(\mathfrak{a}, Q)$.

$\mathcal{A} := \{\gamma \prec \mathbb{Q}^r_{\geq 0} \mid \gamma \text{ is an } \mathfrak{a}\text{-face}\}$

$\Omega := \{Q(\gamma) \mid \gamma \in \mathcal{A}\}$

Choose a vector $w_0 \in Q(\mathbb{Q}^r_{\geq 0})$ *such that* $\dim(\lambda_\Omega(w_0)) = k$.

$\mathcal{C} := \{\lambda_\Omega(w_0)\}$

$\mathcal{F} := \{(\eta, \lambda_\Omega(w_0)) \mid \eta \prec \lambda_\Omega(w_0) \text{ interior facet}\}$.

while *there is* $(\eta, \lambda) \in \mathcal{F}$ **do**

    Find $w \in Q(\gamma)$ *such that* $w \notin \lambda$ *and* $\lambda_\Omega(w) \cap \lambda = \eta$.

    $\mathcal{C} := \mathcal{C} \cup \{\lambda_\Omega(w)\}$

    $\mathcal{F} := \mathcal{F} \ominus \{(\tau, \lambda_\Omega(w)) \mid \tau \subset \lambda_\Omega(w) \text{ interior facet}\}$

return $\mathcal{C}$

Fast Monomial Containment Test

Generalization of (Sturmfels, 1996), where degree reverse lex (dp) is used:

Fast Monomial Containment Test

Generalization of (Sturmfels, 1996), where degree reverse lex (dp) is used:

Proposition

Let $>$ be a monomial ordering on $R = K[Y_1, \ldots, Y_n]$ and $\mathcal{G}$ a Gröbner basis of I. Suppose that for all $f \in \mathcal{G}$

$$Y_n \mid f \quad \Longleftrightarrow \quad Y_n \mid \mathrm{LM}_>(f).$$

Fast Monomial Containment Test

Generalization of (Sturmfels, 1996), where degree reverse lex (dp) is used:

Proposition

Let $>$ be a monomial ordering on $R = K[Y_1, \ldots, Y_n]$ and $\mathcal{G}$ a Gröbner basis of I. Suppose that for all $f \in \mathcal{G}$

$$Y_n \mid f \quad \Longleftrightarrow \quad Y_n \mid \mathrm{LM}_>(f).$$

Then

$$\left\{ \frac{f}{Y_n^i} \;\middle|\; f \in \mathcal{G} \text{ and } i \geq 0 \text{ maximal such that } Y_n^i \mid f \right\}$$

is a Gröbner basis for $I : Y_n^\infty$.

Fast Monomial Containment Test

Generalization of (Sturmfels, 1996), where degree reverse lex (dp) is used:

Proposition

Let $>$ be a monomial ordering on $R = K[Y_1, \ldots, Y_n]$ and $\mathcal{G}$ a Gröbner basis of I. Suppose that for all $f \in \mathcal{G}$

$$Y_n \mid f \quad \Longleftrightarrow \quad Y_n \mid \mathrm{LM}_>(f).$$

Then

$$\left\{ \frac{f}{Y_n^i} \;\middle|\; f \in \mathcal{G} \text{ and } i \geq 0 \text{ maximal such that } Y_n^i \mid f \right\}$$

is a Gröbner basis for $I : Y_n^\infty$.

Algorithm

To compute $I : (Y_1 \cdot \ldots \cdot Y_n)^\infty$,

Fast Monomial Containment Test

Generalization of (Sturmfels, 1996), where degree reverse lex (dp) is used:

Proposition

Let $>$ be a monomial ordering on $R = K[Y_1, \ldots, Y_n]$ and $\mathcal{G}$ a Gröbner basis of I. Suppose that for all $f \in \mathcal{G}$

$$Y_n \mid f \quad \Longleftrightarrow \quad Y_n \mid \mathrm{LM}_>(f).$$

Then

$$\left\{ \frac{f}{Y_n^i} \;\middle|\; f \in \mathcal{G} \text{ and } i \geq 0 \text{ maximal such that } Y_n^i \mid f \right\}$$

is a Gröbner basis for $I : Y_n^\infty$.

Algorithm

To compute $I : (Y_1 \cdot \ldots \cdot Y_n)^\infty$, replace any remainder $r \neq 0$ in Buchberger's algorithm by

$$\frac{r}{Y_1^{a_1} \cdot \ldots \cdot Y_n^{a_n}} \quad \text{where } a_j \text{ is maximal s.t. } Y_j^{a_j} \mid r.$$

Timings

Saturation in product of variables for ideal $\mathfrak{a}$ with 225 generators in 40 variables with variables not in J equal to 0:

Timings

Saturation in product of variables for ideal $\mathfrak{a}$ with 225 generators in 40 variables with variables not in J equal to 0:

$\{1,\ldots,40\}\setminus J$	$40-\lvert J\rvert$	$\mathfrak{a}$-face	divgbsat	gbsat	sat	rabinowitsch
$\{3,4,5,7,\ldots,15\}$	28	no	1	761	517	342
$\{9,11,12,13,15\}$	35	no	1	57200	$*$	$*$
$\{11,12,13,15\}$	36	no	1	44100	$*$	$*$
$\{9,11,14,15\}$	36	yes	64	121000	$*$	$*$
$\{9,11,15\}$	37	yes	1170	114000	$*$	$*$
$\{9,11,13\}$	37	no	1	31400	$*$	$*$

(in seconds, * did not finish in >2 days)

Symmetry Groups of Torus Actions

Definition

A **symmetry group** of the action of $(\mathbb{C}^*)^k$ on X is a subgroup $G \subset S_r$ of the symmetric group such that there are group actions

Definition

A **symmetry group** of the action of $(\mathbb{C}^*)^k$ on X is a subgroup $G \subset S_r$ of the symmetric group such that there are group actions

$$\begin{array}{lllll} G \times \mathbb{K}[T_1, \ldots, T_r] & \to & \mathbb{K}[T_1, \ldots, T_r], & (\sigma, T_j) & \mapsto & \sigma(T_j) = c_{\sigma,j} \cdot T_{\sigma(j)} \\ G \times \mathbb{Q}^r & \to & \mathbb{Q}^r, & (\sigma, e_j) & \mapsto & \sigma(e_j) = e_{\sigma(j)} \\ G \times \mathbb{Q}^k & \to & \mathbb{Q}^k, & (\sigma, v) & \mapsto & A_\sigma \cdot v \end{array}$$

Definition

A **symmetry group** of the action of $(\mathbb{C}^*)^k$ on X is a subgroup $G \subset S_r$ of the symmetric group such that there are group actions

$$\begin{array}{llllll} G & \times & \mathbb{K}[T_1,\ldots,T_r] & \to & \mathbb{K}[T_1,\ldots,T_r], & (\sigma, T_j) & \mapsto & \sigma(T_j) = c_{\sigma,j}\cdot T_{\sigma(j)} \\ G & \times & \mathbb{Q}^r & \to & \mathbb{Q}^r, & (\sigma, e_j) & \mapsto & \sigma(e_j) = e_{\sigma(j)} \\ G & \times & \mathbb{Q}^k & \to & \mathbb{Q}^k, & (\sigma, v) & \mapsto & A_\sigma \cdot v \end{array}$$

with $A_\sigma \in \mathrm{GL}(k,\mathbb{Q})$ and $c_\sigma \in \mathbb{T}^r$

Definition

A **symmetry group** of the action of $(\mathbb{C}^*)^k$ on X is a subgroup $G \subset S_r$ of the symmetric group such that there are group actions

$$\begin{array}{lllll} G \times \mathbb{K}[T_1,\ldots,T_r] & \to & \mathbb{K}[T_1,\ldots,T_r], & (\sigma, T_j) & \mapsto \sigma(T_j) = c_{\sigma,j} \cdot T_{\sigma(j)} \\ G \times \mathbb{Q}^r & \to & \mathbb{Q}^r, & (\sigma, e_j) & \mapsto \sigma(e_j) = e_{\sigma(j)} \\ G \times \mathbb{Q}^k & \to & \mathbb{Q}^k, & (\sigma, v) & \mapsto A_\sigma \cdot v \end{array}$$

with $A_\sigma \in \mathrm{GL}(k,\mathbb{Q})$ and $c_\sigma \in \mathbb{T}^r$ such that $G \cdot \mathfrak{a} = \mathfrak{a}$ and that for each $\sigma \in G$ the following diagram is commutative:

Definition

A **symmetry group** of the action of $(\mathbb{C}^*)^k$ on X is a subgroup $G \subset S_r$ of the symmetric group such that there are group actions

$$\begin{array}{llllll} G \times \mathbb{K}[T_1,\ldots,T_r] & \to & \mathbb{K}[T_1,\ldots,T_r], & (\sigma, T_j) & \mapsto & \sigma(T_j) = c_{\sigma,j} \cdot T_{\sigma(j)} \\ G \times \mathbb{Q}^r & \to & \mathbb{Q}^r, & (\sigma, e_j) & \mapsto & \sigma(e_j) = e_{\sigma(j)} \\ G \times \mathbb{Q}^k & \to & \mathbb{Q}^k, & (\sigma, v) & \mapsto & A_\sigma \cdot v \end{array}$$

with $A_\sigma \in \mathrm{GL}(k,\mathbb{Q})$ and $c_\sigma \in \mathbb{T}^r$ such that $G \cdot \mathfrak{a} = \mathfrak{a}$ and that for each $\sigma \in G$ the following diagram is commutative:

$$\begin{array}{ccc} e_j & \longmapsto & e_{\sigma(j)} \\ \mathbb{Q}^r & \longrightarrow & \mathbb{Q}^r \\ Q \downarrow & & \downarrow Q \\ \mathbb{Q}^k & \underset{A_\sigma}{\longrightarrow} & \mathbb{Q}^k \end{array}$$

Representation of GIT-Cones

Perfect hash function for cones with compatible group action

Perfect hash function for cones with compatible group action

$$h_\Omega\colon \Lambda(\mathfrak{a}, Q) \to \{0,1\}^\Omega, \qquad \lambda \mapsto \left[\begin{array}{l} \Omega \to \{0,1\} \\ \vartheta \mapsto \begin{cases} 1 & \lambda \subset \vartheta \\ 0 & \lambda \not\subset \vartheta \end{cases} \end{array} \right]$$

