Computational Algebraic Geometry

Algebraische Geometrie auf dem Computer

Teil 2

Joachim Wehler

Ludwig-Maximilians-Universität München, Sommersemester 2005

Version 2.0

6 Projektive Varietäten

Affine Varietäten sind die ersten anschaulich gegebenen Objekte der Algebraischen Geometrie. Allerdings gelten global für affine Varietäten und ihre Morphismen nicht die einfachsten Aussagen. Es sind an verschiedenen Stellen Zusatzvoraussetzungen oder Fallunterscheidungen nötig.

Beispiele:

- Das Bild einer regulären Abbildung zwischen zwei affinen Varietäten ist wieder eine affine Varietät unter der Zusatzvoraussetzung der Endlichkeit der Abbildung.
- Zwei verschiedene Geraden der affinen Ebene sind entweder parallel, oder sie schneiden sich in genau einem Punkt.

Durch Abschließung des affinen Raumes zum projektiven Raum werden viele globale Sätze allgemeiner, Fallunterscheidungen werden überflüssig:

- Das Bild einer regulären Abbildung zwischen projektiven Varietäten ist wieder eine projektive Varietät.
- Zwei verschiedene Geraden in der projektiven Ebene schneiden sich in genau einem Punkt – im Falle paralleler Geraden der affinen Ebene ist der Schnittpunkt ein unendlich ferner Punkt.

In diesem Kapitel sei k ein Körper und $K = \bar{k}$ sein algebraischer Abschluß.

Projektive Varietäten und reguläre Abbildungen

Der projektive Raum und projektive Varietäten lassen sich als Vervollständigung des affinen Raums bzw. von affinen Varietäten durch „fehlende“ unendlich ferne Punkte auffassen. Sie ähneln damit den kompakten Räumen in der Topologie. Da die Zariski Topologie jedoch keine Hausdorff Topologie ist, handelt es sich nur um eine Analogie.

6.1 Definition (Projektiver Raum)

Der *projektive Raum* $\boldsymbol{P}^n = \boldsymbol{P}^n(K)$ ist der Quotient des punktierten affinen Raums $\boldsymbol{A}^{n+1}(K) - 0$ nach der Äquivalenzrelation „liegt auf derselben Geraden durch den Nullpunkt“:

$$\boldsymbol{P}^n(K) := \left(\boldsymbol{A}^{n+1}(K) - 0 \right) / \sim$$

mit

$$x \sim y \text{ für } x, y \in \boldsymbol{A}^{n+1}(K) - 0 :<=> \exists \lambda \in K^* \text{ mit } x = \lambda \cdot y \,.$$

6.2 Bemerkung (Projektiver Raum)

i) Die Äquivalenzklasse eines Punktes $x \in \boldsymbol{A}^{n+1} - 0$ ist die Menge aller von Null verschiedener Punkte auf der Geraden in $\boldsymbol{A}^{n+1}$, die durch x und durch den Nullpunkt geht. Die Punkte des projektiven Raumes $\boldsymbol{P}^n$ entsprechen also bijektiv den Geraden durch den Nullpunkt des affinen Raumes $\boldsymbol{A}^{n+1}$.

Die Äquivalenzklasse

$$p = [\ x\ ] = [\ (x_0, x_1, ..., x_n)\ ] \in \boldsymbol{P}^n$$

bezeichnet man mit

$$p = (x_0 : x_1 : ... : x_n)$$

und nennt ein zugehöriges Tupel $(x_0, x_1, ..., x_n)$ *homogene Koordinaten* des Punktes p. Die homogenen Koordinaten eines Punktes sind nur bis auf einen gemeinsamen Faktor $\lambda \in K^*$ eindeutig bestimmt.

ii) Als Menge ist der projektive Raum $\boldsymbol{P}^n$ in natürlicher Weise die disjunkte Vereinigung des affinen Raumes

$$\boldsymbol{A}^n \xrightarrow{\cong} \{\ (x_0 : x_1 : x_2 : ... : x_n) \in \boldsymbol{P}^n : x_0 \neq 0\ \},\ (x_1, x_2, ..., x_n) \mapsto (1 : x_1 : x_2 : ... : x_n)$$

und eines niederdimensionalen projektiven Raumes

$$\boldsymbol{P}^{n-1} \xrightarrow{\cong} \{\ (x_0 : x_1 : x_2 : ... : x_n) \in \boldsymbol{P}^n : x_0 = 0\ \},\ (x_0 : x_1 : ... : x_{n-1}) \mapsto (0 : x_0 : x_1 : ... : x_{n-1})$$

d.h.

$$\boldsymbol{P}^n = \boldsymbol{A}^n \dot{\cup} \boldsymbol{P}^{n-1}.$$

Insbesondere entsteht die projektive Gerade $\boldsymbol{P}^1$ aus der affinen Geraden $\boldsymbol{A}^1$ durch Hinzunahme eines einzigen Punktes

$$\boldsymbol{P}^0 = \{\ (0:1)\ \}.$$

Aus Sicht der affinen Geraden ist dieser Punkt der unendlich ferne Punkt

$$\infty = (0:1) \in \boldsymbol{P}^1.$$

Analog ist die projektive Ebene $\boldsymbol{P}^2$ die Vervollständigung der affinen Ebene $\boldsymbol{A}^2$ durch die unendlich ferne projektive Gerade

$$\boldsymbol{P}^1 \cong \{\ (z_0 : z_1 : z_2) \in \boldsymbol{P}^2 : z_0 = 0\ \}.$$

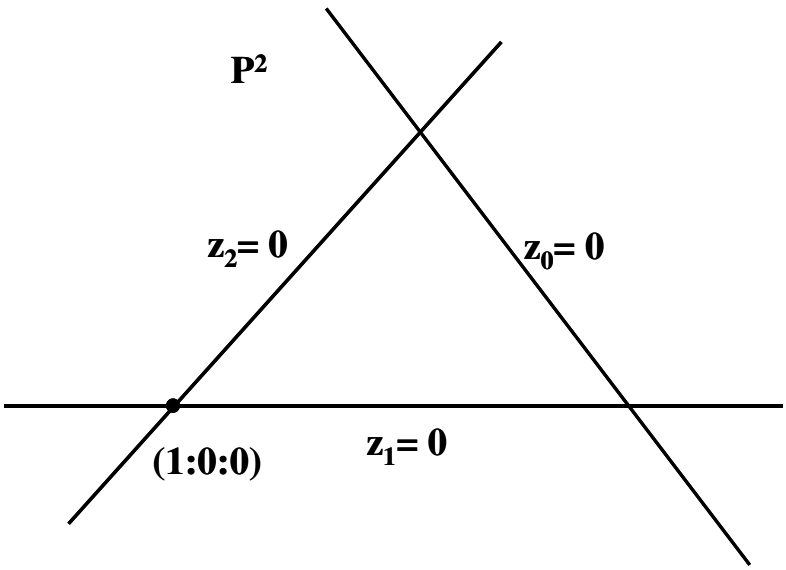

Abbildung 1: Projektive Ebene mit unendlich ferner Geraden

Polynome lassen sich i.a. nicht als Funktionen auf dem projektiven Raum auffassen, da ihr Funktionswert in einem gegebenem Punkt des projektiven Raumes von dem gewählten Repräsentanten des Punktes abhinge.

Die richtige Teilklasse für projektive Räume sind homogene Polynome. Auch ein homogenes Polynom definiert allerdings keine Funktion auf dem projektiven Raum. Es läßt sich jedoch entscheiden, ob ein homogenes Polynom in einem Punkt des projektiven Raumes verschwindet oder nicht. Daher haben homogene Polynome ein wohldefiniertes Nullstellengebilde im projektiven Raum, sie definieren projektive Varietäten.

6.3 **Bemerkung** *(Homogenes Polynom, homogenes Ideal)*

i) Im Falle eines unendlichen Körpers k gilt die Äquivalenz: Ein Polynom ist genau dann homogen vom Grad d, wenn für alle $x \in k^{n+1}$ und für alle $\lambda \in k^*$ gilt:

$$f(\lambda \cdot x) = \lambda^d \cdot f(x).$$

ii) Jedes Polynom

$$f \in k[T_0, T_1, ..., T_n]$$

hat eine eindeutige Summendarstellung

$$f = \sum_{i=0}^{d} f_i$$

mit homogenen Polynomen

$$0 \neq f_i \in k[T_0, T_1, ..., T_n]$$

des Grades i, seinen *homogenen Komponenten* vom Grad i. Das Polynom f gehört genau dann zu einem homogenen Ideal

$$I \subset k[T_0, T_1, ..., T_n],$$

wenn jede seiner homogenen Komponenten zu I gehört.

iii) Die reduzierte Gröbner Basis eines homogenen Ideals besteht aus homogenen Polynomen.

iv) Das Radikal eines homogenen Ideals ist wieder ein homogenes Ideal.

Beweis. ad iii) siehe [CLO1997], Chap.7, §3, Theor. 2.

6.4 **Definition** *(Projektive Varietät)*

Eine *projektive k-Varietät* X ist das gemeinsame Nullstellengebilde einer Familie homogener Polynome

$$f_j \in k[T_0, T_1, ..., T_n],\ j \in J,$$

im projektiven Raum $\boldsymbol{P}^n(K)$. Die definierenden Polynome dürfen unterschiedliche Grade haben. Für die Varietät als Teilmenge von $\boldsymbol{P}^n(K)$ schreibt man

$$X = \{(x_0 : ... : x_n) \in \boldsymbol{P}^n(K) : f_j(x_0, ..., x_n) = 0 \text{ für alle } j \in J\}.$$

Da alle Polynome f_j homogen sind, gilt die Eigenschaft

$$f_j(x_0, x_1, ..., x_n) = 0,\ j \in J\ ,$$

unabhängig vom gewählten Repräsentanten des Punktes $p = [\,(x_0, x_1, ..., x_n)\,]$.

6.5 **Bemerkung** *(Projektive Varietät und homogenes Ideal)*

i) Einem homogenen Ideal

$$I \subset k[T_0, T_1, ..., T_n]$$

wird die projektive Varietät zugeordnet

$$Var(I) := \{\,(x_0 : ... : x_n) \in \boldsymbol{P}^n : f(x_0, ..., x_n) = 0 \text{ für alle homogenen } f \in I\,\}.$$

Dabei definiert neben dem Einheitsideal

$$\langle\, 1 \,\rangle \subset k[T_0, T_1, ..., T_n]$$

auch schon das *irrelevante* homogene Ideal

$$m := \langle\, T_0, T_1, ..., T_n \,\rangle \subset k[T_0, T_1, ..., T_n]$$

die leere projektive Varietät

$$\varnothing = Var(m) \subset \boldsymbol{P}^n\ .$$

ii) Einer projektiven Varietät $X \subset \boldsymbol{P}^n$ wird als homogenes Ideal ihr *Verschwindungsideal* zugeordnet

$$Id(X) := \langle f \in k[T_0, ..., T_n] : f \text{ homogen und } f(x_0, ..., x_n) = 0 \text{ für alle } (\,x_0 : ... : x_n\,) \in X \rangle\ .$$

Wegen der Noether-Eigenschaft des Polynomringes läßt sich die Varietät eines homogenen Ideals bereits durch endlich viele homogene Polynome definieren.

6.6 **Definition** *(Homogener Koordinatenring)*

Der *homogene Koordinatenring* einer projektiven k-Varietät $X \subset \boldsymbol{P}^n(K)$ ist die graduierte k-Algebra

$$S(X) := k[T_0, T_1, ..., T_n] / Id(X)$$

mit der vom Polynomring $k[T_0, T_1, ..., T_n]$ induzierten Graduierung.

Über einem algebraisch-abgeschlossenen Koordinatenkörper folgt aus dem Hilbertschen Nullstellensatz und der Tatsache, daß das Radikal eines homogenen Ideals wieder homogen ist, ähnlich wie im affinen Fall die Charakterisierung projektiver Varietäten durch ihr Verschwindungsideal:

6.7 **Satz** *(Projektive Varietäten und homogene Ideale)*

Die beiden folgenden Zuordnungen sind zueinander invers:

$$\left\{\, I \subset k[T_0, ..., T_n] : \text{homogen}, I \underset{\neq}{\subset} m, \sqrt{I} = I \,\right\} \underset{Id}{\overset{Var}{\rightleftarrows}} \left\{\, X \subset \boldsymbol{P}^n : \text{projektive } k-\text{Varietät} \,\right\}$$

Es gilt

$$Var\left(\sum_j I_j\right) = \bigcap_j Var(I_j),\ j \in J \text{ beliebig},$$

$$\text{und } Var\left(\bigcap_j I_j\right) = \bigcup_j Var(I_j),\ j \in J \text{ endlich}$$

Beweis. siehe [CLO1997], Chap. 8, §3, Theor. 10 und Ex. 7.

6.8 Definition (Zariski Topologie)

Die *k-Topologie* des projektiven Raumes $\boldsymbol{P}^n(K)$ ist die eindeutig bestimmte Topologie mit den projektiven k-Varietäten des $\boldsymbol{P}^n(K)$ als abgeschlossenen Mengen.

Die Zariski-Topologie einer projektiven k-Varietät $X \subset \boldsymbol{P}^n(K)$ ist die induzierte Unterraumtopologie: Genau die Mengen der Form

$$X \cap A \text{ mit einer Zariski-abgeschlossenen Teilmenge } A \subset \boldsymbol{P}^n(K)$$

sind nach Definition die abgeschlossenen Mengen von X.

Offene Teilmengen einer projektiven Varietät heißen *quasi-projektiv*.

Eine reguläre Abbildung zwischen zwei projektiven k-Varietäten ist eine Abbildung, die *lokal* durch homogene Polynome ohne gemeinsame Nullstelle gegeben werden kann. Anders als im affinen Falle fordert man im projektiven Falle nicht die globale Gültigkeit der Polynomdarstellung.

6.9 Definition (Reguläre Abbildung)

Eine *k-reguläre Abbildung* zwischen zwei projektiven k-Varietäten $X \subset \boldsymbol{P}^n$ und $Y \subset \boldsymbol{P}^m$ ist eine Abbildung

$$g : X \longrightarrow Y,$$

mit folgender Eigenschaft: Zu jedem Punkt $p \in X$ existieren eine offene Umgebung U von p und $m+1$ homogene Polynome eines festen Grades d

$$g_i \in k[T_0, T_1, ..., T_n],\ \deg g_i = d,\ i = 0, 1, ..., m,$$

ohne gemeinsame Nullstelle in U mit

$$g(x_0 : ... : x_n) = (g_0(x_0, ..., x_n) : ... : g_m(x_0, ..., x_n)) \text{ für alle } (x_0 : ... : x_n) \in U.$$

6.10 Beispiel (Stereographische Projektion)

Die stereographische Projektion entspringt im Rellen, der Grundkörper ist $k = \boldsymbol{R}$. Statt über dem algebraischen Abschluß $K = \boldsymbol{C}$ werden wir in den ersten beiden Abschnitten dieses Beispiels die Nullstellengebilde als Teilmengen des reellen affinen Raumes

$$\boldsymbol{A}^{n+1} = \boldsymbol{A}^{n+1}(\boldsymbol{R}) := \boldsymbol{R}^{n+1}$$

bzw. des rellen projektiven Raumes

$$\boldsymbol{P}^n = \boldsymbol{P}^n(\boldsymbol{R}) := \left(\boldsymbol{R}^{n+1} - 0\right)/\sim$$

betrachten. Alle verwendeten Begriffe gelten analog auch für diesen Fall.

Bei der stereographischen Projektion des Einheitskreises auf eine Geraden handelt es sich um eine Abbildung, die auf einem offenen Teil einer affinen Varietät definiert ist und zu einer regulären Abbildung zwischen den zugehörigen projektiven Varietäten fortgesetzt werden kann.

i) Es sei

$$C := \left\{ (x, y) \in \boldsymbol{A}^2 : 1 = x^2 + y^2 \right\}$$

der reelle Einheitskreis mit einem ausgezeichneten Punkt

$$N := (0, 1) \in C .$$

Wir definieren als die stereographische Projektion des Kreises C vom Punkt N auf die x-Achse $\boldsymbol{A}^1$ die Abbildung

$$g : C - N \longrightarrow \boldsymbol{A}^1, (x, y) \mapsto \frac{x}{1-y} .$$

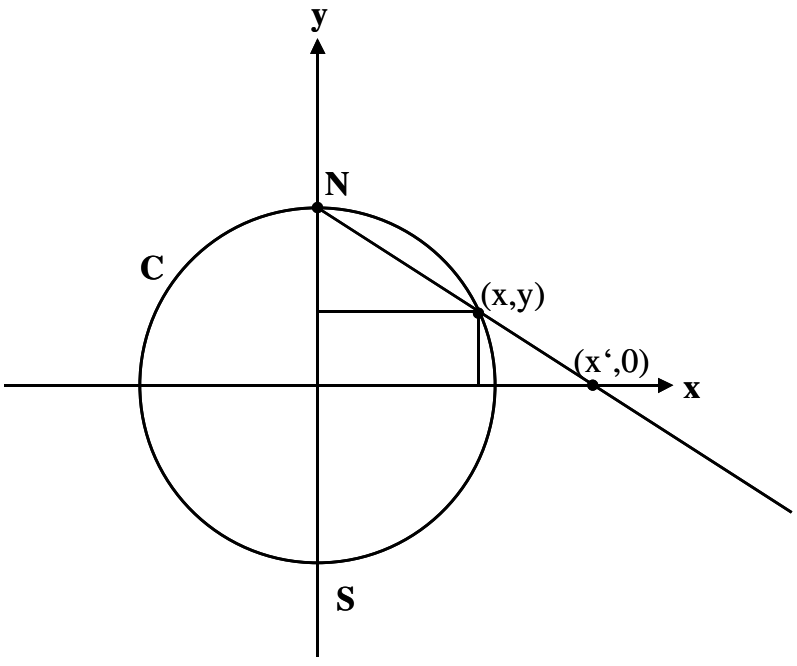

Abbildung 2: Stereographische Projektion

Nach dem Strahlensatz, siehe Abbildung 2, gilt

$$\frac{1-y}{1} = \frac{x}{x'},$$

also

$$x' = \frac{x}{1-y} \text{ und } x' = \frac{x}{1-y} = \frac{1+y}{x} \; \textit{für} \; (x, y) \neq S .$$

Der Funktionswert der Umkehrabbildung

$$h : \boldsymbol{A}^1 \longrightarrow C - N,\ x' \mapsto (x, y)$$

berechnet sich aus

$$x = x' \cdot (\,1 - y\,) = x' \cdot \left(1 - \sqrt{1 - x^2}\,\right)$$

als

$$x = \frac{2 \cdot x'}{x'^2 + 1},\ y = \frac{x'^2 - 1}{x'^2 + 1}.$$

ii) Die bisher im Affinen definierte stereographische Projektion läßt sich fortsetzen zu einer regulären Abbildung zwischen projektiven Varietäten. Gesucht werden eine projektive Varietät

$$Q \subset \boldsymbol{P}^2$$

und eine reguläre Abbildung zwischen projektiven Varietäten

$$G : Q \longrightarrow \boldsymbol{P}^1,$$

so daß folgendes Diagramm kommutiert:

$$\begin{array}{ccccc} \boldsymbol{A}^2 & \xleftarrow{\supset} & C - N & \xrightarrow{g} & \boldsymbol{A}^1 \\ \downarrow\subset & & \downarrow\subset & & \downarrow\subset \\ \boldsymbol{P}^2 & \xleftarrow{\supset} & Q & \xrightarrow{G} & \boldsymbol{P}^1 \end{array}$$

Dazu fassen wir im Definitionsbereich von G die affine Ebene als Teilmenge der projektiven Ebene auf

$$\boldsymbol{A}^2 \xrightarrow{\subset} \boldsymbol{P}^2,\ (x, y) \mapsto (1 : x : y)$$

und identifizieren den affinen Einheitskreis $C \subset \boldsymbol{A}^2$ mit seinem Bild, der projektiven Quadrik

$$Q := \left\{ (x_0 : x_1 : x_2) \in \boldsymbol{P}^2 : x_0^2 = x_1^2 + x_2^2 \right\} \subset \boldsymbol{P}^2.$$

Im Wertebereich von G fassen wir die affine x-Achse als Teilmenge der projektiven Geraden auf

$$\boldsymbol{A}^1 \xrightarrow{\subset} \boldsymbol{P}^1,\ v \mapsto (1 : v).$$

Wie läßt sich die gegebene Abbildung g zu einer regulären Abbildung

$$G : Q \longrightarrow \boldsymbol{P}^1$$

fortsetzen? Heuristik: Die stereographische Projektion

$$g : C - N \longrightarrow \boldsymbol{A}^1,\ (x, y) \mapsto \frac{x}{1 - y}$$

schreibt sich bzgl. der üblichen Einbettungen

$$C - N \xrightarrow{\subset} Q \subset \boldsymbol{P}^2 \text{ und } \boldsymbol{A}^1 \xrightarrow{\subset} \boldsymbol{P}^1$$

als

$$g : C - N \longrightarrow \boldsymbol{A}^1, (1 : x_1 : x_2) \mapsto \left(1 : \frac{x_1}{1 - x_2}\right) = (1 - x_2 : x_1).$$

Diese Definition wird nun „homogenisiert“, d.h. in homogenen Koordinaten und unter Verwendung homogener Polynome geschrieben:

$$g : C - N \longrightarrow \boldsymbol{A}^1, (x_0 : x_1 : x_2) \mapsto (x_0 - x_2 : x_1).$$

In dieser Form läßt sie sich in den Punkt $N = (1 : 0 : 1)$ hinein jedoch nicht fortsetzen.

Gemäß Definition 6.9 überdecken wir den Definitionsbereich Q durch die beiden offenen Teilmengen

$$U_0 := Q - N = Q - (1 : 0 : 1) \text{ und } U_1 := Q - S = Q - (1 : 0 : -1)$$

und definieren jeweils lokal

$$G_0 : U_0 \longrightarrow \boldsymbol{P}^1, (x_0 : x_1 : x_2) \mapsto (x_0 - x_2 : x_1),\ G_1 : U_1 \longrightarrow \boldsymbol{P}^1, (x_0 : x_1 : x_2) \mapsto (x_1 : x_0 + x_2).$$

Hierdurch ist eine reguläre Abbildung

$$G : Q \longrightarrow \boldsymbol{P}^1$$

wohldefiniert. Denn im Durchschnitt

$$U_0 \cap U_1 \subset Q$$

gilt unter Benutzung der definierenden Gleichung $x_0^2 = x_1^2 + x_2^2$:

$$\frac{x_1}{x_0 - x_2} = \frac{x_1 \cdot (x_0 + x_2)}{(x_0 - x_2) \cdot (x_0 + x_2)} = \frac{x_1 \cdot (x_0 + x_2)}{x_0^2 - x_2^2} = \frac{x_1 \cdot (x_0 + x_2)}{x_1^2} = \frac{x_0 + x_2}{x_1}.$$

Insbesondere gilt

$$G(S) = G_0(S) = (1 : 0)$$

und

$$G(N) = G_1(N) = (0 : 1), \text{ unendlich ferner Punkt.}$$

Die reguläre Abbildung

$$G : Q \longrightarrow \boldsymbol{P}^1$$

ist ein Isomorphismus der Quadrik auf die projektive Gerade mit der Umkehrabbildung

$$H : \boldsymbol{P}^1 \longrightarrow Q, (u : v) \mapsto (v^2 + u^2 : 2uv : v^2 - u^2).$$

Denn es gilt:

$$G \circ H = id_{\boldsymbol{P}^1} \text{ und } H \circ G = id_Q.$$

iii) Im projektiven Zusammenhang ist die Definition der regulären Abbildungen G und H sowie der zugehörigen Varietäten nicht mehr an die rellen Zahlen gebunden. Beide Abbildungen lassen sich vielmehr durch die angegebenen Formeln für jeden Definitionskörper k, $char\ k \neq 2$, und die zugehörigen Varietäten mit Koordinatenkörper $K = \overline{k}$ definieren. Beide Abbildungen sind zueinander inverse reguläre Abbildungen. Sie stellen einen regulären Iso-

morphismus der Quadrik und der projektiven Gerade dar. Dennoch sind die homogenen Koordinatenringe beider Varietäten

$$S(\boldsymbol{P}^1) = k[T_0, T_1] \text{ und } S(Q) = k[T_0, T_1, T_2] \Big/ \left\langle T_0^2 - T_1^2 - T_2^2 \right\rangle$$

nicht isomorph.