Perfect hash function for cones with compatible group action

$$h_\Omega \colon \Lambda(\mathfrak{a}, Q) \to \{0,1\}^\Omega, \qquad \lambda \mapsto \left[\begin{array}{l} \Omega \to \{0,1\} \\ \vartheta \mapsto \begin{cases} 1 & \lambda \subset \vartheta \\ 0 & \lambda \not\subset \vartheta \end{cases} \end{array}\right]$$

$$G \times \{0,1\}^\Omega \to \{0,1\}^\Omega, \qquad (g, b) \mapsto \left[\begin{array}{l} \Omega \to \{0,1\} \\ \vartheta \mapsto b(g^{-1} \cdot \vartheta) \end{array}\right]$$

Perfect hash function for cones with compatible group action

$$h_\Omega\colon \Lambda(\mathfrak{a}, Q) \to \{0,1\}^\Omega, \qquad \lambda \mapsto \left[\begin{array}{l} \Omega \to \{0,1\} \\ \vartheta \mapsto \begin{cases} 1 & \lambda \subset \vartheta \\ 0 & \lambda \not\subset \vartheta \end{cases} \end{array} \right]$$

$$G \times \{0,1\}^\Omega \to \{0,1\}^\Omega, \qquad (g,b) \mapsto \left[\begin{array}{l} \Omega \to \{0,1\} \\ \vartheta \mapsto b(g^{-1} \cdot \vartheta) \end{array} \right]$$

such that

$$g \cdot h_\Omega(\lambda) = h_\Omega(g \cdot \lambda).$$

Symmetric GIT-Algorithm

Algorithm (System of representatives of the G-orbits on $\Lambda(\mathfrak{a}, Q)(k)$)

1: $\mathcal{S}$:= *system of representatives of G-orbits of* $\mathrm{faces}(\mathbb{Q}^r_{\geq 0})$

Symmetric GIT-Algorithm

Algorithm (System of representatives of the G-orbits on $\Lambda(\mathfrak{a}, Q)(k)$)

1: $\mathcal{S} :=$ *system of representatives of G-orbits of* $\mathrm{faces}(\mathbb{Q}^r_{\geq 0})$
2: $\mathcal{A} := \{\gamma \in \mathcal{S} \mid \gamma$ *is* $\mathfrak{a}$*-face*$\}$

Symmetric GIT-Algorithm

Algorithm (System of representatives of the G-orbits on $\Lambda(\mathfrak{a}, Q)(k)$)

1: $\mathcal{S} :=$ *system of representatives of* *G-orbits of* $\text{faces}(\mathbb{Q}^r_{\geq 0})$
2: $\mathcal{A} := \{\gamma \in \mathcal{S} \mid \gamma$ *is* $\mathfrak{a}$*-face*$\}$
3: $\Omega := \bigcup_{\gamma \in \mathcal{A}} G \cdot Q(\gamma)$

Symmetric GIT-Algorithm

Algorithm (System of representatives of the G-orbits on $\Lambda(\mathfrak{a}, Q)(k)$)

1: $\mathcal{S} :=$ *system of representatives of G-orbits of* $\mathrm{faces}(\mathbb{Q}^r_{\geq 0})$
2: $\mathcal{A} := \{\gamma \in \mathcal{S} \mid \gamma$ *is* $\mathfrak{a}$*-face*$\}$
3: $\Omega := \bigcup_{\gamma \in \mathcal{A}} G \cdot Q(\gamma)$
4: $\Omega :=$ *set of minimal elements of* $\Omega(k)$

Algorithm (System of representatives of the G-orbits on $\Lambda(\mathfrak{a}, Q)(k)$)

1: $\mathcal{S} :=$ *system of representatives of G-orbits of* $\mathrm{faces}(\mathbb{Q}^r_{\geq 0})$

2: $\mathcal{A} := \{\gamma \in \mathcal{S} \mid \gamma$ *is* $\mathfrak{a}$*-face*$\}$

3: $\Omega := \bigcup_{\gamma \in \mathcal{A}} G \cdot Q(\gamma)$

4: $\Omega :=$ *set of minimal elements of* $\Omega(k)$

5: *Choose* $w_0 \in Q(\Gamma)$ *such that* $\dim(\lambda_\Omega(w_0)) = k$.

Algorithm (System of representatives of the G-orbits on $\Lambda(\mathfrak{a}, Q)(k)$)

1: $\mathcal{S} :=$ *system of representatives of* *G-orbits of* $\text{faces}(\mathbb{Q}^r_{\geq 0})$

2: $\mathcal{A} := \{\gamma \in \mathcal{S} \mid \gamma \text{ is } \mathfrak{a}\text{-face}\}$

3: $\Omega := \bigcup_{\gamma \in \mathcal{A}} G \cdot Q(\gamma)$

4: $\Omega :=$ *set of minimal elements of* $\Omega(k)$

5: *Choose* $w_0 \in Q(\Gamma)$ *such that* $\dim(\lambda_\Omega(w_0)) = k$.

6: $\mathcal{C} := \{\lambda_\Omega(w_0)\}$, $\mathcal{H} := \{h_\Omega(\lambda_\Omega(w_0))\}$

Symmetric GIT-Algorithm

Algorithm (System of representatives of the G-orbits on $\Lambda(\mathfrak{a}, Q)(k)$)

1: $\mathcal{S} :=$ *system of representatives of* *G-orbits of* $\text{faces}(\mathbb{Q}^r_{\geq 0})$
2: $\mathcal{A} := \{\gamma \in \mathcal{S} \mid \gamma$ *is* $\mathfrak{a}$*-face*$\}$
3: $\Omega := \bigcup_{\gamma \in \mathcal{A}} G \cdot Q(\gamma)$
4: $\Omega :=$ *set of minimal elements of* $\Omega(k)$
5: *Choose* $w_0 \in Q(\Gamma)$ *such that* $\dim(\lambda_\Omega(w_0)) = k$.
6: $\mathcal{C} := \{\lambda_\Omega(w_0)\}$, $\mathcal{H} := \{h_\Omega(\lambda_\Omega(w_0))\}$
7: $\mathcal{F} := \{(\eta, v) \mid \eta \prec \lambda_\Omega(w_0)$ *interior facet with inner normal* $v\}$

Symmetric GIT-Algorithm

Algorithm (System of representatives of the G-orbits on $\Lambda(\mathfrak{a}, Q)(k)$)

$\mathcal{S} :=$ *system of representatives of G-orbits of* $\mathrm{faces}(\mathbb{Q}^r_{\geq 0})$
$\mathcal{A} := \{\gamma \in \mathcal{S} \mid \gamma$ *is* $\mathfrak{a}$*-face*$\}$
$\Omega := \bigcup_{\gamma \in \mathcal{A}} G \cdot Q(\gamma)$
$\Omega :=$ *set of minimal elements of* $\Omega(k)$
Choose $w_0 \in Q(\Gamma)$ *such that* $\dim(\lambda_\Omega(w_0)) = k$.
$\mathcal{C} := \{\lambda_\Omega(w_0)\}$, $\mathcal{H} := \{h_\Omega(\lambda_\Omega(w_0))\}$
$\mathcal{F} := \{(\eta, v) \mid \eta \prec \lambda_\Omega(w_0)$ *interior facet with inner normal* $v\}$
while *there is* $(\eta, v) \in \mathcal{F}$ **do**
    Find $w \in Q(\Gamma)$ *such that* $\eta \prec \lambda_\Omega(w)$ *is a facet and* $-v \in \lambda_\Omega(w)^\vee$.

Algorithm (System of representatives of the G-orbits on $\Lambda(\mathfrak{a}, Q)(k)$)

$\mathcal{S} :=$ *system of representatives of G-orbits of* faces$(\mathbb{Q}^r_{\geq 0})$
$\mathcal{A} := \{\gamma \in \mathcal{S} \mid \gamma$ *is* $\mathfrak{a}$*-face*$\}$
$\Omega := \bigcup_{\gamma \in \mathcal{A}} G \cdot Q(\gamma)$
$\Omega :=$ *set of minimal elements of* $\Omega(k)$
Choose $w_0 \in Q(\Gamma)$ *such that* $\dim(\lambda_\Omega(w_0)) = k$.
$\mathcal{C} := \{\lambda_\Omega(w_0)\}$, $\mathcal{H} := \{h_\Omega(\lambda_\Omega(w_0))\}$
$\mathcal{F} := \{(\eta, v) \mid \eta \prec \lambda_\Omega(w_0)$ *interior facet with inner normal* $v\}$
while *there is* $(\eta, v) \in \mathcal{F}$ **do**
    Find $w \in Q(\Gamma)$ *such that* $\eta \prec \lambda_\Omega(w)$ *is a facet and* $-v \in \lambda_\Omega(w)^\vee$.
    if $G \cdot h_\Omega(\lambda_\Omega(w)) \cap \mathcal{H} = \emptyset$ **then**
        $\mathcal{C} := \mathcal{C} \cup \{\lambda_\Omega(w)\}, \mathcal{H} := \mathcal{H} \cup \{h_\Omega(\lambda_\Omega(w))\}$

Symmetric GIT-Algorithm

Algorithm (System of representatives of the G-orbits on $\Lambda(\mathfrak{a}, Q)(k)$)

$\mathcal{S}$:= *system of representatives of G-orbits of* $\text{faces}(\mathbb{Q}^r_{\geq 0})$

$\mathcal{A} := \{\gamma \in \mathcal{S} \mid \gamma \text{ is } \mathfrak{a}\text{-face}\}$

$\Omega := \bigcup_{\gamma \in \mathcal{A}} G \cdot Q(\gamma)$

Ω := *set of minimal elements of* $\Omega(k)$

Choose $w_0 \in Q(\Gamma)$ *such that* $\dim(\lambda_\Omega(w_0)) = k$.