6.11 Toolbeispiel (Reguläre Abbildung)

i) Die Projektivierung der affinen kubischen Kurve ist die projektive kubische Kurve *(twisted cubic curve)*

$$f : \boldsymbol{P}^1 \longrightarrow \boldsymbol{P}^3,$$

die global definiert ist durch die homogenen Polynome vom Grad 3

$$f(t_0 : t_1) := \left(t_0^3 : t_0^2 t_1 : t_0 t_1^2 : t_1^3 \right).$$

Denn die Einschränkung auf die affine Gerade

$$\boldsymbol{A}^1 = \left\{ (1 : t) \in \boldsymbol{P}^1 \right\}$$

ist die Abbildung

$$f \mid \boldsymbol{A}^1 \longrightarrow \boldsymbol{A}^3 \subset \boldsymbol{P}^3, (1 : t) \mapsto (1 : t : t^2 : t^3).$$

Zu den affinen Punkten kommt im Projektiven ein einziger weiterer Punkt hinzu:

$$f(0 : 1) := \left(0 : 0 : 0 : 1 \right).$$

Das Bild

$$Z := f\left(\boldsymbol{P}^1\right) \subset \boldsymbol{P}^3$$

wird als projektive Varietät

$$Z = Var(I) \subset \boldsymbol{P}^3$$

durch das homogene Ideal

$$I = \left\langle T_2^2 - T_1 T_3, T_1 T_2 - T_0 T_3, T_1^2 - T_0 T_2 \right\rangle = \left\langle T_1 T_2 - T_0 T_3, T_1^2 - T_2 T_0 \right\rangle \subset k[T_0, ..., T_3]$$

definiert. Dieses Ideal wird erzeugt von den 2x2-Minoren der Matrix

$$\begin{pmatrix} T_0 & T_1 & T_2 \\ T_1 & T_2 & T_3 \end{pmatrix}.$$

Die Einschränkung

$$f : \boldsymbol{P}^1 \longrightarrow Z \subset \boldsymbol{P}^3$$

ist ein regulärer Isomorphismus: Die Umkehrabbildung ist die reguläre Abbildung

$$g : Z \longrightarrow \boldsymbol{P}^1,$$

die auf der offenen Teilmenge

$$U_0 := \left\{ (x_0 : x_1 : x_2 : x_3) \in Z : x_0 \neq 0 \right\}$$

definiert ist als

$$g_0 : U_0 \longrightarrow \boldsymbol{P}^1, (x_0 : x_1 : x_2 : x_3) \mapsto (x_0 : x_1)$$

und auf der offenen Teilmenge

$$U_3 := \{ (x_0 : x_1 : x_2 : x_3) \in Z : x_3 \neq 0 \}$$

als

$$g_3 : U_3 \longrightarrow \boldsymbol{P}^1, (x_0 : x_1 : x_2 : x_3) \mapsto (x_2 : x_3).$$

Es gilt

$$Z = U_0 \cup U_3 \text{ und } g_0 \mid U_0 \cap U_3 = g_3 \mid U_0 \cap U_3 .$$

Siehe Macaulay2 Script

- MyExamples/RegularMap/Examples

ii) Die *Veronese Abbildung*

$$f : \boldsymbol{P}^2 \longrightarrow \boldsymbol{P}^5$$

ist die reguläre Abbildung, die global durch die quadratischen homogenen Polynome

$$f(t_0 : t_1 : t_2) := (t_0^2 : t_0 t_1 : t_0 t_2 : t_1^2 : t_1 t_2 : t_2^2)$$

gegeben ist. Das Bild

$$Z := f(\boldsymbol{P}^2) \subset \boldsymbol{P}^5$$

wird als projektive Varietät

$$Z = Var(I) \subset \boldsymbol{P}^5$$

durch das homogene Ideal

$$I = \langle T_4^2 - T_3 T_5, T_2 T_4 - T_1 T_5, T_2 T_3 - T_1 T_4, T_2^2 - T_0 T_5, T_1 T_2 - T_0 T_4, T_1^2 - T_0 T_3 \rangle \subset k[T_0, ..., T_5]$$

definiert. Die Einschränkung

$$f : \boldsymbol{P}^2 \longrightarrow \mathrm{Z} \subset \boldsymbol{P}^5$$

ist ein regulärer Isomorphimus: Die reguläre Umkehrabbildung

$$g : Z \longrightarrow \boldsymbol{P}^2$$

wird lokal auf jeder der drei offenen Teilmengen

$$U_i := \{ (x_0 : x_1 : x_2 : x_3 : x_4 : x_5) \in Z : x_i \neq 0, \}, i = 0, 3, 5,$$

durch die zugehörige Abbildung

$$g_0 : U_0 \longrightarrow \boldsymbol{P}^2, (x_0 : x_1 : x_2 : x_3 : x_4 : x_5) \mapsto (x_0 : x_1 : x_2)$$

$$g_3 : U_3 \longrightarrow \boldsymbol{P}^2, (x_0 : x_1 : x_2 : x_3 : x_4 : x_5) \mapsto (x_1 : x_3 : x_4)$$

$$g_5 : U_5 \longrightarrow \boldsymbol{P}^2, (x_0 : x_1 : x_2 : x_3 : x_4 : x_5) \mapsto (x_2 : x_4 : x_5)$$

gegeben.

Siehe Macaulay2 Script

- MyExamples/RegularMap/Examples

iii) Die global durch homogene quadratische Polynome definierte Abbildung

$$f : \boldsymbol{P}^n \times \boldsymbol{P}^m \longrightarrow \boldsymbol{P}^N,\ N = (n+1)\cdot(m+1)-1$$

$$\left((x_0 : \ldots : x_n), (y_0 : \ldots : y_m) \right) \mapsto (x_0 y_0 : \ldots : x_0 y_m : \ldots : x_n y_0 : \ldots x_n y_m) = \left(z_{ij} := x_i y_j\right)_{0 \le i \le n, 0 \le j \le m}$$

heißt *Segre Abbildung*. Sie dient dazu, das Produkt $\boldsymbol{P}^n \times \boldsymbol{P}^m$ zweier projektiver Räume als projektive Varietät in $\boldsymbol{P}^N$ darzustellen. Dazu zeigen wir, daß die Segre Abbildung eine injektive Abbildung mit abgeschlossenem Bild

$$Z := f\left(\boldsymbol{P}^n \times \boldsymbol{P}^m\right) \subset \boldsymbol{P}^N$$

ist. Wir fassen

$$\boldsymbol{P}^N := \left[M(n+1 \times m+1, K) - 0\right] / \sim$$

als den Quotienten aller von Null verschiedener Matrizen auf. Die Segre Abbildung wird dann von dem Matrizenprodukt zweier Vektoren

$$K^{n+1} \times K^{m+1} \longrightarrow M(n+1 \times m+1, K),\ (x, y) \mapsto x \cdot y^T.$$

durch Übergang zu den Äquivalenzklassen induziert. Nun sind für eine nichtverschwindende Matrix

$$A \in M(n+1 \times m+1, K)$$

äquivalent:

- $A = x \cdot y^T$ mit Vektoren $x,\ y \neq 0$
- $rang\ A = 1$

Die letzte Bedingung bedeutet, daß die Determinanten aller 2-Minoren von A verschwinden. Die Determinante eines gegebenen 2-Minors ist ein homogenes Polynom in den Koeffizienten der Matrix. Daher wird durch die Bedingung

$$rang\ A = 1$$

das Bild der Segre Abbildung

$$Z \subset \boldsymbol{P}^N := \left[M(n+1 \times m+1, K) - 0\right] / \sim$$

als projektive Varietät definiert.

Zur Injektivität der Segre Abbildung: Aus einer Gleichung

$$x \cdot y^T = \lambda \cdot x' \cdot y'^T \neq 0 \text{ für ein } \lambda \in K^*$$

folgt die Existenz eines Index i_0 mit

$$x \cdot y_{i_0} = \lambda \cdot x' \cdot y'_{i_0} \neq 0, \text{ also } x \cdot = \lambda \cdot \frac{y'_{i_0}}{y_{i_0}} \cdot x',$$

und die Existenz eines Index j_0 mit

$$x_{j_0} \cdot y = \lambda \cdot x'_{j_0} \cdot y' \neq 0, \text{ also } y \cdot = \lambda \cdot \frac{x'_{j_0}}{x_{j_0}} \cdot y'.$$

Vermöge der Segre Abbildung

$$f : \boldsymbol{P}^n \times \boldsymbol{P}^m \xrightarrow{\cong} Z \subset \boldsymbol{P}^N,\ N = (n+1)\cdot(m+1)-1$$

überträgt man die Zariski Topologie der projektiven Varietät $Z \subset \boldsymbol{P}^N$ auf das Produkt der beiden projektiven Räume $\boldsymbol{P}^n \times \boldsymbol{P}^m$. Die resultierende Topologie heißt die *Zariski Topologie* des Produktes $\boldsymbol{P}^n \times \boldsymbol{P}^m$. Bezüglich dieser Topologie faßt man

$$\boldsymbol{P}^n \times \boldsymbol{P}^m$$

als projektive Varietät auf. Die Zariski Topologie des Produktes ist i.a. echt feiner als das Produkt der Zariski Topologien beider Faktoren.

Im Spezialfall $n = 1, m = 1$ hat die Segre Abbildung

$$f : \boldsymbol{P}^1 \times \boldsymbol{P}^1 \longrightarrow \boldsymbol{P}^3$$

die Gestalt

$$f((x_0 : x_1),(y_0 : y_1)) := (x_0y_0 : x_0y_1 : x_1y_0 : x_1y_1).$$

Ihr Bild ist die quadratische Hyperfläche

$$Z = Var(I) \subset \boldsymbol{P}^3$$

zum Ideal

$$I := \langle Z_0Z_3 - Z_1Z_2 \rangle \subset k[Z_0, Z_1, Z_2, Z_3],$$

Man erhält eine Darstellung des Produktes zweier projektiver Geraden als eine quadratische Hyperfläche

$$f : \boldsymbol{P}^1 \times \boldsymbol{P}^1 \xrightarrow{\cong} Z \subset \boldsymbol{P}^3.$$

iv) Die projektive Ebene $\boldsymbol{P}^2$ läßt sich gemäß Abbildung 1 veranschaulichen als Abschluß der affinen Ebene $\boldsymbol{A}^2$ durch die unendlich ferne Gerade

$$Var(\langle Z_0 \rangle) \subset \boldsymbol{P}^2.$$

Der Punkt $(1:0:0) \in \boldsymbol{P}^2$ ist der Nullpukt der affinen Ebene. In Verallgemeinerung des affinen Blow-up bilden die Geraden in der projektiven Ebene $\boldsymbol{P}^2$, welche durch den Punkt $(1:0:0) \in \boldsymbol{P}^2$ gehen, die projektive Varietät

$$Z := \{ ((x_0 : x_1 : x_2),(z_0 : z_1)) \in \boldsymbol{P}^2 \times \boldsymbol{P}^1 : z_0 \cdot x_2 - z_1 \cdot x_1 = 0 \},$$

das *Blow-up* von $\boldsymbol{P}^2$ im Punkt $(1:0:0) \in \boldsymbol{P}^2$. Die erste Projektion induziert eine reguläre Abbildung

$$f : Z \longrightarrow \boldsymbol{P}^2,\ (x,z) \mapsto x.$$

Das Urbild

$$E := f^{-1}(1:0:0) \subset Z$$

heißt die *exzeptionelle Gerade* des Blow-up. Sie läßt sich unter der zweiten Projektion

$$pr_2 : E \xrightarrow{\cong} \boldsymbol{P}^1,\ (x,z) \mapsto z$$

mit einer projektiven Geraden identifizieren. Die Punkte der exzeptionellen Geraden parametrisieren dabei die Geraden der projektiven Ebene durch den Punkt $(1:0:0) \in \boldsymbol{P}^1$

$$E \ni z = (z_0 : z_1) \mapsto L_z = \{ (x_0 : x_1 : x_2) \in \boldsymbol{P}^2 : z_0 x_2 - z_1 x_1 = 0 \} \subset \boldsymbol{P}^2 .$$

Die y-Achse hat dabei den Parameter

$$(0:1) \in \boldsymbol{P}^1 ,$$

und entspricht der fehlenden Geraden im affinen Beispiel. Die x-Achse hat den Parameter

$$(1:0) \in \boldsymbol{P}^1 .$$

Die wichtigste Eigenschaft einer regulären Abbildung zwischen projektiven Varietäten ist ihre Abgeschlossenheit. Der entsprechende Satz gilt im Affinen nur unter Zusatzvoraussetzungen wie z.B. der Endlichkeit.

6.12 Satz *(Projektionssatz)*

i) Die Projektion

$$pr_2 : \boldsymbol{P}^n \times \boldsymbol{P}^m \longrightarrow \boldsymbol{P}^m, (x, y) \mapsto y ,$$

ist eine abgeschlossene Abbildung, d.h. das Bild einer Zariski-abgeschlossenen Teilmenge von $\boldsymbol{P}^n \times \boldsymbol{P}^m$ ist eine projektive Varietät von $\boldsymbol{P}^m$.

ii) Jede reguläre Abbildung

$$f : X \longrightarrow Y$$

zwischen projektiven Varietäten ist abgeschlossen.

Beweis. ad i) Gegeben sei eine projektive k-Varietät

$$X = \{ (x, y) \in \boldsymbol{P}^n \times \boldsymbol{P}^m : f_i(x, y) = 0 \ \textit{für}\ i = 1,...,N \},$$

welche definiert wird durch Polynome

$$f_i = f_i(X_0, X_1, ..., X_n, Y_0, Y_1, ..., Y_m) \in k[X_0, X_1, ..., X_n, Y_0, Y_1, ..., Y_m]$$

der Form

$$f_i = f_i(X_0, X_1, ..., X_n, Y_0, Y_1, ..., Y_m) = \sum_j a_j^i(X_0, X_1, ..., X_n) \cdot b_j^i(Y_0, Y_1, ..., Y_m) .$$

Dabei sind die Polynome

$$a_j^i(X_0, X_1, ..., X_n) \in k[X_0, X_1, ..., X_n] \text{ homogen vom Grad } d_i$$

und

$$b_j^i(Y_0, Y_1, ..., Y_m) \in k[Y_0, Y_1, ..., Y_m] \text{ homogen.}$$

Für einen Punkt $y \in \boldsymbol{A}^{m+1} - 0$ und seine Äquivalenzklasse $[y] \in \boldsymbol{P}^m$ sind die folgenden beiden Aussagen äquivalent:

- $[y] \in pr_2(X)$
- Die Polynome

$$f_1(-, y), ..., f_N(-, y) \in k[X_0, X_1, ..., X_n]$$

haben eine gemeinsame Nullstelle in $\boldsymbol{A}^{n+1} - 0$.

Die letzte Eigenschaft wird nun mit Hilfe des Hilbertschen Nullstellensatzes in eine Aussage der Linearen Algebra umgeformt. Nach dem Hilbertschen Nullstellensatz sind äquivalent:

- $[y] \notin pr_2(X)$
- $Var(\langle f_1(-,y),...,f_N(-,y)\rangle) = \{0\}$
- $\sqrt{\langle f_1(-,y),...,f_N(-,y)\rangle} = \langle X_0,...,X_n\rangle$
- Es gibt ein $d \in \boldsymbol{N}$ mit

$$\langle X_0,...,X_n\rangle^d \subset \langle f_1(-,y),...,f_N(-,y)\rangle .$$

Daher ist zu zeigen, daß die Teilmenge

$$\left\{ y \in \boldsymbol{A}^{m+1} : Es\ gibt\ kein\ d \in \boldsymbol{N}\ mit \langle X_0,...,X_n\rangle^d \subset \langle f_1(-,y),...,f_N(-,y)\rangle \right\}$$

das Nullstellengebilde homogener Polynome ist. Wir bezeichnen mit

$$H(e) \subset k[X_0, X_1,..., X_n]$$

den endlich-dimensionalen k-Vektorraum der homogenen Polynome vom Grad $e \in \boldsymbol{N}$ und betrachten für jedes $y \in \boldsymbol{A}^{m+1}$ und $d \in \boldsymbol{N}$ die k-lineare Abbildung

$$F(d,\ y): H(d-d_1) \oplus H(d-d_2) \oplus ... \oplus H(d-d_N) \longrightarrow H(d), (g_1,...,g_N) \mapsto \sum_{i=1}^{N} f_i(-,y) \cdot g_i$$

Diese Abbildung ist genau dann surjektiv, wenn

$$\langle X_0,...,X_n\rangle^d \subset \langle f_1(-,y),...,f_N(-,y)\rangle .$$

Nach Wahl von Basen wird die lineare Abbildung $F(d,y)$ durch eine Matrix $M(F(d,y))$ beschrieben, deren Koeffizienten als Funktionen von $y \in \boldsymbol{A}^{m+1}$ homogene Polynome in den Veränderlichen $Y_0, Y_1,..., Y_m$ sind. Es sind äquivalent:

- $F(d,y)$ nicht surjektiv
- Der Rang von $F(d,y)$ ist nicht maximal
- Alle maximalen Minoren der Matrix $M(F(d,y))$ haben eine verschwindende Determinante.

Die Determinante einer Matrix ist ein homogenes Polynom in den Koeffizienten der Matrix, und die Koeffizienten der Matrix $M(F(d,y))$ hängen homogen von den Veränderlichen $Y_0, Y_1,..., Y_m$ ab. Daher bedeutet die letzte Bedingung das Verschwinden homogener Polynome in $Y_0, Y_1,..., Y_m$ an der Stelle $y \in \boldsymbol{A}^{m+1}$, definiert also bei variablem $[y] \in \boldsymbol{P}^m$ eine projektive Varietät $Y(d) \subset \boldsymbol{P}^m$. Die Darstellung

$$pr_2(X) = \bigcap_{d \in \boldsymbol{N}} Y(d)$$

zeigt, daß auch

$$pr_2(X) \subset \boldsymbol{P}^m$$

eine projektive Varietät ist.

ad ii) Sind $X \subset \boldsymbol{P}^n$ und $Y \subset \boldsymbol{P}^m$ projektive Varietäten, so betrachtet man den Graphen von f

$$\Gamma_f := \{ (x, y) \in X \times Y : y = f(x) \} \subset \boldsymbol{P}^n \times \boldsymbol{P}^m$$

der ebenfalls eine projektive Varietät ist. Die erste Projektion

$$pr_1 : \Gamma_f \xrightarrow{\cong} X$$

ist ein regulärer Isomorphismus, sei

$$j := f^{-1} : X \longrightarrow \Gamma_f$$

die Umkehrabbildung. Die gegebene Abbildung faktorisiert als

$$f = \left[X \xrightarrow{j} \Gamma_f \xrightarrow{pr_2} Y \right]$$

Nach dem bereits bewiesenen Teil des Satzes ist

$$f(X) = pr_2(\Gamma_f)$$

als Teilmenge von $\boldsymbol{P}^m$ und damit auch als Teilmenge von Y abgeschlossen, q.e.d.

Numerische Invarianten einer projektiven Varietät

Wir haben im Kapitel über Affine Varietäten das Hilbert Polynom eingeführt, um die Dimension einer affinen Varietät zu definieren. Dabei wurde deutlich, daß das Hilbert Polynom primär allerdings gar keine Eigenschaft der affinen Koordinatenringe, sondern eine Eigenschaft von graduierten k-Algebren ist. Das Hilbert Polynom codiert die Eigenschaften des homogenen Koordinatenringes. und der affine Fall läßt sich durch Homogenisierung auf den homogenen Fall zurückführen.

Da die projektive algebraische Geometrie nun gerade von graduierten Algebren handelt, läßt sich die Theorie des Hilbert Polynoms auf projektiv-algebraische Varietäten und ihre Koordinatenringe direkt anwenden.

6.13 Definition (Hilbert Polynom)

Das Hilbert Polynom

$$HP_X(T) \in \boldsymbol{Q}[T]$$

einer projektiven k-Varietät $X \subset \boldsymbol{P}^n$ ist das Hilbert Polynom ihres homogenen Koordinatenringes $S(X)$.

6.14 Definition (Hilbert Polynom und numerische Invarianten)

Es sei $X \subset \boldsymbol{P}^n$ eine projektive Varietät mit Hilbert Polynom

$$HP_X(T) = e \cdot \frac{T^m}{m!} + \textit{Terme niedrigeren Grades in } T$$

Man nennt

- den Grad $m \in \boldsymbol{N}$ die *Dimension* $dim(X)$ von X
- die Zahl $e \in \boldsymbol{N}$ den *Grad* $\deg(X, \boldsymbol{P}^n)$ der Einbettung $X \xrightarrow{\subset} \boldsymbol{P}^n$
- und den Funktionswert

  $$p_a(X) := (-1)^m \cdot (HP_X(0) - 1)$$

  das *arithmetische Geschlecht* von X.

Für den projektiven Raum $\boldsymbol{P}^n$ gilt $m = n$, $e = 1$ und $p_a = 0$.

6.15 Toolbeispiel (Hilbert Polynom und numerische Invarianten)

Macaulay2 berechnet in

- „MyExamples/HilbertPolynomial/Examples“

die folgenden Beispiele:

Getwistete Kubik

Veronese Fläche

Segre Einbettung

Elliptische Kurve

6.16 Bemerkung *(Numerische Invarianten einer Varietät)*

i) Der Grad der Einbettung einer projektiven Varietät ist eine positive ganze Zahl. Sie hängt nicht nur von der Varietät, sondern auch von der Art der Einbettung ab.

ii) Dimension und arithmetisches Geschlecht einer projektiven Varietät hängen dagegen nicht von der Einbettung ab, sind also intrinsische Größen der Varietät.

Das arithmetische Geschlecht ist eine nicht-negative ganze Zahl.

Neben dem hier definierten *arithmetischen* Geschlecht besitzt eine projektive Varietät auch noch ein *geometrisches* Geschlecht. Beide Größen entspringen aus verschiedenen Quellen: Das arithmetische Geschlecht aus den regulären Funktionen, das geometrische aus den regulären Diffentialformen. Im Falle einer nicht-singulären Kurve fallen beide numerischen Invarianten zusammen.

iii) Der Grad einer projektiven Hyperfläche

$$X = Var(f) \subset \boldsymbol{P}^n$$

stimmt überein mit dem Grad des definierenden homogenen Polynoms

$$f \in k[T_0,...,T_n].$$

Eine der wichtigsten Anwendungen des Grades ist die Bestimmung der Schnittzahl zweier projektiver Kurven. Die Schwierigkeit besteht darin, lokal für jeden Schnittpunkt seine Vielfachheit zu definieren. Dann gilt:

6.17 Satz *(Bezout)*

Es sei $X_1, X_2 \subset \boldsymbol{P}^2$ zwei projektive Kurven, deren Durchschnitt

$$X = X_1 \cap X_2$$

keine irreduzible Komponente von X_1 oder von X_2 enthält. Dann besteht der Durchschnitt X aus endlich vielen Punkten. Ihre Anzahl – mit Vielfachheit – ist das Produkt der Grade

$$\deg(X_1) \cdot \deg(X_2).$$

Beweis. [CLO1997], Chap. 8, §7.

Der Satz von Bezout läßt sich auf höherdimensionale Varietäten verallgemeinern.