$\mathcal{C} := \{\lambda_\Omega(w_0)\}$, $\mathcal{H} := \{h_\Omega(\lambda_\Omega(w_0))\}$

$\mathcal{F} := \{(\eta, v) \mid \eta \prec \lambda_\Omega(w_0) \text{ interior facet with inner normal } v\}$

while *there is* $(\eta, v) \in \mathcal{F}$ **do**

    Find $w \in Q(\Gamma)$ *such that* $\eta \prec \lambda_\Omega(w)$ *is a facet and* $-v \in \lambda_\Omega(w)^\vee$.

    if $G \cdot h_\Omega(\lambda_\Omega(w)) \cap \mathcal{H} = \emptyset$ **then**

        $\mathcal{C} := \mathcal{C} \cup \{\lambda_\Omega(w)\}, \mathcal{H} := \mathcal{H} \cup \{h_\Omega(\lambda_\Omega(w))\}$

        $\mathcal{F} := \mathcal{F} \ominus \{(\tilde{\eta}, \tilde{v}) \mid \tilde{\eta} \prec \lambda_\Omega(w) \text{ interior facet w. inner normal } \tilde{v}\}$

Symmetric GIT-Algorithm

Algorithm (System of representatives of the G-orbits on $\Lambda(\mathfrak{a}, Q)(k)$)

$\mathcal{S} :=$ *system of representatives of G-orbits of* $\mathrm{faces}(\mathbb{Q}^r_{\geq 0})$
$\mathcal{A} := \{\gamma \in \mathcal{S} \mid \gamma \text{ is } \mathfrak{a}\text{-face}\}$
$\Omega := \bigcup_{\gamma \in \mathcal{A}} G \cdot Q(\gamma)$
$\Omega :=$ *set of minimal elements of* $\Omega(k)$
Choose $w_0 \in Q(\Gamma)$ *such that* $\dim(\lambda_\Omega(w_0)) = k$.
$\mathcal{C} := \{\lambda_\Omega(w_0)\}$, $\mathcal{H} := \{h_\Omega(\lambda_\Omega(w_0))\}$
$\mathcal{F} := \{(\eta, v) \mid \eta \prec \lambda_\Omega(w_0) \text{ interior facet with inner normal } v\}$
while *there is* $(\eta, v) \in \mathcal{F}$ **do**
    Find $w \in Q(\Gamma)$ *such that* $\eta \prec \lambda_\Omega(w)$ *is a facet and* $-v \in \lambda_\Omega(w)^\vee$.
    if $G \cdot h_\Omega(\lambda_\Omega(w)) \cap \mathcal{H} = \emptyset$ **then**
        $\mathcal{C} := \mathcal{C} \cup \{\lambda_\Omega(w)\}, \mathcal{H} := \mathcal{H} \cup \{h_\Omega(\lambda_\Omega(w))\}$
        $\mathcal{F} := \mathcal{F} \ominus \{(\tilde{\eta}, \tilde{v}) \mid \tilde{\eta} \prec \lambda_\Omega(w) \text{ interior facet w. inner normal } \tilde{v}\}$
    else
        $\mathcal{F} := \mathcal{F} \setminus \{(\eta, v)\}$

Symmetric GIT-Algorithm

Algorithm (System of representatives of the G-orbits on $\Lambda(\mathfrak{a}, Q)(k)$)

$\mathcal{S} :=$ *system of representatives of G-orbits of* $\mathrm{faces}(\mathbb{Q}^r_{\geq 0})$
$\mathcal{A} := \{\gamma \in \mathcal{S} \mid \gamma \text{ is } \mathfrak{a}\text{-face}\}$
$\Omega := \bigcup_{\gamma \in \mathcal{A}} G \cdot Q(\gamma)$
$\Omega :=$ *set of minimal elements of* $\Omega(k)$
Choose $w_0 \in Q(\Gamma)$ *such that* $\dim(\lambda_\Omega(w_0)) = k$.
$\mathcal{C} := \{\lambda_\Omega(w_0)\}$, $\mathcal{H} := \{h_\Omega(\lambda_\Omega(w_0))\}$
$\mathcal{F} := \{(\eta, v) \mid \eta \prec \lambda_\Omega(w_0) \text{ interior facet with inner normal } v\}$
while *there is* $(\eta, v) \in \mathcal{F}$ **do**
    Find $w \in Q(\Gamma)$ *such that* $\eta \prec \lambda_\Omega(w)$ *is a facet and* $-v \in \lambda_\Omega(w)^\vee$.
    if $G \cdot h_\Omega(\lambda_\Omega(w)) \cap \mathcal{H} = \emptyset$ **then**
        $\mathcal{C} := \mathcal{C} \cup \{\lambda_\Omega(w)\}, \mathcal{H} := \mathcal{H} \cup \{h_\Omega(\lambda_\Omega(w))\}$
        $\mathcal{F} := \mathcal{F} \ominus \{(\tilde{\eta}, \tilde{v}) \mid \tilde{\eta} \prec \lambda_\Omega(w) \text{ interior facet w. inner normal } \tilde{v}\}$
    else
        $\mathcal{F} := \mathcal{F} \setminus \{(\eta, v)\}$
return $\mathcal{C}$

An Example with D_4-Symmetry

Example

$$\mathfrak{a} = \langle T_1 T_3 - T_2 T_4 \rangle \subset \mathbb{K}[T_1, \ldots, T_4] \qquad \deg(T_j) = q_j$$

$$Q = (q_1, \ldots, q_4) = \begin{pmatrix} 1 & -1 & -1 & 1 \\ 1 & 1 & -1 & -1 \end{pmatrix}$$

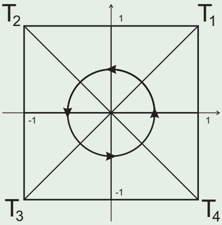
An Example with D_4-Symmetry

Example

$$\mathfrak{a} = \langle T_1 T_3 - T_2 T_4 \rangle \subset \mathbb{K}[T_1, \ldots, T_4] \qquad \deg(T_j) = q_j$$

$$Q = (q_1, \ldots, q_4) = \begin{pmatrix} 1 & -1 & -1 & 1 \\ 1 & 1 & -1 & -1 \end{pmatrix}$$

$$G = D_4 = \langle (1,2)(3,4), (1,2,3,4) \rangle \subset S_4$$



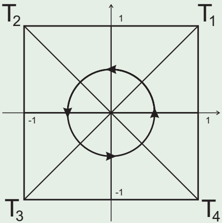
An Example with D_4-Symmetry

Example

$$\mathfrak{a} = \langle T_1 T_3 - T_2 T_4 \rangle \subset \mathbb{K}[T_1, \ldots, T_4] \qquad \deg(T_j) = q_j$$

$$Q = (q_1, \ldots, q_4) = \begin{pmatrix} 1 & -1 & -1 & 1 \\ 1 & 1 & -1 & -1 \end{pmatrix}$$

$$G = D_4 = \langle (1,2)(3,4), (1,2,3,4) \rangle \subset S_4$$



$$A_{(1,2)(3,4)} = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix} \qquad A_{(1,2,3,4)} = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$$

Example with D_4-symmetry

Example

γ	$\lvert G\cdot\gamma\rvert$	$\mathfrak{a}\vert_{T_i=0 \text{ for } e_i\notin\gamma}$	$\mathfrak{a}$-face
$\gamma_0 = \text{cone}(0)$	1	0	`true`
$\gamma_1 = \text{cone}(e_1)$	4	0	`true`
$\gamma_2 = \text{cone}(e_1, e_2)$	4	0	`true`
$\gamma_2' = \text{cone}(e_1, e_3)$	2	$\langle T_1 T_3\rangle$	`false`
$\gamma_3 = \text{cone}(e_1, e_2, e_3)$	4	$\langle T_1 T_3\rangle$	`false`
$\gamma_4 = \text{cone}(e_1, e_2, e_3, e_4)$	1	$\langle T_1 T_3 - T_2 T_4\rangle$	`true`

Example

γ	$\lvert G\cdot\gamma\rvert$	$\mathfrak{a}\vert_{T_i=0 \text{ for } e_i\notin\gamma}$	$\mathfrak{a}$-face
$\gamma_0 = \text{cone}(0)$	1	0	true
$\gamma_1 = \text{cone}(e_1)$	4	0	true
$\gamma_2 = \text{cone}(e_1, e_2)$	4	0	true
$\gamma_2' = \text{cone}(e_1, e_3)$	2	$\langle T_1 T_3\rangle$	false
$\gamma_3 = \text{cone}(e_1, e_2, e_3)$	4	$\langle T_1 T_3\rangle$	false
$\gamma_4 = \text{cone}(e_1, e_2, e_3, e_4)$	1	$\langle T_1 T_3 - T_2 T_4\rangle$	true

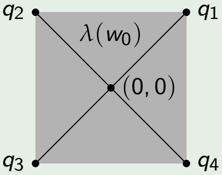
$$Q(\gamma_0) = \text{cone}(0), \quad Q(\gamma_1) = \text{cone}\begin{bmatrix}1\\1\end{bmatrix}, \quad Q(\gamma_2) = \text{cone}\left(\begin{bmatrix}1\\1\end{bmatrix}, \begin{bmatrix}-1\\1\end{bmatrix}\right), \quad Q(\gamma_4) = \mathbb{Q}^2$$

Example

γ	$\lvert G\cdot\gamma\rvert$	$\mathfrak{a}\vert_{T_i=0 \text{ for } e_i\notin\gamma}$	$\mathfrak{a}$-face
$\gamma_0 = \text{cone}(0)$	1	0	true
$\gamma_1 = \text{cone}(e_1)$	4	0	true
$\gamma_2 = \text{cone}(e_1, e_2)$	4	0	true
$\gamma_2' = \text{cone}(e_1, e_3)$	2	$\langle T_1T_3\rangle$	false
$\gamma_3 = \text{cone}(e_1, e_2, e_3)$	4	$\langle T_1T_3\rangle$	false
$\gamma_4 = \text{cone}(e_1, e_2, e_3, e_4)$	1	$\langle T_1T_3 - T_2T_4\rangle$	true

$$Q(\gamma_0) = \text{cone}(0), \quad Q(\gamma_1) = \text{cone}\begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad Q(\gamma_2) = \text{cone}\left(\begin{bmatrix} 1 \\ 1 \end{bmatrix}, \begin{bmatrix} -1 \\ 1 \end{bmatrix}\right), \quad Q(\gamma_4) = \mathbb{Q}^2$$

$$w_0 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$



Mori Dream Spaces

A projective variety X over $\mathbb{C}$ is called a **Mori dream space** if its Cox ring $R(X) = \sum_{[D]\in \mathrm{Cl}(X)} H^0(X, \mathcal{O}_X(D))$ is finitely generated.

Example

- Fano varieties.
- Projective toric varieties ($\Leftrightarrow R(X)$ polynomial ring).

Like toric varieties, admit construction as GIT-quotient (Hu, Keel, 2000):

$$X = \hat{X} /\!/ G$$

where

$$\begin{array}{ccc} \hat{X} & \subset & \overline{X} := \operatorname{Spec} R(X) \text{ open invariant} \\ & & G := \operatorname{Spec} \mathbb{C}[\mathrm{Cl}(X)] \end{array}$$

Mori Dream Spaces

A projective variety X over $\mathbb{C}$ is called a **Mori dream space** if its Cox ring $R(X) = \sum_{[D]\in \mathrm{Cl}(X)} H^0(X, \mathcal{O}_X(D))$ is finitely generated.

Example

- Fano varieties.
- Projective toric varieties ($\Leftrightarrow R(X)$ polynomial ring).

Like toric varieties, admit construction as GIT-quotient (Hu, Keel, 2000):

$$X = \hat{X} /\!/ G$$

where

$$\begin{aligned} \hat{X} \quad \subset \quad \overline{X} &:= \operatorname{Spec} R(X) \text{ open invariant} \\ G &:= \operatorname{Spec} \mathbb{C}[\mathrm{Cl}(X)] \end{aligned}$$

Remark

The GIT-fan yields the Mori chamber decomposition, which describes all birational modifications (analogous to the GKZ-fan of a toric varietiy).

Moduli Spaces of Stable Maps

For the Deligne-Mumford compactification **moduli space of stable curves** of genus 0 with n marked points $\overline{M}_{0,n}$ (only double points, on each component ≥ 3 marked or double points) we have:

Moduli Spaces of Stable Maps

For the Deligne-Mumford compactification **moduli space of stable curves** of genus 0 with n marked points $\overline{M}_{0,n}$ (only double points, on each component ≥ 3 marked or double points) we have:

- $\overline{M}_{0,n}$ for $n \leq 6$ is a Mori dream space:
  - Castravet, 2009, for $n = 6$.

For the Deligne-Mumford compactification **moduli space of stable curves** of genus 0 with n marked points $\overline{M}_{0,n}$ (only double points, on each component ≥ 3 marked or double points) we have:

- $\overline{M}_{0,n}$ for $n \leq 6$ is a Mori dream space:
  - Castravet, 2009, for $n = 6$.
- $\overline{M}_{0,n}$ for $n \geq 10$ is not a Mori dream space:
  - Castravet, Tevelev, 2013, for $n \geq 134$.
  - Gonzáles, Karu, 2016, for $n \geq 13$.
  - Hausen, Keicher, Laface, 2016, for $n \geq 10$.

Example: GIT-fan for $\mathbb{G}(2,5)$

Example

Cox ring of $\overline{M}_{0,5}$ is isomorphic to $\mathbb{Z}^5$-graded coordinate ring $R = \mathbb{K}[T_1, \ldots, T_{10}]/\mathfrak{a}$ of affine cone over $\mathbb{G}(2,5)$.

Example: GIT-fan for $\mathbb{G}(2,5)$

Example

Cox ring of $\overline{M}_{0,5}$ is isomorphic to $\mathbb{Z}^5$-graded coordinate ring $R = \mathbb{K}[T_1, \ldots, T_{10}]/\mathfrak{a}$ of affine cone over $\mathbb{G}(2,5)$. Symmetry group action of

$$S_5 \cong G = \langle (2,3)(5,6)(9,10),\ (1,5,9,10,3)(2,7,8,4,6) \rangle \subset S_{10}$$

Example: GIT-fan for $\mathbb{G}(2,5)$

Example

Cox ring of $\overline{M}_{0,5}$ is isomorphic to $\mathbb{Z}^5$-graded coordinate ring $R = \mathbb{K}[T_1, \ldots, T_{10}]/\mathfrak{a}$ of affine cone over $\mathbb{G}(2,5)$. Symmetry group action of

$$S_5 \cong G = \langle (2,3)(5,6)(9,10),\ (1,5,9,10,3)(2,7,8,4,6) \rangle \subset S_{10}$$

	number	number of orbits
monomial containment tests	$2^{10} = 1024$	34
$\mathfrak{a}$-faces	172	14

Example: GIT-fan for $\mathbb{G}(2,5)$

Example

Cox ring of $\overline{M}_{0,5}$ is isomorphic to $\mathbb{Z}^5$-graded coordinate ring $R = \mathbb{K}[T_1, \ldots, T_{10}]/\mathfrak{a}$ of affine cone over $\mathbb{G}(2,5)$. Symmetry group action of

$$S_5 \cong G = \langle (2,3)(5,6)(9,10),\ (1,5,9,10,3)(2,7,8,4,6) \rangle \subset S_{10}$$

	number	number of orbits
monomial containment tests	$2^{10} = 1024$	34
$\mathfrak{a}$-faces	172	14

$172 = (1+1) + (5+5) + (10+10+10+10+10) + (15+15) + 20 + (30+30)$

Example

Cox ring of $\overline{M}_{0,5}$ is isomorphic to $\mathbb{Z}^5$-graded coordinate ring $R = \mathbb{K}[T_1, \ldots, T_{10}]/\mathfrak{a}$ of affine cone over $\mathbb{G}(2,5)$. Symmetry group action of

$$S_5 \cong G = \langle (2,3)(5,6)(9,10),\ (1,5,9,10,3)(2,7,8,4,6) \rangle \subset S_{10}$$

	number	number of orbits
monomial containment tests	$2^{10} = 1024$	34
$\mathfrak{a}$-faces	172	14

$172 = (1+1) + (5+5) + (10+10+10+10+10) + (15+15) + 20 + (30+30)$

$$|\Omega(5)/G| = 4$$

Example: GIT-fan for $\mathbb{G}(2,5)$

Example

Cox ring of $\overline{M}_{0,5}$ is isomorphic to $\mathbb{Z}^5$-graded coordinate ring $R = \mathbb{K}[T_1, \ldots, T_{10}]/\mathfrak{a}$ of affine cone over $\mathbb{G}(2,5)$. Symmetry group action of

$$S_5 \cong G = \langle (2,3)(5,6)(9,10),\ (1,5,9,10,3)(2,7,8,4,6) \rangle \subset S_{10}$$

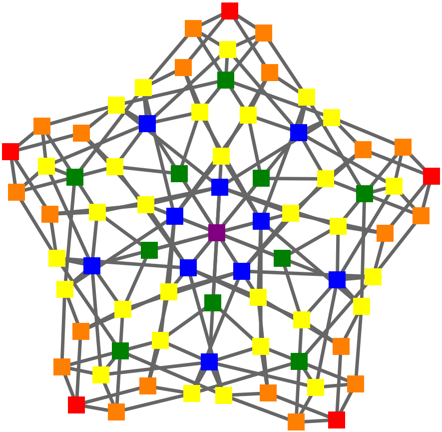
	number	number of orbits
monomial containment tests	$2^{10} = 1024$	34
$\mathfrak{a}$-faces	172	14

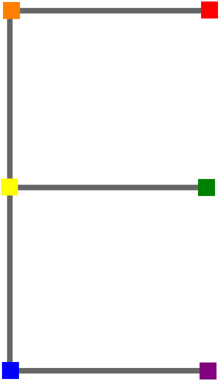
$$172 = (1+1) + (5+5) + (10+10+10+10+10) + (15+15) + 20 + (30+30)$$

$$|\Omega(5)/G| = 4$$

$$|\Lambda(5)| = 76 = 1 + 10 + 30 + 10 + 20 + 5$$

Adjacency graph of the maximal-dimensional GIT-cones and their orbits:





Mori Chamber Decomposition of Mov($\overline{M}_{0,6}$)

The moving cone Mov($\overline{M}_{0,6}$) classifies all small modifications (rational maps which are isomorphisms on open subsets which have a complement of codimension ≥ 2).

Example

Cox ring is $\mathbb{Z}^{16}$-graded, has 40 generators (Castravet, 2009),

Mori Chamber Decomposition of Mov($\overline{M}_{0,6}$)

The moving cone Mov($\overline{M}_{0,6}$) classifies all small modifications (rational maps which are isomorphisms on open subsets which have a complement of codimension ≥ 2).

Example

Cox ring is $\mathbb{Z}^{16}$-graded, has 40 generators (Castravet, 2009), and 225 relations (Bernal Guillen, 2012),

Mori Chamber Decomposition of Mov($\overline{M}_{0,6}$)

The moving cone Mov($\overline{M}_{0,6}$) classifies all small modifications (rational maps which are isomorphisms on open subsets which have a complement of codimension ≥ 2).

Example

Cox ring is $\mathbb{Z}^{16}$-graded, has 40 generators (Castravet, 2009), and 225 relations (Bernal Guillen, 2012), and natural $G = S_6$–action.

Mori Chamber Decomposition of $\mathrm{Mov}(\overline{M}_{0,6})$

The moving cone $\mathrm{Mov}(\overline{M}_{0,6})$ classifies all small modifications (rational maps which are isomorphisms on open subsets which have a complement of codimension ≥ 2).

Example

Cox ring is $\mathbb{Z}^{16}$-graded, has 40 generators (Castravet, 2009), and 225 relations (Bernal Guillen, 2012), and natural $G = S_6$–action.
The moving cone $\mathrm{Mov}(\overline{M}_{0,6})$ has

$$176\,512\,225$$

GIT-cones of maximal dimension 16, which decompose into

$$249\,605$$

orbits under the S_6-action:

Mori Chamber Decomposition of $\mathrm{Mov}(\overline{M}_{0,6})$

The moving cone $\mathrm{Mov}(\overline{M}_{0,6})$ classifies all small modifications (rational maps which are isomorphisms on open subsets which have a complement of codimension ≥ 2).

Example

Cox ring is $\mathbb{Z}^{16}$-graded, has 40 generators (Castravet, 2009), and 225 relations (Bernal Guillen, 2012), and natural $G = S_6$–action.
The moving cone $\mathrm{Mov}(\overline{M}_{0,6})$ has

$$176\,512\,225$$

GIT-cones of maximal dimension 16, which decompose into

$$249\,605$$

orbits under the S_6-action:

cardinality	1	6	10	15	20	30	45	60
no. of orbits	1	1	1	4	1	1	10	27

cardinality	72	90	120	180	240	360	720
no. of orbits	4	46	32	488	4	7934	241051

The cone with orbit length one is the semiample cone (dual of Mori cone).

References

- J. Boehm, S. Keicher, Y. Ren. *Computing GIT-fans with symmetry and the Mori chamber decomposition of* $\overline{M}_{0,6}$, arXiv:1603.09241 (2016).
- S. Keicher. *Computing the GIT-fan*, Int. J. Algebra Comput. (2012).

- D. Mumford, J. Fogarty, F. Kirwan. *Geometric invariant theory*. Ergebnisse der Mathematik und ihrer Grenzgebiete. Springer (1994).
- I. V. Dolgachev and Y. Hu. *Variation of geometric invariant theory quotients*. Publ. Math., Inst. Hautes Etud. Sci. (1998).
- F. Berchtold, J. Hausen. *GIT equivalence beyond the ample cone*. Michigan Math. J. (2006).
- I. Arzhantsev, U. Derenthal, J. Hausen, A. Laface. *Cox Rings*, Cambridge studies in advanced mathematics (2014).