6.18 Satz *(Grad und Geschlecht ebener projektiver Kurven)*

Bei einer projektiven Kurve, die sich als Hyperfläche $X \subset \boldsymbol{P}^2$ in die projektive Ebene einbetten läßt, stehen Geschlecht

$$p = p_a(X)$$

und Grad

$$d = d(X)$$

in folgender Beziehung

$$p = \frac{(d-1)\cdot(d-2)}{2}.$$

Beweis. Die Kurve werde durch das Ideal

$$I = \langle f \rangle \subset R := k[T_0, T_1, T_2]$$

mit einem homogenen Polynom $f \in R$ vom Grad d definiert. Dann gibt es eine exakte Sequenz

$$0 \longrightarrow R \xrightarrow{f} R \longrightarrow R/I = k[X] \longrightarrow 0,$$

die zweite Abbildung ist die Multiplikation mit f, die dritte die kanonische Restklassenabbildung. Die Hilbert Funktion ist additiv. Es gilt:

$$Hilb_R(s) = Hilb_R(s-d) + Hilb_{k[X]}(s), s \in \mathbf{Z},$$

also

$$Hilb_{k[X]}(s) = Hilb_R(s) - Hilb_R(s-d) = \binom{2+s}{2} - \binom{2-d+s}{2}$$

$$= d \cdot s + 1 + \frac{(d-2)\cdot(d-1)}{2}.$$

Hieraus liest man ab

$$m = dim(X) = 1,\ deg(X) = d \text{ und } p(X) = \frac{(d-2)\cdot(d-1)}{2}, \text{ q.e.d.}$$

7 Arithmetische und komplexe Punkte

Dieses Kapitel zeigt die enge Verbindung der Algebraischen Geometrie zu zwei anderen Gebieten der Mathematik, zur Zahlentheorie und zur Theorie der Riemannschen Flächen. Das Kapitel gibt dabei einen Ausblick auf einige der tiefsten Resultate der Mathematik im 20. Jahrhundert.

Eine k-Varietät wird auf Seiten der Algebra durch ein Ideal

$$I \subset k[X_1,...,X_n]$$

in einem Polynomring über dem Körper k definiert. Der Körper k enthält zumindest alle Koeffizienten der definierenden Polynome. Auf der Seite der Geometrie betrachtet man das Nullstellengebilde im affinen Raum mit dem algebraischen Abschluß $K = \bar{k}$ als Koordinatenkörper:

$$Var(I) \subset \boldsymbol{A}^n(K).$$

Mit demselben Recht kann man aber auch das Nullstellengebilde in einem affinen Raum $\boldsymbol{A}^n(L)$ über einem beliebigen Erweiterungskörper $L \supset k$ betrachten, der nicht notwendig algebraisch-abgeschlossen ist. Bei einer Varietät X spielen also zumindest zwei Körper eine Rolle:

- Der Definitionskörper k, über dem X definiert ist (Algebra)
- Der Koordinatenkörper L, über dem man das Nullstellengebilde betrachtet (Geometrie).

Um auszudrücken, daß X über k *definiert* ist, schreibt man X / k, d.h.

$$I \subset k[X_1,...,X_n].$$

Dagegen schreibt man für die Menge der L*-wertigen Punkte* von X

$$X(L) := \{\, x \in \boldsymbol{A}^n(L) :\ f(x) = 0 \ \textit{für alle}\ f \in I \,\}$$

Alle Überlegungen gelten nicht nur für affine Varietäten, sondern genauso für projektive Varietäten.

Zahlentheorie

Wenn man die Zahlentheorie in der Sprache der Algebraischen Geometrie formuliert, so sind zunächst die Fälle

$$k = \boldsymbol{Q}$$

als Definitionskörper und

$$L \supset \boldsymbol{Q}$$

eine algebraische Körpererweiterung als Koordinatenkörper interessant. Die L-wertigen Punkte heißen in diesem Falle *arithmetische* Punkte.

Desweiteren studiert man sogar den Fall, daß die Koeffizienten der definierenden Polynome aus einem kommutativen Ring stammen, der kein Körper ist. Daß die Varietät X also z.B. über dem Ring der ganzen Zahlen $\boldsymbol{Z}$ definiert ist, d.h. durch ein Ideal

$$I \subset \boldsymbol{Z}[X_1,...,X_n],$$

dessen Elemente Polynome mit ganzzahligen Koeffizienten sind. Jeder Ringmorphismus

$$\boldsymbol{Z} \longrightarrow R$$

induziert dann einen *Basiswechsel*, nämlich die durch das erweiterte Ideal

$$I^e \subset R[X_1,...,X_n]$$

definierte Varietät X_R. Die interessantesten Basiswechsel in der Zahlentheorie stammen von den Morphismen

$$\boldsymbol{Z} \longrightarrow \boldsymbol{F}_q$$

zu den endlichen Restklassenkörpern $\boldsymbol{F}_p$ mit einer Primzahl p, die Basiswechsel

$$\boldsymbol{Z} \xrightarrow{\subset} \boldsymbol{Q}_p$$

zu den p-adischen Körpern $\boldsymbol{Q}_p$, und der Basiswechsel

$$\boldsymbol{Z} \xrightarrow{\subset} \boldsymbol{R}$$

zum Körper der reellen Zahlen.

Eine erste Unterscheidung der beim Studium von L-wertigen Punkte auftretenden Erscheinungen läßt sich anhand des Grades bzw. des dahinter stehenden Geschlechtes durchführen.

7.1 **Bemerkung** *(Linearer Teilraum)*

Eine *lineare* projektive Varietät X / k wird durch ein Ideal $I \subset k[X_0,...,X_n]$ definiert, das von homogenen Polynomen 1. Grades, d.h. linearen Polynomen erzeugt wird. Die Menge der L-wertigen Punkte, $L \supset k$, bildet für jeden Koordinatenkörper L immer einen linearen Unterraum fester Dimension des projektiven Raums:

$$X(L) \cong \boldsymbol{P}^d(L) \subset \boldsymbol{P}^n(L).$$

7.2 **Bemerkung** *(Quadrik)*

Eine *Quadrik* ist eine projektive Varietät X / k, deren Ideal durch ein homogenes Polynom 2. Grades, d.h. ein quadratisches Polynom erzeugt wird. Die Menge der L-wertigen Punkte hängt stark vom Koordinatenkörper $L \supset k$ ab.

Beispielsweise hat die projektive Kurve $X / \boldsymbol{Q}$, die durch das Polynom

$$f(X_0, X_1, X_2) = X_0^2 + X_1^2 + X_2^2 \in k[X_0, X_1, X_2]$$

definiert ist, keinen $\boldsymbol{Q}$-wertigen (d.h. rationalen) oder $\boldsymbol{R}$-wertigen (d.h. reellen) Punkt. Sie hat aber viele $\boldsymbol{C}$-wertige (d.h. komplexe) Punkte, es gilt sogar

$$X(\boldsymbol{C}) \cong \boldsymbol{P}^1(\boldsymbol{C}).$$

Der folgende Satz stellt ein Lokal-Global Prinzip dar. Man betrachtet seine Aussage über dem Zahlkörper $\boldsymbol{Q}$ als eine globale Aussage und seine Aussagen über die Komplettierungen von

$\boldsymbol{Q}$ bzgl. der verschiedenen Bewertungen als eine lokale Aussage. Als weiterführende Literatur siehe [Maz1993].

7.3 Satz *(Hasse-Minkowski, 1920)*

Eine Quadrik $X / \boldsymbol{Q}$ hat genau dann einen rationalen Punkt, wenn der Basiswechsel X_L

- für den Körper $L = \boldsymbol{R}$
- und für alle Primzahlen p für den Körper $L = \boldsymbol{Q}_p$ der p-adischen Zahlen

jeweils einen L-wertigen Punkt besitzt.

Beweis. Für einen Beweis siehe [Fre1984], Kap. V, Satz 3.9.

7.4 Definition *(Elliptische Kurve)*

Eine *elliptische Kurve* E / k ist eine nicht-singuläre projektive Kurve mit Geschlecht $g = 1$, zusammen mit einem ausgezeichneten k-wertigen Punkt $0 \in E(k)$.

7.5 Bemerkung *(Elliptische Kurve)*

Jede elliptische Kurve E / k läß sich schon als Hyperfläche in der projektiven Ebene realisieren:

$$E = Var(f) \subset \boldsymbol{P}^2$$

mit einem homogenen Polynom

$$f \in k[X, Y, Z].$$

Nach Satz 6.18 hat f den Grad $\deg f = 3$. Im Falle Charakteristik $char\, k \neq 2, 3$ kann man erreichen, daß f die Homogenisierung eines Polynoms g in Weierstraß Normalform ist

$$g(X, Y) = Y^2 - X^3 - A \cdot X - B \in k[X, Y]$$

und daß der ausgezeichnete Punkt der unendlich ferne Punkt ist

$$0 := (0:1:0) \in E(k) \subset \boldsymbol{P}^2 .$$

Das Nichtverschwinden der Diskriminante

$$\Delta := -16 \left(4 \cdot A^3 + 27 \cdot B^2\right) \neq 0$$

garantiert, daß die projektive Kurve nicht-singulär ist.

Für jeden Erweiterungskörper $L \supset k$ bildet die Menge $E(L)$ eine Abelsche Gruppe mit dem ausgezeichneten Punkt als neutralem Element 0.

Hinweis. Das grundlegende Lehrbuch über elliptische Kurven ist [Sil1986].

7.6 Toolbeispiel *(Elliptische Kurve und Diskriminante)*

Pari-Beispiel

- MyExamples/Example1

i) Die elliptische Kurve $E/\boldsymbol{Q}$ mit der Weierstraß Normalform

$$Y^2 = X^3 + X$$

hat die Diskriminante

$$\Delta = -64 .$$

Die Kurve hat eine relle Nullstelle

$$x_1 = 0$$

und die beiden konjugiert-komplexen Nullstellen

$$x_{2,3} = +/- i .$$

Die relle Nullstelle definiert einen rationalen Punkt

$$p := (0, 0) \in E(\boldsymbol{Q}),$$

der in der Abelschen Gruppe $(E(\boldsymbol{Q}), +)$ die Ordnung 2 hat und damit die Torsionsuntergruppe erzeugt.

ii) Die elliptische Kurve $E/\boldsymbol{Q}$ mit der Weierstraß Normalform

$$Y^2 = X^3 + 1$$

hat die Diskriminante

$$\Delta = -432 .$$

Die Kurve hat eine reelle Nullstelle

$$x_1 = -1$$

und zwei konjugiert-komplexe Nullstellen. Die relle Nullstelle definiert einen rationalen Punkt

$$p := (-1, 0) \in E(\boldsymbol{Q}),$$

der in der Abelschen Gruppe $(E(\boldsymbol{Q}), +)$ die Ordnung 2 hat. Der Punkt

$$q := (2, 3) \in E(\boldsymbol{Q})$$

hat die Ordnung 6 und erzeugt damit die Torsionsuntergruppe.

7.7 Satz (Mordell, 1922)

Es sei k ein Zahlkörper und E/k eine elliptische Kurve. Die Abelsche Gruppe $E(k)$ der k-wertigen Punkte ist endlich erzeugt, d.h.

$$E(k) \cong \boldsymbol{Z}^r \oplus E_{tors}(k),$$

wobei $r \in \boldsymbol{N}$ ihren Rang und $E_{tors}(k)$ ihre Torsionsuntergruppe bezeichnet.

Beweis. [Sil1986], Chap. VIII, Theor. 6.7.

Der Rang einer elliptischen Kurve ist eine subtile arithmetische Charakteristik. Es wird vermutet, daß elliptische Kurven mit beliebig großem Rang existieren.

7.8 Toolbeispiel *(Eliptische Kurve mit positivem Rang)*

Pari-Beispiel

- MyExamples/Example2

Die elliptische Kurve $E/\boldsymbol{Q}$ mit der Weierstraß Normalform

$$Y^2 = X^3 - 7 \cdot X + \frac{25}{4}$$

hat die Diskriminante

$$\Delta = 5077 .$$

Die Torsionuntergruppe der rationalen Punkte von $E(\boldsymbol{Q})$ ist trivial. Die Gruppe $E(\boldsymbol{Q})$ enthält jedoch zahlreiche Punkte, hat also positiven Rang. Es ist bekannt, daß der Rang $= 3$ ist.