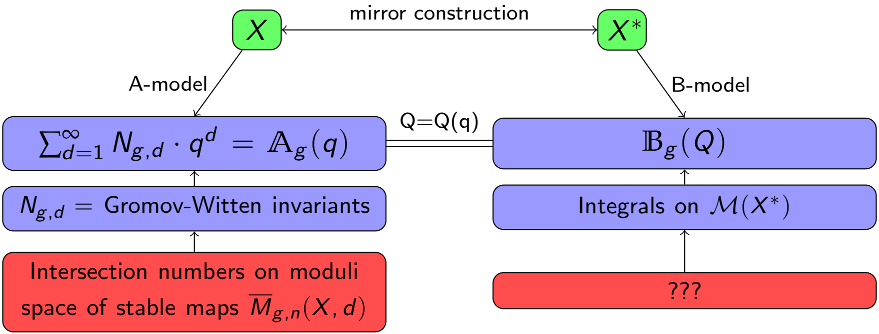
- A.-M. Castravet. *The Cox ring of* $\overline{M}_{0,6}$. Trans. Amer. Math. Soc. (2009).
- M. M. Bernal Guillen. *Relations in the Cox Ring of* $\overline{M}_{0,6}$. PhD (2012).

Mirror symmetry

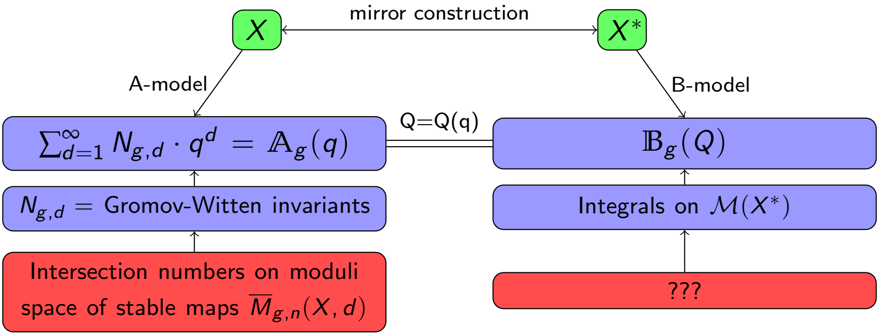
For Calabi-Yau variety X (elliptic curve, quintic in $\mathbb{P}^4$,...) and $g \in \mathbb{N}_0$:

Mirror symmetry

For Calabi-Yau variety X (elliptic curve, quintic in $\mathbb{P}^4$,...) and $g \in \mathbb{N}_0$:



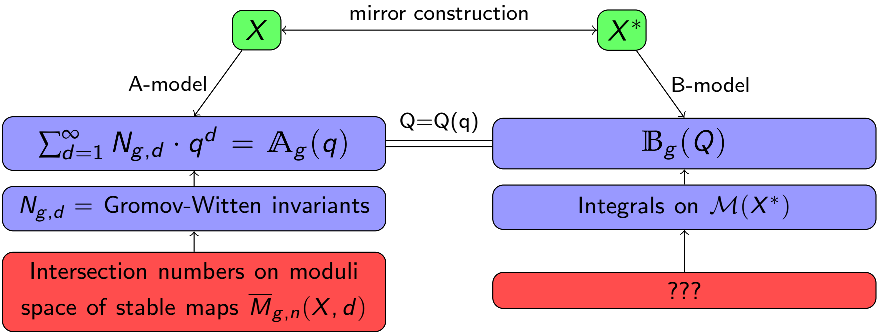
For Calabi-Yau variety X (elliptic curve, quintic in $\mathbb{P}^4$,...) and $g \in \mathbb{N}_0$:



- Mirror constructions: Greene-Plesser '90, Batyrev '93,...

Mirror symmetry

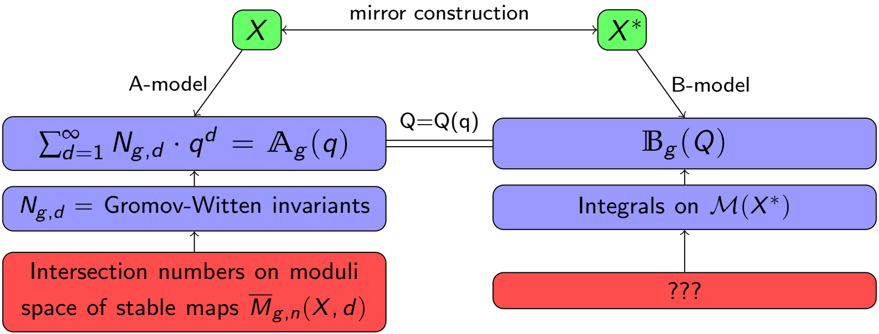
For Calabi-Yau variety X (elliptic curve, quintic in $\mathbb{P}^4$,...) and $g \in \mathbb{N}_0$:



- Mirror constructions: Greene-Plesser '90, Batyrev '93,...
- String theory: Candelas-Horowitz-Strominger-Witten '85, Candelas-de la Ossa-Green-Parkes '91,...

Mirror symmetry

For Calabi-Yau variety X (elliptic curve, quintic in $\mathbb{P}^4$,...) and $g \in \mathbb{N}_0$:



- Mirror constructions: Greene-Plesser '90, Batyrev '93,...
- String theory: Candelas-Horowitz-Strominger-Witten '85, Candelas-de la Ossa-Green-Parkes '91,...
- Algebraic/symplectic geometry: Fulton-Pandharipande '95, Kontsevich '95, Behrend-Fantechi '97,...

Mirror theorems

Theorem (Givental '96, Lian-Liu-Yau '97, Gathmann '03)

$\mathbb{A}_0 = \mathbb{B}_0$ *for quintic hypersurface in* $\mathbb{P}^4$.

Theorem (Givental '96, Lian-Liu-Yau '97, Gathmann '03)

$\mathbb{A}_0 = \mathbb{B}_0$ *for quintic hypersurface in* $\mathbb{P}^4$.

$$\Rightarrow \mathbb{A}_0(q) = 23 \cdot 5^3 + \left(4874 \cdot 5^3 + \frac{23 \cdot 5^3}{2^3}\right) \cdot q + \left(2537651 \cdot 5^3 + \frac{23 \cdot 5^3}{3^3}\right) \cdot q^2 + \ldots$$

Theorem (Givental '96, Lian-Liu-Yau '97, Gathmann '03)

$\mathbb{A}_0 = \mathbb{B}_0$ *for quintic hypersurface in* $\mathbb{P}^4$.

$$\Rightarrow \mathbb{A}_0(q) = 23 \cdot 5^3 + \left(4874 \cdot 5^3 + \frac{23 \cdot 5^3}{2^3}\right) \cdot q + \left(2537651 \cdot 5^3 + \frac{23 \cdot 5^3}{3^3}\right) \cdot q^2 + \ldots$$

Is enumerative geometry result on X: number of lines, conics, cubics,...
(number of genus 0 curves on X of degree d, delicate counting).

Mirror theorems

Theorem (Givental '96, Lian-Liu-Yau '97, Gathmann '03)

$\mathbb{A}_0 = \mathbb{B}_0$ *for quintic hypersurface in* $\mathbb{P}^4$.

$$\Rightarrow \mathbb{A}_0(q) = 23 \cdot 5^3 + (4874 \cdot 5^3 + \frac{23\cdot 5^3}{2^3}) \cdot q + (2537651 \cdot 5^3 + \frac{23\cdot 5^3}{3^3}) \cdot q^2 + \ldots$$

Is enumerative geometry result on X: number of lines, conics, cubics,...
(number of genus 0 curves on X of degree d, delicate counting).

Similar theorems for $g = 0, 1$ in case of degree $n+1$ hypersurfaces in $\mathbb{P}^n$ (Klemm-Pandharipande '07, Zinger '07)

Theorem (Givental '96, Lian-Liu-Yau '97, Gathmann '03)

$\mathbb{A}_0 = \mathbb{B}_0$ *for quintic hypersurface in* $\mathbb{P}^4$.

$$\Rightarrow \mathbb{A}_0(q) = 23 \cdot 5^3 + (4874 \cdot 5^3 + \frac{23 \cdot 5^3}{2^3}) \cdot q + (2537651 \cdot 5^3 + \frac{23 \cdot 5^3}{3^3}) \cdot q^2 + \ldots$$

Is enumerative geometry result on X: number of lines, conics, cubics,...
(number of genus 0 curves on X of degree d, delicate counting).

Similar theorems for $g = 0, 1$ in case of degree $n + 1$ hypersurfaces in $\mathbb{P}^n$
(Klemm-Pandharipande '07, Zinger '07)

Questions:

- Mirror theorems for other Calabi-Yau varieties and $g \geq 2$?

Mirror theorems

Theorem (Givental '96, Lian-Liu-Yau '97, Gathmann '03)

$\mathbb{A}_0 = \mathbb{B}_0$ *for quintic hypersurface in* $\mathbb{P}^4$.

$$\Rightarrow \mathbb{A}_0(q) = 23 \cdot 5^3 + (4874 \cdot 5^3 + \frac{23 \cdot 5^3}{2^3}) \cdot q + (2537651 \cdot 5^3 + \frac{23 \cdot 5^3}{3^3}) \cdot q^2 + \ldots$$

Is enumerative geometry result on X: number of lines, conics, cubics,...
(number of genus 0 curves on X of degree d, delicate counting).

Similar theorems for $g = 0, 1$ in case of degree $n+1$ hypersurfaces in $\mathbb{P}^n$
(Klemm-Pandharipande '07, Zinger '07)

Questions:

- Mirror theorems for other Calabi-Yau varieties and $g \geq 2$?
- Geometric understanding of mirror theorem beyond combinatorics?

Mirror theorems

Theorem (Givental '96, Lian-Liu-Yau '97, Gathmann '03)

$\mathbb{A}_0 = \mathbb{B}_0$ *for quintic hypersurface in* $\mathbb{P}^4$.

$$\Rightarrow \mathbb{A}_0(q) = 23 \cdot 5^3 + (4874 \cdot 5^3 + \tfrac{23 \cdot 5^3}{2^3}) \cdot q + (2537651 \cdot 5^3 + \tfrac{23 \cdot 5^3}{3^3}) \cdot q^2 + \ldots$$

Is enumerative geometry result on X: number of lines, conics, cubics,...
(number of genus 0 curves on X of degree d, delicate counting).

Similar theorems for $g = 0, 1$ in case of degree $n + 1$ hypersurfaces in $\mathbb{P}^n$
(Klemm-Pandharipande '07, Zinger '07)

Questions:

- Mirror theorems for other Calabi-Yau varieties and $g \geq 2$?
- Geometric understanding of mirror theorem beyond combinatorics?
- What are the B-model integrals?

Start with easiest Calabi-Yau: elliptic curve E (e.g. smooth plane cubic).
Here, Gromov-Witten numbers are numbers of covers:

Start with easiest Calabi-Yau: elliptic curve E (e.g. smooth plane cubic). Here, Gromov-Witten numbers are numbers of covers:

Definition (Hurwitz numbers)

$N_{d,g} = \frac{1}{|\mathrm{Aut}(f)|}$-weighted number of degree d covers $f : C \to E$, where C is smooth of genus g and f has $2g-2$ simple ramifications points.

according to Riemann-Hurwitz formula $2g(C)-2 = d\cdot(2g(E)-2)+\sum_{P\in C}(e(P)-1)$

Elliptic curves

Start with easiest Calabi-Yau: elliptic curve E (e.g. smooth plane cubic). Here, Gromov-Witten numbers are numbers of covers:

Definition (Hurwitz numbers)

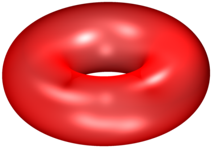
$N_{d,g} = \frac{1}{|\mathrm{Aut}(f)|}$-weighted number of degree d covers $f : C \to E$, where C is smooth of genus g and f has $2g-2$ simple ramifications points.

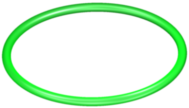
according to Riemann-Hurwitz formula $2g(C) - 2 = d \cdot (2g(E) - 2) + \sum_{P \in C} (e(P) - 1)$

$N_{d,0} = 0$, so have to look at $g \geq 1$ invariants!

How to understand *all* $N_{g,d}$? Pass to **tropical geometry**:

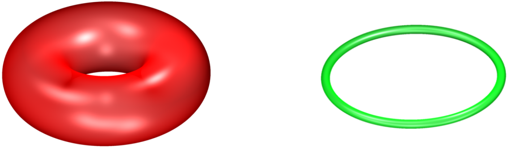
$$E \quad \mapsto \quad \mathrm{trop}(E)$$

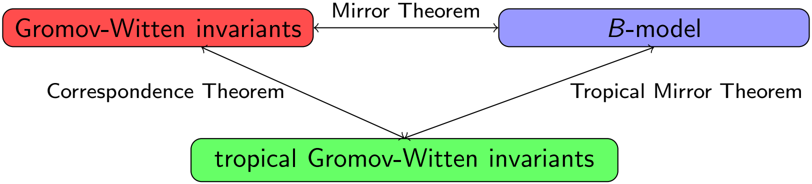




How to understand *all* $N_{g,d}$? Pass to **tropical geometry**:

$$E \mapsto \mathrm{trop}(E)$$

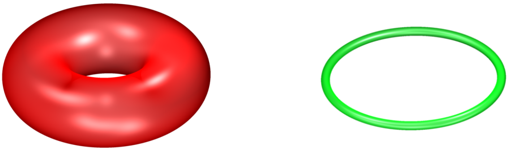


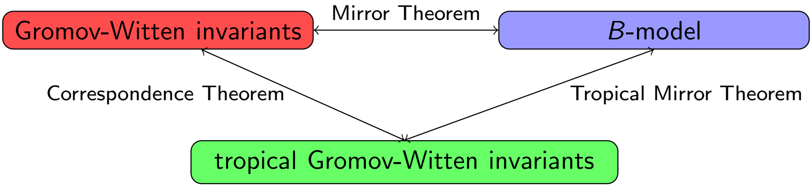


Tropical point of view

How to understand *all* $N_{g,d}$? Pass to **tropical geometry**:

$$E \mapsto \mathrm{trop}(E)$$



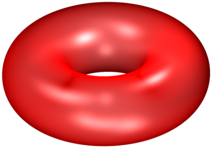


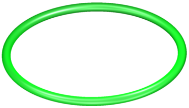
For $X = \mathbb{P}^2$ (building block of C-Y) and $g = 0$:

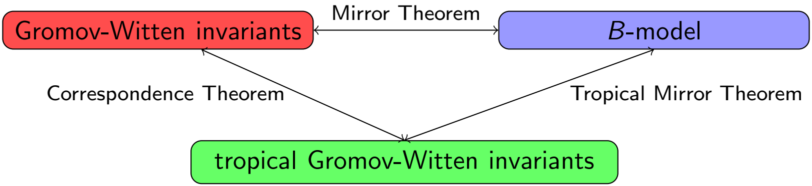
- tropical mirror theorem (Gross '10)

How to understand *all* $N_{g,d}$? Pass to **tropical geometry**:

$$E \quad \mapsto \quad \text{trop}(E)$$



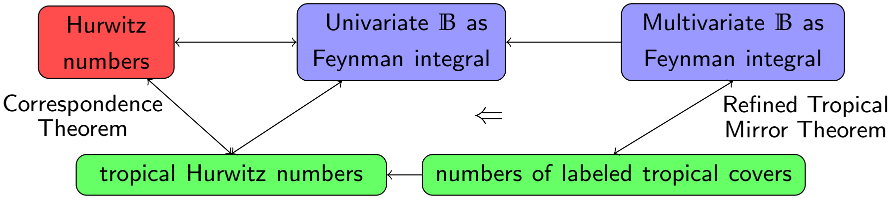




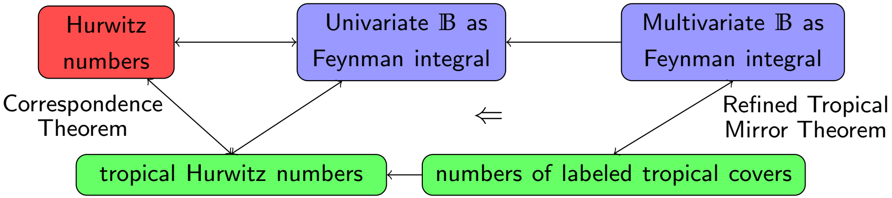
For $X = \mathbb{P}^2$ (building block of C-Y) and $g = 0$:

- tropical mirror theorem (Gross '10)
- partial correspondence theorem (Markwig-Rau '09, Mikhalkin '05)

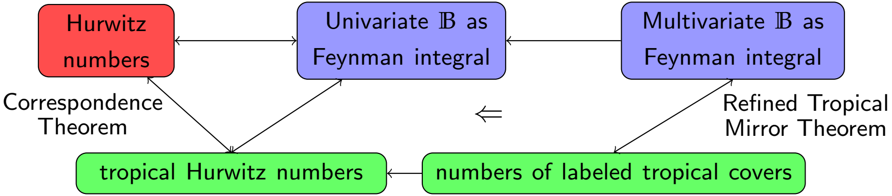
Tropical Mirror Symmetry



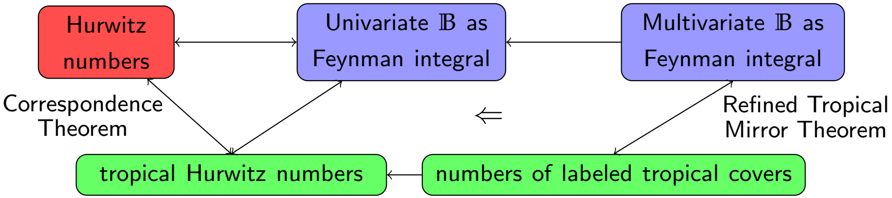
Tropical Mirror Symmetry



- Correspondence theorem for all g and d.



- Correspondence theorem for all g and d.
- Tropical mirror theorem for all g as corollary to



- Correspondence theorem for all g and d.
- Tropical mirror theorem for all g as corollary to
- refined tropical mirror theorem for each trivalent connected graph of genus g and branch type.
- Computation of refined Feynman integrals.

Definition

A **Feynman graph** is a 3-valent, connected graph Γ of genus g.

Feynman integrals (B-side)

Definition

A **Feynman graph** is a 3-valent, connected graph Γ of genus g.

By $g(\Gamma) = 1 - |\text{vert}(\Gamma)| + |\text{edges}(\Gamma)|$ and $3\,|\text{vert}(\Gamma)| = 2\,|\text{edges}(\Gamma)|$

$$|\text{vert}(\Gamma)| = 2g - 2 \qquad |\text{edges}(\Gamma)| = 3g - 3$$

Feynman integrals (B-side)

Definition

A **Feynman graph** is a 3-valent, connected graph Γ of genus g.

By $g(\Gamma) = 1 - |\text{vert}(\Gamma)| + |\text{edges}(\Gamma)|$ and $3\,|\text{vert}(\Gamma)| = 2\,|\text{edges}(\Gamma)|$

$$|\text{vert}(\Gamma)| = 2g - 2 \qquad |\text{edges}(\Gamma)| = 3g - 3$$

Fix labeling z_i for vertices and q_i for edges.

Feynman integrals (B-side)

Definition

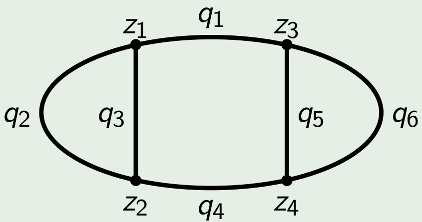
A **Feynman graph** is a 3-valent, connected graph Γ of genus g.

By $g(\Gamma) = 1 - |\text{vert}(\Gamma)| + |\text{edges}(\Gamma)|$ and $3\,|\text{vert}(\Gamma)| = 2\,|\text{edges}(\Gamma)|$

$$|\text{vert}(\Gamma)| = 2g - 2 \qquad |\text{edges}(\Gamma)| = 3g - 3$$

Fix labeling z_i for vertices and q_i for edges.