7.9 Beispiel *(Selmer, 1951)*

Für Kurven 3. Grades gilt das Analogon des Satzes von Hasse-Minkowski nicht mehr: Die projektive Kurve $X/\boldsymbol{Q}$, welche durch das homogene Polynom

$$f(X_0, X_1, X_2) = 3 \cdot X_0^3 + 4 \cdot X_1^3 + 5 \cdot X_2^3 \in k[X_0, X_1, X_2]$$

definiert ist, hat keinen rationalen Punkt. Sie enthält aber den reellen Punkt

$$\left(\sqrt[3]{3} : -1 : -1 \right) \in X(\boldsymbol{R}) .$$

Außerdem gilt nach einem Satz von Hasse für die Anzahl der $\boldsymbol{F}_p$-wertigen Punkte

$$\left| \# X(\boldsymbol{F}_p) - p \right| \leq 2 \cdot \sqrt{p}$$, p eine Primzahl.

Insbesondere hat $X/\boldsymbol{Q}$ also neben dem ausgezeichneten Punkt noch weitere $\boldsymbol{F}_p$-wertige Punkte. Nach dem Hensel Lemma hat $X/\boldsymbol{Q}$ dann auch weitere $\boldsymbol{Q}_p$-wertige Punkte.

Hinweis. Als weiterführende Literatur siehe [Maz1993].

Projektive Kurven $X/\boldsymbol{Q}$ höheren Geschlechtes haben immer nur endlich viele rationale Punkte, diese Eigenschaft wurde 1922 von Mordell vermutet.

7.10 Satz *(Faltings, 1983)*

Eine projektive nicht-singuläre Kurve $X/\boldsymbol{Q}$ vom Geschlecht $g \geq 2$ hat nur endlich viele rationale Punkte, d.h.

$$\# X(\boldsymbol{Q}) < \infty .$$

Hinweis. Für eine Übersicht siehe [Fal1984], für eine fortgeschrittene Darstellung siehe den Tagungsband [CS1986].

Das bisher wichtigste Ergebnis aus der Theorie elliptischer Kurven ist der Beweis der Taniyama-Weil Vermutung. Aus ihr ergibt sich die Fermat Vermutung (1637).

7.11 Satz (Wiles, 1994)

Die projektive Varietät $X / \boldsymbol{Q}$, die durch das Polynom

$$f(X_0, X_1, X_2) = -X_0^d + X_1^d + X_2^d \in \boldsymbol{Q}[X_0, X_1, X_2]$$

definiert ist, hat für $d \geq 3$ keinen rationalen Punkt, d.h. $X(\boldsymbol{Q}) = \varnothing$.

Hinweis. Für eine Übersicht siehe [Fal1995], für eine fortgeschrittene Darstellung siehe den Tagungsband [CSS1995].

Riemannsche Flächen

Der Basiswechsel vermöge

$$\boldsymbol{Z} \xrightarrow{\subset} \boldsymbol{C}$$

und das Studium der komplexen Punkte führt in die Funktionentheorie, die Theorie der Riemannschen Flächen oder allgemeiner in die Komplexe Analysis. Die Komplexe Analysis untersucht „Komplexe Mannigfaltigkeiten“ bzw. „Komplexe Räume“ mit holomorphen Abbildungen als zugehörigen Morphismen. Diese werden lokal durch konvergente Potenzreihen gegeben.

Die komplexwertigen Punkte

$$X(\boldsymbol{C})$$

einer Varietät, versehen mit der Euklidischen Topologie, sind komplexe Räume bzw. im nicht-singulären Fall komplexe Mannigfaltigkeiten. Komplexe Mannigfaltigkeiten der Dimension 1 heißen Riemannsche Flächen:

7.12 Bemerkung (Riemannsche Flächen)

Ein topologischer Hausdorff Raum, der jeweils lokal homöomorph ist zu einer offenen Menge in der komplexen Zahlenebene $\boldsymbol{C}$, zusammen mit einem Atlas biholomorph verträglicher Karten, heißt *Riemannsche Fläche*.

Hinweis. Als Literatur siehe [For1977].

7.13 Satz (Kompakte Riemannsche Flächen)

Jede kompakte Riemannsche Fläche ist projektiv algebraisch, d.h. zu jeder kompakten Riemannschen Fläche Y existiert eine nicht-singuläre projektive Kurve $X / \boldsymbol{C}$ und ein biholomorpher Isomorphismus

$$Y \cong X(\boldsymbol{C}).$$

Beweis. [Har1977] Chap. IV, Sect. 3.

Dieser Satz ist unter anderem deswegen bemerkenswert, weil Riemannsche Flächen durch holomorphe Funktionen, also lokal durch konvergente Potenzreihen, definiert werden. Die Varietäten der Algebraischen Geometrie sind dagegen immer durch Polynome definiert.

7.14 Bemerkung (Geschlecht kompakter Riemannscher Flächen)

Die wichtigste numerische Invariante einer kompakten Riemannschen Fläche ist ihr *Geschlecht*. Dieses ist über die holomorphe Struktur definiert und stimmt mit der Anzahl der linear unabhängigen holomorphen Differentialformen überein; siehe [For1977] Bem. 17.10. Als Geschlecht treten alle nicht-negativen ganzen Zahlen auf.

Das Geschlecht stellt sich als eine topologische Invariante heraus und liefert sogar eine topologische Klassifizierung: Zwei kompakte Riemannsche Flächen sind als topologische Mannigfaltigkeiten genau dann isomorph, wenn sie dasselbe Geschlecht haben.

In Satz 7.13 stimmt das Geschlecht der Riemannschen Fläche Y mit dem arithmetischen Geschlecht der projektiven Kurve $X / \boldsymbol{C}$ überein.

Alle nicht-singulären projektiven Kurven X / k lassen sich bereits in einem 3-dimensionalen projektiven Raum realisieren

$$X \xrightarrow{\subset} \boldsymbol{P}^3,$$

nicht-singuläre Kurven vom Geschlecht $g \leq 1$ sogar schon als Hyperflächen in der projektiven Ebene $\boldsymbol{P}^2$.

7.15 Definition *(Torus)*

Ein *Torus* ist der Quotient $(\boldsymbol{C}, +)/\Gamma$ der additiven Gruppe der komplexen Zahlen nach einem Gitter

$$\Gamma = \boldsymbol{Z} \cdot \omega_1 + \boldsymbol{Z} \cdot \omega_2 \text{ mit } \omega_1, \omega_2 \in \boldsymbol{C} \text{ linear unabhängig über } \boldsymbol{R}.$$

Man nennt Γ das *Periodengitter* und $\omega_1, \omega_2 \in \boldsymbol{C}$ die *Perioden* des Torus.

Jeder Torus ist eine kompakte Riemannsche Fläche vom Geschlecht $g = 1$. Außerdem ist ein Torus eine Abelsche Gruppe bezüglich der von $(\boldsymbol{C}, +)$ induzierten Addition.

Die Isomorphieklassen von Tori unter biholomorphen Abbildungen sind wohlbekannt. Sie entsprechen bijektiv den Äquivalenzklassen der Operation der Modulgruppe auf der oberen Halbebene $\boldsymbol{H}$. Nach einem klassischen Resultat aus der Theorie der Modulfunktionen läßt sich die Menge dieser Äquivalenzklassen durch die absolute Modulfunktion bijektiv auf die komplexen Zahlen abbilden. Grundlage dieser Theorie sind die auf der oberen Halbebene für jedes $k > 1$ definierten Eisenstein Reihen

$$G_k(\tau) := \sum_{\omega \in \Gamma(\tau) - 0} \frac{1}{\omega^{2k}}, \ \tau \in \boldsymbol{H}, \ \Gamma(\tau) := \boldsymbol{Z} \cdot 1 + \boldsymbol{Z} \cdot \tau.$$

7.16 Lemma *(Isomorphieklassen von Tori)*

i) Jeder Torus ist isomorph unter einer holomorphen Abbildung zu einem Torus mit einem normierten Periodengitter

$$\Gamma(\tau), \tau \in \boldsymbol{H}.$$

ii) Zwei normierte Periodengitter $\Gamma(\tau_1)$ und $\Gamma(\tau_2)$ definieren genau dann isomorphe Tori, wenn es eine Matrix gibt

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \in SL(2, \boldsymbol{Z}) \text{ mit } \tau_2 = \frac{a \cdot \tau_1 + b}{c \cdot \tau_1 + d},$$

d.h. die Äquivalenzklassen isomorpher Tori entsprechen bijektiv den Bahnen der Gruppenoperation

$$SL(2, \boldsymbol{Z}) \times \boldsymbol{H} \longrightarrow \boldsymbol{H}, (A, \tau) \mapsto \frac{a \cdot \tau + b}{c \cdot \tau + d}.$$

iii) Die absolute Modulfunktion

$$j : \boldsymbol{H} \longrightarrow \boldsymbol{C} \text{ mit } j := 1728 \cdot \frac{g_2^{\,3}}{\Delta}$$

$$g_2 := 60 \cdot G_2,\ g_3 := 140 \cdot G_3, \Delta := {g_2}^3 - 27 \cdot {g_3}^2$$

ist invariant unter dieser Operation und induziert eine bijektive Abbildung des Bahnenraumes

$$\boldsymbol{H}/SL(2, \boldsymbol{Z}) \xrightarrow{\cong} \boldsymbol{C}.$$

Hinweis. Zur Literatur siehe [Gun1972], [Ser1973].

7.17 Satz (Tori und elliptische Kurven)

i) Es sei $X = \boldsymbol{C}/\Gamma$ ein Torus mit einem normierten Periodengitter $\Gamma(\tau), \tau \in \boldsymbol{H}$. Dann definieren seine Weierstraß'sche $\wp$-Funktion

$$\wp(z) := \frac{1}{z^2} + \sum_{\omega \in \Gamma - 0} \left(\frac{1}{(z-\omega)^2} - \frac{1}{\omega^2} \right)$$

und ihre Ableitung

$$\wp'(z) = -2 \cdot \sum_{\omega \in \Gamma} \frac{1}{(z-\omega)^3}$$

eine injektive holomorphe Abbildung

$$\boldsymbol{C}/\Gamma \xrightarrow{\subset} \boldsymbol{P}^2,\ [z] \mapsto \begin{cases} \left(\wp(z) : \frac{1}{2}\wp'(z) : 1 \right) & z \notin \Gamma \\ (0:1:0) & z \in \Gamma \end{cases}.$$

Diese bildet den Torus biholomorph ab auf die komplexen Punkte der elliptischen Kurve $E/\boldsymbol{C}$ mit der Weierstraß Normalform

$$g(X,Y) = Y^2 - X^3 - A \cdot X - B \in \boldsymbol{C}[X,Y],\ A := -15 \cdot G_2(\tau),\ B := -35 \cdot G_3(\tau).$$

Die Abbildung ist zudem ein Isomorphismus Abelscher Gruppen

$$(\boldsymbol{C}/\Gamma, +) \xrightarrow{\cong} (E(\boldsymbol{C}), +).$$

ii) Jede Isomorphieklasse einer elliptische Kurve $E/\boldsymbol{C}$ tritt auf diese Art als Bild eines Torus auf.