Example



Feynman integrals (B-side)

Definition (Propagator)

$$P(z, q) = -\frac{1}{4\pi^2}\wp(z, q) - \frac{1}{12}E_2(q) \qquad \text{for } z \in E = \mathbb{C}/\Lambda$$

Definition (Propagator)

$$P(z,q) = -\frac{1}{4\pi^2}\wp(z,q) - \frac{1}{12}E_2(q) \qquad \text{for } z \in E = \mathbb{C}/\Lambda$$

with Weierstraß-$\wp$-function $\wp = \frac{1}{z^2} + \ldots$ and the Eisenstein series

$$E_2 = 1 - 24\sum_{d=1}^{\infty}\sigma_1(d)q^{2d} = 1 - 24q^2 - 72q^4 - \ldots \qquad \sigma_1(d) = \sum_{m|d} m$$

Feynman integrals (B-side)

Definition (Propagator)

$$P(z,q) = -\frac{1}{4\pi^2}\wp(z,q) - \frac{1}{12}E_2(q) \qquad \text{for } z \in E = \mathbb{C}/\Lambda$$

with Weierstraß-$\wp$-function $\wp = \frac{1}{z^2} + \ldots$ and the Eisenstein series

$$E_2 = 1 - 24\sum_{d=1}^{\infty}\sigma_1(d)q^{2d} = 1 - 24q^2 - 72q^4 - \ldots \qquad \sigma_1(d) = \sum_{m|d} m$$

Definition (Feynman integral)

For ordering $\Omega \in S_{2g-2}$ of integration paths on E

$$I_{\Gamma,\Omega} = \int_{\gamma_{2g-2}} \ldots \int_{\gamma_1} \left(\prod_{e\in \text{edges}(\Gamma)} P(z_e^+ - z_e^-, q) \right) dz_{\Omega(1)} \ldots dz_{\Omega(2g-2)}$$

As a direct generalization of (Cavalieri-Johnson-Markwig '10) and (Bertrand-Brugallé-Mikhalkin '11) obtain correspondence theorem:

Correspondence Theorem

As a direct generalization of (Cavalieri-Johnson-Markwig '10) and (Bertrand-Brugallé-Mikhalkin '11) obtain correspondence theorem:

Theorem (BBBM '15)

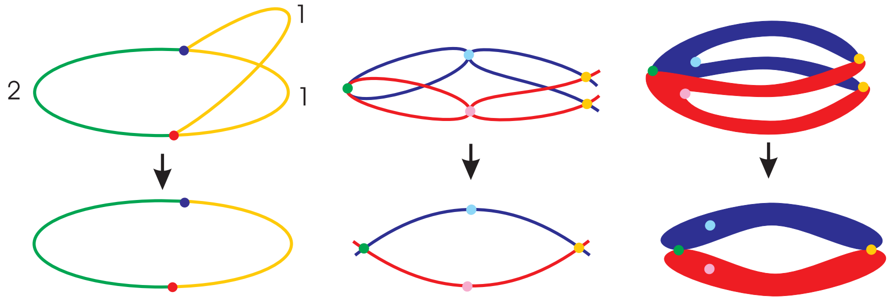
$N_{d,g} = N_{d,g}^{\text{trop}}$ *by correspondence of tropical and algebraic covers.*

Correspondence Theorem

As a direct generalization of (Cavalieri-Johnson-Markwig '10) and (Bertrand-Brugallé-Mikhalkin '11) obtain correspondence theorem:

Theorem (BBBM '15)

$N_{d,g} = N^{\text{trop}}_{d,g}$ *by correspondence of tropical and algebraic covers.*

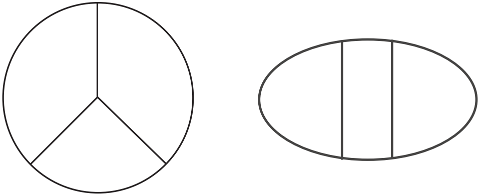


Tropical Hurwitz numbers – Example

$$N_{3,3}^{trop} = ?$$

$$N_{3,3}^{trop} = ?$$

Two trivalent, connected combinatorial types (non-metric graphs)

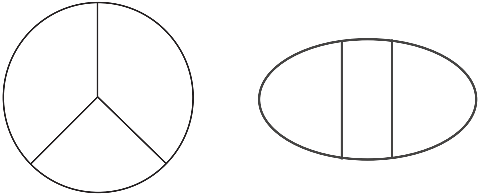


of genus $g = 3$ with

- $2g - 2 = 4$ vertices
- $3g - 3 = 6$ edges
- no bridges

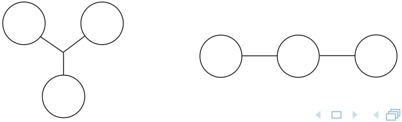
$$N_{3,3}^{trop} = ?$$

Two trivalent, connected combinatorial types (non-metric graphs)



of genus $g = 3$ with

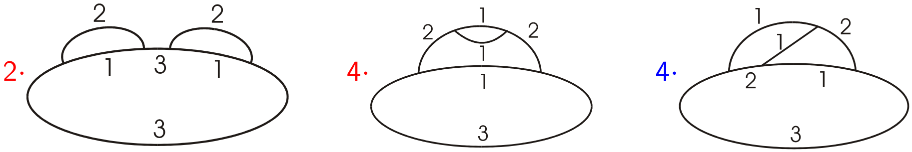
- $2g - 2 = 4$ vertices
- $3g - 3 = 6$ edges
- no bridges (weight 0 edges would be contracted):



Tropical Hurwitz numbers – Example

$$N_{3,3}^{trop} =$$

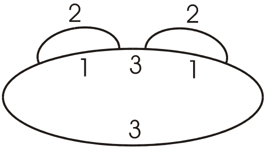
$$N^{trop}_{3,3} =$$



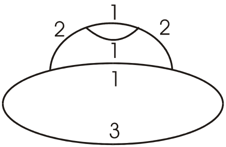
$\text{mult}(\pi) = 2^2 \cdot 3^2 = 36$

$$N^{trop}_{3,3} =$$

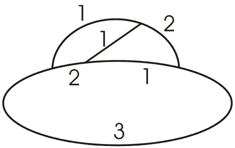
$2\cdot$



$4\cdot$



$4\cdot$

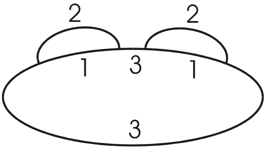


$\text{mult}(\pi) = 2^2 \cdot 3^2 = 36$ $\quad$ $\text{mult}(\pi) = \frac{1}{2} \cdot 2^2 \cdot 3 = 6$

Tropical Hurwitz numbers – Example

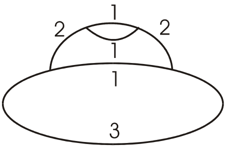
$$N_{3,3}^{trop} =$$

$2\cdot$



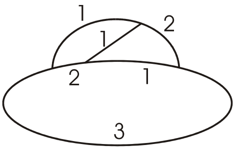
$\text{mult}(\pi) = 2^2 \cdot 3^2 = 36$

$4\cdot$



$\text{mult}(\pi) = \frac{1}{2} \cdot 2^2 \cdot 3 = 6$

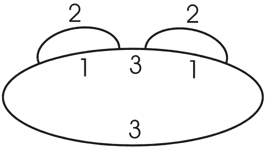
$4\cdot$



$\text{mult}(\pi) = 2^2 \cdot 3 = 12$

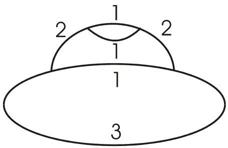
$$N_{3,3}^{trop} =$$

2·



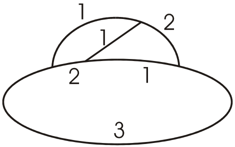
$\text{mult}(\pi) = 2^2 \cdot 3^2 = 36$

4·



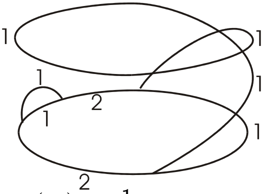
$\text{mult}(\pi) = \frac{1}{2} \cdot 2^2 \cdot 3 = 6$

4·



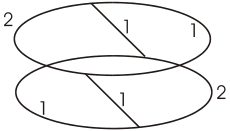
$\text{mult}(\pi) = 2^2 \cdot 3 = 12$

4·



$\text{mult}(\pi) = \frac{1}{2} \cdot 2 \cdot 2 = 2$

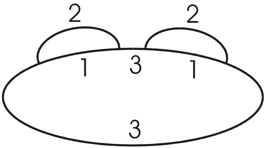
2·



Tropical Hurwitz numbers – Example

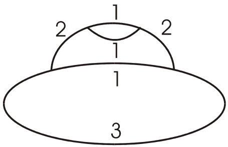
$$N_{3,3}^{trop} =$$

$2\cdot$



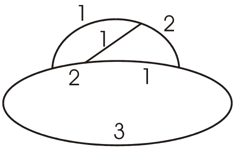
$\mathrm{mult}(\pi) = 2^2 \cdot 3^2 = 36$

$4\cdot$



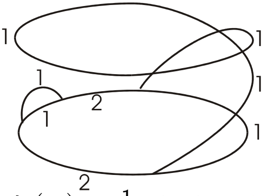
$\mathrm{mult}(\pi) = \frac{1}{2} \cdot 2^2 \cdot 3 = 6$

$4\cdot$



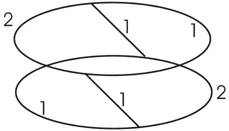
$\mathrm{mult}(\pi) = 2^2 \cdot 3 = 12$

$4\cdot$



$\mathrm{mult}(\pi) = \frac{1}{2} \cdot 2 \cdot 2 = 2$

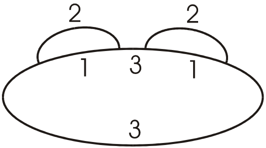
$2\cdot$



$\mathrm{mult}(\pi) = 2^2 = 4$

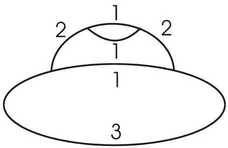
$$N_{3,3}^{trop} = 112 + 48 = 160$$

$2\cdot$



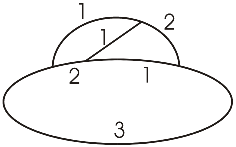
$\text{mult}(\pi) = 2^2 \cdot 3^2 = 36$

$4\cdot$



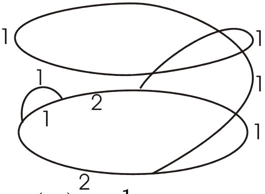
$\text{mult}(\pi) = \frac{1}{2} \cdot 2^2 \cdot 3 = 6$

$4\cdot$



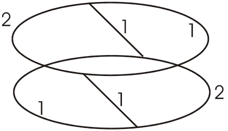
$\text{mult}(\pi) = 2^2 \cdot 3 = 12$

$4\cdot$



$\text{mult}(\pi) = \frac{1}{2} \cdot 2 \cdot 2 = 2$

$2\cdot$



$\text{mult}(\pi) = 2^2 = 4$

Refined Feynman integrals

Definition (Refined Feynman integrals)

$$I_{\Gamma,\Omega}(q_1, ..., q_{3g-3}) = \int_{\gamma_{2g-2}} ... \int_{\gamma_1} \left(\prod_{k=1}^{3g-3} P(z_k^+ - z_k^-, q_k) \right) dz_{\Omega(1)} ... dz_{\Omega(2g-2)}$$