Beweis. ad i) Die Aussage folgt aus der Differentialgleichung der Weierstraß'schen $\wp$-Funktion

$$\wp'^2 = 4 \cdot \wp^3 - g_2 \cdot \wp - g_3$$

und den Additionstheoremen der $\wp$-Funktion; siehe [Ahl1966].

ad ii) Jede über dem algebraisch-abgeschlossenen Körper $\boldsymbol{C}$ definierte elliptische Kurve $E/\boldsymbol{C}$ läßt sich auch durch eine Gleichung 3. Grades in Legendre Normalform definieren:

$$Y^2 = X \cdot (X-1) \cdot (X-\lambda) \in \boldsymbol{C}[X,Y]$$

mit einer eindeutig bestimmten komplexen Zahl $\lambda \in \boldsymbol{C}$ mit

$$|\lambda| < 1 \ \textit{und}\ |1-\lambda| < 1.$$

Die beiden elliptischen Integrale

$$\omega_1 := \int_{-\infty}^{0} \frac{dx}{\sqrt{x \cdot (x-1) \cdot (x-\lambda)}} \text{ und } \omega_2 := \int_{1}^{\infty} \frac{dx}{\sqrt{x \cdot (x-1) \cdot (x-\lambda)}}$$

definieren ein Gitter

$$\Gamma = \boldsymbol{Z} \cdot \omega_1 + \boldsymbol{Z} \cdot \omega_2, \ \frac{\omega_1}{\omega_2} \in \boldsymbol{H} \text{ oder } \frac{\omega_2}{\omega_1} \in \boldsymbol{H} .$$

Es bleibt zu zeigen: Mit

$$\tau := \frac{\omega_1}{\omega_2}, \text{ o.E. } \tau \in \boldsymbol{H} ,$$

erhält man einen Torus $X = \boldsymbol{C}/\Gamma(\tau)$, der unter der Abbildung von Teil i) biholomorph auf eine zu $E(\boldsymbol{C})$ isomorphe Varietät abgebildet wird; siehe [Sil1986] Chap. VI, q. e. d.

7.18 Toolbeispiel (Tori und elliptische Kurven)

Pari-Beispiel

- MyExamples/Example3

Berechung der elliptischen Kurven zu Tori mit vorgebenen normierten Periodengittern.

8 Skripte ausgewählter Toolbeispiele

Dieses Kapitel enthält die Skripte aller mit den Tools Singular, Macaulay2 und Pari in dieser Vorlesung behandelten Beispiele. Die Skripte der Beispiele für das Tool Surf sind nicht wiedergegeben.

Singular-Skripte

Ein Singular-Skript in der Datei „file“ im Arbeitsverzeichnis von Singular wird mit dem Befehl

< „file“;

ausgeführt.

8.1 Singular-Skript (Neil Parabola)

```
// Singular
// Maps between coordinate rings
print ("============================================================");
LIB "all.lib";


// Coordinate ring of affine line
ring      Q_T      = 0, T, dp;


print ("------------------------------------------------------------");
print ("Neil parabola");
// Neil parabola
ring      Q_XY     = 0, ( X, Y ), dp;
ideal     I1       = Y^2 - X^3 ;
// projection onto x-axis
// as map from Q_T to basering
setring ( Q_XY );
map phi1 = Q_T, X;
print ("Projecting Neil parabola onto x-axis: ");
print ("Ring map is injective? " + print ( is_injective ( phi1, Q_T, I1 ), "%p" ) );
print ("Ring map is surjective? " + print ( is_surjective ( phi1, Q_T, I1 ), "%p" ));
print ("Ring map is finite? " + print (mapIsFinite ( phi1, Q_T, I1), "%p" ));
// projection onto y-axis
map phi2 = Q_T, Y;
print ("Projecting Neil parabola onto y-axis: ");
print ("Ring map is injective? " + print ( is_injective ( phi2, Q_T, I1 ), "%p" ) );
print ("Ring map is surjective? " + print ( is_surjective ( phi2, Q_T, I1 ), "%p" ));
print ("Ring map is finite? " + print (mapIsFinite ( phi2, Q_T, I1), "%p" ));
// Parametrization
print ("Parametrizing Neil parabola: ");
setring ( Q_T );
```

```
map phi3 = Q_XY, T^2, T^3;
print ("Kernel of parametrization: " + alg_kernel ( phi3, Q_XY ) );
print ("Ring map is surjective? " + print ( is_surjective ( phi3, Q_XY ), "%p" ));
print ("Ring map is finite? " + print (mapIsFinite ( phi3, Q_XY), "%p" ));
// Dimension und singularities
setring ( Q_XY );
print ("Neil parabola: ");
print ("Dimension: " + print ( dim ( I1 ), "%p") );
print ("Dimension of singular locus: " + print ( dim_slocus ( I1 ), "%p") );
print ("Singular locus: " + print ( slocus ( I1 ), "%p") );
print ("============================================================");
```

8.2 Singular-Skript *(Hyperbel)*

```
// Singular
// Maps between coordinate rings
print ("============================================================");
LIB "all.lib";

// Coordinate ring of affine line
ring       Q_T      = 0, T, dp;

print ("------------------------------------------------------------");
print ("Hyperbola");
// Hyoerbola
ring       Q_XY   = 0, ( X, Y ), dp;
ideal      I1         = X*Y -  1 ;

// projection onto x-axis
// as map from Q_T to basering
setring ( Q_XY );
map phi1 = Q_T, X;
print ("Projecting hyperbola onto x-axis: ");
print ("Ring map is injective? " + print ( is_injective ( phi1, Q_T, I1 ), "%p" ) );
print ("Ring map is surjective? " + print ( is_surjective ( phi1, Q_T, I1 ), "%p" ));
print ("Ring map is finite? " + print (mapIsFinite ( phi1, Q_T, I1), "%p" ));

// Dimension und Singularitäten
setring ( Q_XY );
print ("Hyperbola: ");
print ("Dimension: " + print ( dim ( I1 ), "%p") );
print ("Dimension of singular locus: " + print ( dim_slocus ( I1 ), "%p") );
print ("Singular locus: " + print ( slocus ( I1 ), "%p") );
```

```
print ("============================================================");
```

8.3 Singular-Skript (Blow-Up)

```
// Singular
// Blow up
print ("============================================================");
LIB "all.lib";
print ("Blow up");
ring      Q_XY   = 0, ( X, Y ), dp;
ring      Q_XYZ = 0, ( X, Y, Z ), dp;
ring      Q_Z      = 0,  Z, dp;
ring      Q_XZ   = 0,  (X, Z ), dp;
setring ( Q_XYZ );
ideal     I1         = Y - Z*X ;
qring     blowUp = I1;
print ("------------------------------------------------------------");
// Dimension und singularities
setring ( Q_XYZ );
print ("Blow up: ");
print ("Dimension: " +                print ( dim ( I1 ), "%p") );
print ("Dimension of singular locus: " +      print ( dim_slocus ( I1 ), "%p") );
print ("------------------------------------------------------------");
// projection along z-axis onto plane
setring ( Q_XYZ );
map phi1 = Q_XY, X, Y;
print ("Projecting blow up onto x-y-plane: ");
print ("Ring map is injective? " +   print ( is_injective   ( phi1, Q_XY, I1 ), "%p" ) );
print ("Ring map is surjective? " + print ( is_surjective ( phi1, I1 ), "%p" ) );
print ("Ring map is finite? " +                print ( mapIsFinite   ( phi1, Q_XY, I1), "%p" ) );
print ("------------------------------------------------------------");
// projection onto exeptional fibre E = z-axis
setring ( Q_XYZ );
map phi2 = Q_Z, Z;
print ("Fibering blow up as line bundle over exeptional fibre: ");
print ("Ring map is injective? " +   print ( is_injective   ( phi2, Q_Z, I1  ), "%p" ) );
print ("Ring map is surjective? " + print ( is_surjective ( phi2, I1 ), "%p" ) );
print ("Ring map is finite? " +                print ( mapIsFinite   ( phi2, Q_Z, I1 ), "%p" ) );
print ("------------------------------------------------------------");
// fibering blow up as line bundle over exeptional fibre
setring ( blowUp );
map phi3 = Q_XZ, X, Z;
print ("Trivialization of line bundle as product: ");
```

```
print ("Ring map is injective? " +   print ( is_injective   ( phi3, Q_XZ ), "%p" ) );
print ("Ring map is surjective? " +  print ( is_surjective ( phi3 ), "%p" ) );
// special treatment to compute finitess of map with quotient ring as target
setring ( Q_XYZ );
map phi31 = Q_XZ, X, Z;
print ("Ring map is finite? " +                print ( mapIsFinite   ( phi31, Q_XZ, I1 ), "%p" ) );
setring ( Q_XZ );
map phi4 = blowUp, X, X*Z, Z;
print ("Inverse map of trivialization: ");
print ("Ring map is injective? " +   print ( is_injective   ( phi4, blowUp ), "%p" ) );
print ("Ring map is surjective? " +  print ( is_surjective ( phi4 ), "%p" ) );
print ("Ring map is finite? " +                print ( mapIsFinite   ( phi4, blowUp ), "%p" ) );
print ("============================================================");
```

8.4 Singular-Skript *(Normalisierung)*

```
// Singular
// Normalization
print ("-----------------------------------------------");
LIB "all.lib";

print ("Normalization of Neil parabola");
// Neil parabola
ring Q_XY = 0, ( X, Y ), dp;
ideal I1 = Y^2 - X^3  ;

// check if ideal is prime
list primaryComponent = primdecGTZ( I1 );
if (          size (primaryComponent)  != 1
   or size ( reduce( primaryComponent[1][2], std( primaryComponent[1][1] ), 1 ) ) != 0 )
{
   print ("Ideal ");
   print ( I1 );
   print (" not prime") ;
   ERROR ("==> Computation cancelled");
}

qring neil_Parabola = std ( I1 );
setring Q_XY;
list nor1 = normal (I1);
show (nor1);
def R = nor1 [1];
setring R;
```

```
qring normalization = std ( norid ) ;
setring R;
normap;
is_bijective ( normap, neil_Parabola );

print ("-------------------------");
print ("Normalization of Whitney umbrella");
ring Q_XYZ = 0, ( X, Y, Z ),dp;
//ideal I2 = Y^2 - Z*X^2;
ideal I2 = X^2*Y - Z^2;
// check if ideal is prime
list primaryComponent = primdecGTZ( I2 );
if (            size (primaryComponent)  != 1
   or size ( reduce( primaryComponent[1][2], std( primaryComponent[1][1] ), 1 ) ) != 0 )
{
   print ("Ideal ");
   print ( I2 );
   print (" not prime") ;
   ERROR ("==> Computation cancelled");
}
qring whitney_Umbrella = std ( I2 );
setring Q_XYZ;
list nor2 = normal (I2);
show (nor2);
def R = nor2 [1];
setring R;
qring normalization = std ( norid ) ;
setring R;
normap;
is_bijective ( normap, whitney_Umbrella );

print ("-------------------------");
print ("Normalization of 5-nodal curve");
// Neil parabola
ring Q_XY = 0, ( X, Y ), dp;
ideal I3 =
   32*X^2 - 2097152*Y^11 + 1441792*Y^9 - 360448*Y^7 + 39424*Y^5 - 1760*Y^3 + 22*Y - 1;

// check if ideal is prime
list primaryComponent = primdecGTZ( I3 );
if (            size (primaryComponent)  != 1
   or size ( reduce( primaryComponent[1][2], std( primaryComponent[1][1] ), 1 ) ) != 0 )
```

```
{
   print ("Ideal ");
   print ( I3 );
   print (" not prime") ;
   ERROR ("==> Computation cancelled");
}

qring fiveNodalCurve = std ( I3 );
setring Q_XY;
list nor1 = normal (I3);
show (nor1);
def R = nor1 [1];
setring R;
qring normalization = std ( norid ) ;
setring R;
normap;
is_bijective ( normap, fiveNodalCurve );

print ("-------------------------");
print ("Normalization of hyperbola");
// Neil parabola
ring Q_XY = 0, ( X, Y ), dp;
ideal I4 = X*Y - 1;

// check if ideal is prime
list primaryComponent = primdecGTZ( I4 );
if (            size (primaryComponent)  != 1
   or size ( reduce( primaryComponent[1][2], std( primaryComponent[1][1] ), 1 ) ) != 0 )
{
   print ("Ideal ");
   print ( I4 );
   print (" not prime") ;
   ERROR ("==> Computation cancelled");
}

qring hyperbola = std ( I4 );
setring Q_XY;
list nor4 = normal (I4);
show (nor4);
def R = nor4 [1];
setring R;
qring normalization = std ( norid ) ;
```

```
setring R;
normap;
is_bijective ( normap, hyperbola );
print ("-----------------------------------------------");
```