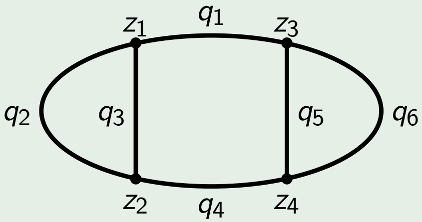
Refined Feynman integrals

Definition (Refined Feynman integrals)

$$I_{\Gamma,\Omega}(q_1,...,q_{3g-3}) = \int_{\gamma_{2g-2}} \cdots \int_{\gamma_1} \left(\prod_{k=1}^{3g-3} P(z_k^+ - z_k^-, q_k) \right) dz_{\Omega(1)} \ldots dz_{\Omega(2g-2)}$$

Example

For



we have to integrate

$$P(z_1 - z_2, q_1) \cdot P(z_1 - z_2, q_2) \cdot P(z_1 - z_3, q_3) \cdot P(z_2 - z_4, q_4) \cdot P(z_3 - z_4, q_5) \cdot P(z_3 - z_4, q_6)$$

Theorem (Multivariate tropical mirror theorem, BBBM '13)

$$\sum_{\underline{a}} N^{trop}_{\underline{a},\Gamma,\Omega}\; q^{2\underline{a}} = I_{\Gamma,\Omega}(q_1, \ldots, q_{3g-3})$$

Tropical mirror theorem

Theorem (Multivariate tropical mirror theorem, BBBM '13)

$$\sum_{\underline{a}} N^{trop}_{\underline{a},\Gamma,\Omega}\ q^{2\underline{a}} = I_{\Gamma,\Omega}(q_1, \ldots, q_{3g-3})$$

Setting $q_i = q$ we get (using the action of $\mathrm{Aut}(\Gamma)$ on labeled covers):

Corollary (Tropical mirror theorem)

$$\sum_{d} N^{trop}_{d,g}\ q^{2d} = \sum_{\Gamma} \frac{1}{|\mathrm{Aut}(\Gamma)|} \sum_{\Omega} I_{\Gamma,\Omega}(q)$$

Tropical mirror theorem

Theorem (Multivariate tropical mirror theorem, BBBM '13)

$$\sum_{\underline{a}} N^{trop}_{\underline{a},\Gamma,\Omega}\ q^{2\underline{a}} = I_{\Gamma,\Omega}(q_1,\ldots,q_{3g-3})$$

Setting $q_i = q$ we get (using the action of $\mathrm{Aut}(\Gamma)$ on labeled covers):

Corollary (Tropical mirror theorem)

$$\sum_{d} N^{trop}_{d,g}\ q^{2d} = \sum_{\Gamma} \frac{1}{|\mathrm{Aut}(\Gamma)|} \sum_{\Omega} I_{\Gamma,\Omega}(q)$$

Together with the correspondence theorem this proves:

Corollary (Mirror symmetry for elliptic curves)

For elliptic curves $\mathbb{A}_g = \mathbb{B}_g$ *for all* g*.*

By coordinate change $x_k = \exp(i\pi z_k)$,

Computing Feynman integrals

By coordinate change $x_k = \exp(i\pi z_k)$, path γ_k becomes circle around 0,

Computing Feynman integrals

By coordinate change $x_k = \exp(i\pi z_k)$, path γ_k becomes circle around 0, factor $\frac{1}{x_k}$,

Computing Feynman integrals

By coordinate change $x_k = \exp(i\pi z_k)$, path γ_k becomes circle around 0, factor $\frac{1}{x_k}$, integral becomes residue, By coordinate change $x_k = \exp(i\pi z_k)$, path γ_k becomes circle around 0, factor $\frac{1}{x_k}$, integral becomes residue, difference becomes quotient.

Proposition (BBBM '15)

$$P(x,q) = \frac{x^2}{(x^2-1)^2} + \sum_{a=1}^{\infty} \sum_{w|a} w(x^{2w} + x^{-2w})q^{2a}$$

By coordinate change $x_k = \exp(i\pi z_k)$, path γ_k becomes circle around 0, factor $\frac{1}{x_k}$, integral becomes residue, difference becomes quotient.

Proposition (BBBM '15)

$$P(x,q) = \frac{x^2}{(x^2-1)^2} + \sum_{a=1}^{\infty} \sum_{w|a} w(x^{2w} + x^{-2w})q^{2a}$$

$$P_a(x,y) := \begin{cases} \frac{x^2y^2}{(x^2-y^2)^2} & \text{for } a = 0 \\ \sum_{w|a} w \frac{x^{4w}+y^{4w}}{(xy)^{2w}} & \text{for } a > 0 \end{cases}$$

Computing Feynman integrals

By coordinate change $x_k = \exp(i\pi z_k)$, path γ_k becomes circle around 0, factor $\frac{1}{x_k}$, integral becomes residue, difference becomes quotient.

Proposition (BBBM '15)

$$P(x,q) = \frac{x^2}{(x^2-1)^2} + \sum_{a=1}^{\infty}\sum_{w|a} w(x^{2w}+x^{-2w})q^{2a}$$

$$P_a(x,y) := \begin{cases} \frac{x^2y^2}{(x^2-y^2)^2} & \text{for } a=0 \\ \sum_{w|a} w\frac{x^{4w}+y^{4w}}{(xy)^{2w}} & \text{for } a>0 \end{cases}$$

Theorem (BBBM '15)

$$N^{trop}_{\underline{a},\Gamma,\Omega} = \mathrm{const}_{x_{\Omega(2g-2)}} \ldots \mathrm{const}_{x_{\Omega(1)}} \prod_{k=1}^{3g-3} P_{a_k}(x_k^+, x_k^-)$$

Sketch of Proof

$$\{\text{labeled tropical covers}\} \overset{1:1}{\leftrightarrows} \{\text{constant products of Laurent monomials}\}$$

Sketch of Proof

$$\{\text{labeled tropical covers}\} \overset{1:1}{\leftrightarrows} \{\text{constant products of Laurent monomials}\}$$

- order the vertices according to Ω,

Sketch of Proof

$$\{\text{labeled tropical covers}\} \overset{1:1}{\leftrightarrows} \{\text{constant products of Laurent monomials}\}$$

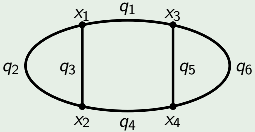
- order the vertices according to Ω,
- associate edges of weight a_i to Laurent monomials.

Sketch of Proof

$$\{\text{labeled tropical covers}\} \overset{1:1}{\leftrightarrows} \{\text{constant products of Laurent monomials}\}$$

- order the vertices according to Ω,
- associate edges of weight a_i to Laurent monomials.

Example



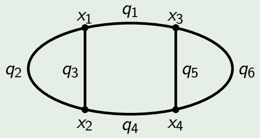
$x_1 < x_3 < x_4 < x_2$

Sketch of Proof

$$\{\text{labeled tropical covers}\} \overset{1:1}{\leftrightarrows} \{\text{constant products of Laurent monomials}\}$$

- order the vertices according to Ω,
- associate edges of weight a_i to Laurent monomials.

Example



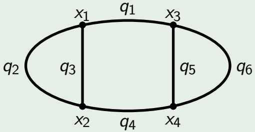
$x_1 < x_3 < x_4 < x_2 \qquad \underline{a} = (0, 2, 2, 0, 1, 0)$

Sketch of Proof

$$\{\text{labeled tropical covers}\} \overset{1:1}{\leftrightarrows} \{\text{constant products of Laurent monomials}\}$$

- order the vertices according to Ω,
- associate edges of weight a_i to Laurent monomials.

Example



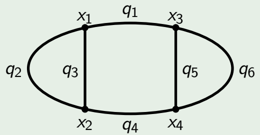
$x_1 < x_3 < x_4 < x_2 \qquad \underline{a} = (0, 2, 2, 0, 1, 0)$

$$\left(\frac{x_1}{x_3}\right)^2 \cdot 2 \cdot \left(\frac{x_2}{x_1}\right)^{2\cdot 2} \cdot \left(\frac{x_1}{x_2}\right)^2 \cdot \left(\frac{x_4}{x_2}\right)^2 \cdot \left(\frac{x_4}{x_3}\right)^2 \cdot 2 \cdot \left(\frac{x_3}{x_4}\right)^{2\cdot 2}$$

$$\{\text{labeled tropical covers}\} \overset{1:1}{\leftrightarrows} \{\text{constant products of Laurent monomials}\}$$

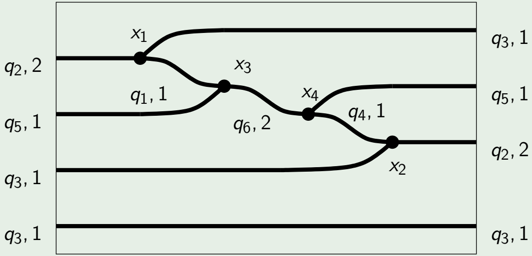
- order the vertices according to Ω,
- associate edges of weight a_i to Laurent monomials.

Example



$x_1 < x_3 < x_4 < x_2 \qquad \underline{a} = (0, 2, 2, 0, 1, 0)$

$$\left(\frac{x_1}{x_3}\right)^2 \cdot 2 \cdot \left(\frac{x_2}{x_1}\right)^{2\cdot 2} \cdot \left(\frac{x_1}{x_2}\right)^2 \cdot \left(\frac{x_4}{x_2}\right)^2 \cdot \left(\frac{x_4}{x_3}\right)^2 \cdot 2 \cdot \left(\frac{x_3}{x_4}\right)^{2\cdot 2}$$



References

- J. Böhm, K. Bringmann, A. Buchholz, H. Markwig, *Tropical mirror symmetry for elliptic curves*, J. Reine Angew. Math. (2015).
- J. Böhm, K. Bringmann, A. Buchholz, H. Markwig, *ellipticcovers.lib. A Singular 4 library for Gromov-Witten invariants of elliptic curves*, SINGULAR distribution.

- A. Okounkov, R. Pandharipande, *Gromov-Witten theory, Hurwitz theory and completed cycles*, Ann. Math. 163 (2006).
- R. Dijkgraaf, *Mirror symmetry and elliptic curves*, in Progr. Math. 129 (1995).
- M. Gross, *Mirror symmetry for* $\mathbb{P}^2$ *and tropical geometry*, Adv. Math. 224 (2010).

- B. Bertrand, E. Brugallé, G. Mikhalkin, *Tropical open Hurwitz numbers*, Rend. Semin. Mat. Univ. Padova 125 (2011).