8.5 Singular-Skript (Noether Normalisierung)

```
// Singular
// Noether Normalization of surfaces in affine 3-space
print ("-----------------------------------------------");
LIB "all.lib";
list       myExampleList, myExample, noethNorm;
string     myText;
ring       myRing = 0, ( X, Y, Z ), dp;
ideal      myIdeal, coordinateChange, variablesSubring, myIdealNewCoordinates;
int        i, j;
//-------------------------------------------------------
i = 0;
j = 0;
string     myText = "Noether normalization of union of plane with transversal line";
myExample = insert (myExample, myText, j);
j++;
ideal      id1, id2;
// y/z-plane
id1                  = X;
// x-axis
id2                  =  Y, Z;
myIdeal = intersect( id1, id2);
myExample = insert (myExample, myIdeal, j);
myExampleList = insert (myExampleList, myExample, i);
//-------------------------------------------------------
i++;
j = 0;
myText = "Noether normalization of blow up";
myExample = insert (myExample, myText, j);
j++;
myIdeal = Y - Z*X;
myExample = insert (myExample, myIdeal, j);
myExampleList = insert (myExampleList, myExample, i);
//-------------------------------------------------------
for ( i = 1; i <= size (myExampleList); i++ )
{
   print ("-------------------------------------------------------------");
```

```
   myExample              = myExampleList[i];
   print("Example: "                    + print (i, "%s")) ;
   myText                 = myExample[1];
   myIdeal                = myExample[2];
   print ( "Coordinate ring of ambient space: "    + print (myRing, "%s"));
   print ( "Variety defined by ideal: "    + print ( myIdeal, "%s"));
   print ( myText );
   noethNorm              = noetherNormal ( myIdeal );
   // Note wrong order of ideals in Singular online help
   coordinateChange       = noethNorm[1];
   variablesSubring       = noethNorm[2];
   map phi                = basering, coordinateChange;
   myIdealNewCoordinates  = phi ( myIdeal);
   print ("Basering: "                   + print (  basering, "%s"));
   print ("Coordinate change: "          + print (  coordinateChange, "%s"));
   print ("Ideal after coordinate change: "  + print ( myIdealNewCoordinates, "%s"));
   print ("Coordinate ring is finite over polynomial subalgebra with generators: " + print (variablesSubring,
   "%s"));
   print("");
}
print ("-----------------------------------------------");
```

8.6 Singular-Skript *(Primärzerlegung)*

```
// Singular
// Primary decomposition of ideals
print ("-----------------------------------------------");
print ("Primary decomposition of ideals");
LIB "all.lib";

print ("2 dimensions -------------------------");
// ring of polynomials over field of rationals
ring Q_xy = 0,(x,y),lp;
short = 0;

int i;
int j;
list primaryComponent;

list idealList;

// plane coordinate axes
ideal I1 = x*y;
idealList = insert (idealList, I1, 0);
```

```
// Two lines
ideal I2 = x^2 - 1;
idealList = insert (idealList, I2, 1);

// Double lines
ideal I3 = x^2 ;
idealList = insert (idealList, I3, 2);

// Embedded component
ideal I4 = x^2, x*y;
idealList = insert (idealList, I4, 3);

// Loop through different ideals and output their primary decomposition
for ( i = 1; i <= size (idealList); i++ )
{
   primaryComponent = primdecGTZ( idealList[i] );
   print ("Ideal: <" + print(idealList[i],"%") + ">" );
   for ( j = 1; j <= size (primaryComponent); j++ )
   {
         print (    "Primary component: <"   + print( primaryComponent[j][1], "%") + ">"
         +          " with radical: <"       + print( primaryComponent[j][2], "%") + ">" );
   }
   print("");
}

print ("3 dimensions -------------------------");
// Plane and transversal line

ring Q_xyz = 0,(x,y,z ),lp;
int i;
int j;
list primaryComponent;

list idealList;

ideal I5 = x*y, x*z;
idealList = insert (idealList,  I5, 0);

ideal I6 = (y^2 - x^3)*y - z^2;
idealList = insert (idealList,  I6, 1);
```

```
// Loop through different ideals and output their primary decomposition
for ( i = 1; i <= size (idealList); i++ )
{
   primaryComponent = primdecGTZ( idealList[i] );
   print ("Ideal: <" + print(idealList[i],"%") + ">" );
   for ( j = 1; j <= size (primaryComponent); j++ )
   {
         print (    "Primary component: <" + print( primaryComponent[j][1], "%") + ">"
         +          " with radical: <"          + print( primaryComponent[j][2], "%") + ">" );
   }
   print("");
}
print ("-----------------------------------------------");
```

8.7 Singular-Skript *(Hilbert Polynom)*

```
// Singular
LIB "all.lib";
print ("-----------------------------------------------");
print ("Hilbert polynomial: Start");
print ("Hilbert polynomials of homogenized ideals");
int       i;
intvec    hilbPolynomial;
string    output = "";
//---------------------------------------------------------------
ring      Q_T0XY                  = 0, (T0, X,Y), Ds;
ideal     idealParabola     = Y - X^2;
ideal     homogIdeal;

homogIdeal                = homog ( idealParabola, T0 );
hilbPolynomial            = hilbPoly ( homogIdeal );

print     ("Ideal : <" + print ( idealParabola, "%") + "> of ring "  + print (basering, "%") );
print     ("has homogenization: " + print (homogIdeal, "%") );

output = "with Hilbert polynomial: ";
for ( i = 1; i <= size (hilbPolynomial); i++ )
{
   if ( i != 1) { output = output + " + "; };
   output = output + print ( hilbPolynomial[i], "%") + "*T^^" + print ( i-1, "%");
}
print ( print ( output, "%") );
print ("-----------------------------------------------------");
```

```
ideal     idealHyperbola   = X*Y - 1;

homogIdeal                 = homog ( idealHyperbola, T0 );
hilbPolynomial             = hilbPoly ( homogIdeal );

print     ("Ideal : <" + print ( idealHyperbola , "%") + "> of ring "  + print (basering, "%") );
print     ("has homogenization: " + print (homogIdeal, "%") );

output = "with Hilbert polynomial: ";
for ( i = 1; i <= size (hilbPolynomial); i++ )
{
   if ( i != 1) { output = output + " + "; };
   output = output + print ( hilbPolynomial[i], "%") + "*T^^" + print ( i-1, "%");
}
print ( print ( output, "%") );
print ("-----------------------------------------------------");
ideal     idealCoordinateAxes       = X*Y;

homogIdeal                 = homog ( idealCoordinateAxes, T0 );
hilbPolynomial             = hilbPoly ( homogIdeal );

print     ("Ideal : <" + print ( idealCoordinateAxes , "%") + "> of ring "  + print (basering, "%") );
print     ("has homogenization: " + print (homogIdeal, "%") );

output = "with Hilbert polynomial: ";
for ( i = 1; i <= size (hilbPolynomial); i++ )
{
   if ( i != 1) { output = output + " + "; };
   output = output + print ( hilbPolynomial[i], "%") + "*T^^" + print ( i-1, "%");
}
print ( print ( output, "%") );
print ("-----------------------------------------------------");
ideal     idealNeilparabola          = Y^2 - X^3;

homogIdeal                 = homog ( idealNeilparabola , T0 );
hilbPolynomial             = hilbPoly ( homogIdeal );

print     ("Ideal : <" + print ( idealNeilparabola, "%") + "> of ring "  + print (basering, "%") );
print     ("has homogenization: " + print (homogIdeal, "%") );

output = "with Hilbert polynomial: ";
for ( i = 1; i <= size (hilbPolynomial); i++ )
```

```
{
   if ( i != 1) { output = output + " + "; };
   output = output + print ( hilbPolynomial[i], "%") + "*T^^" + print ( i-1, "%");
}
print ( print ( output, "%") );
print ("-----------------------------------------------------");
ring     Q_T0XYZ        = 0, (T0, X,Y,Z), Ds;
ideal    idealBlowUp    = Y-Z*X;
ideal    homogIdeal;

homogIdeal              = homog ( idealBlowUp , T0 );
hilbPolynomial          = hilbPoly ( homogIdeal );

print    ("Ideal : <" + print ( idealBlowUp, "%") + "> of ring "  + print (basering, "%") );
print    ("has homogenization: " + print (homogIdeal, "%") );

output = "with Hilbert polynomial: ";
for ( i = 1; i <= size (hilbPolynomial); i++ )
{
   if ( i != 1) { output = output + " + "; };
   output = output + print ( hilbPolynomial[i], "%") + "*T^^" + print ( i-1, "%");
}
print ( print ( output, "%") );
print ("-----------------------------------------------------");
ring     Q_T0XYZ        = 0, (T0, X,Y,Z), Ds;
ideal    idealReducible = X*Y, X*Z;
ideal    homogIdeal;

homogIdeal              = homog ( idealReducible , T0 );
hilbPolynomial          = hilbPoly ( homogIdeal );

print    ("Ideal : <" + print ( idealReducible, "%") + "> of ring "  + print (basering, "%") );
print    ("has homogenization: " + print (homogIdeal, "%") );

output = "with Hilbert polynomial: ";
for ( i = 1; i <= size (hilbPolynomial); i++ )
{
   if ( i != 1) { output = output + " + "; };
   output = output + print ( hilbPolynomial[i], "%") + "*T^^" + print ( i-1, "%");
}
print ( print ( output, "%") );
print ("");
```

```
print ("Hilbert polynomial: End");
print ("-----------------------------------------------");
```

Macaulay2-Skripte

Ein Macaulay2-Skript in der Datei „file“ im Arbeitsverzeichnis von Macaulay2 wird mit dem Befehl

load „file“

ausgeführt.

8.8 Macaulay2-Skript (Divisionsalgorithmus)

```
-- Macaulay2
-- Division in ring of polynomials with rest

division = ( numerator, listDenom ) -> (
   targetRing       := class numerator;
   use targetRing;
   k                := #listDenom;
   i                := 0;
   debug                    := false;
   if ( debug == true ) then
   (
         print ("Division: Start");
         << "Division: " << numerator << " :" << endl;
         i = 0;
         while i < k do
         (
               << "denominator " << i << ": " << listDenom_i << endl;
               i = i + 1;
         );
   );

   p                := numerator;
   r                := 0;
   quotient       := new MutableHashTable;

   -- initialize hash table of quotients
   i = 0;
   while i < k do
   (
         quotient#i          = 0;
         i                   = i + 1;
   );

   -- check if leading term of reduced dividend is divisible by leading term of one of divisors
```

```
while ( p != 0 ) do
(
        i = 0;
        divisionOccurred = false;

        -- loop through divisors
        while ( ( i < k ) and ( divisionOccurred == false ) ) do
        (
                -- check if  leading term of current divisor divides
                -- leading term of reduced dividend
                if ( denominator ( leadTerm p / leadTerm listDenom_i  ) == 1 ) then
                (
                        -- subscript quotient with index of denominator
                        if ( debug == true ) then
                        (
                                << "Division: Divide by " << listDenom_i << endl;
                                << "Division: p: " << p << "; listDenom" << i << ": ";
                                << listDenom_i << endl;
                        );
                        a = substitute (leadTerm p / leadTerm listDenom_i, targetRing );
                        quotient#i = quotient#i + a;
                        p = substitute ( p - a * listDenom_i, targetRing );
                        divisionOccurred = true;
                )
                else
                (
                        i = i + 1 ;
                );
        );
        if ( divisionOccurred == false ) then
        (
                -- move leading term of current dividend to rest
                r = r + leadTerm p;
                p = p - leadTerm p;
        );
);

-- output result
<< "(" << numerator << ") : "      << endl;
<< listDenom << " = " << endl   << endl;
i = 0;
while  i < k do
```

```
   (
        << "+ (" << quotient#i << ") * (" << listDenom_i << ")" << endl;
        i = i + 1;
   );
   << endl << "Rest: " << r << endl;

   if ( debug == true ) then
   (
        << "Division: End" << endl;
   );
);
<< "-----------------------------------" << endl;
R = QQ [ X,Y, MonomialOrder => Lex];
-- example 1
 g1 = X*Y + 1;
 g2 = Y^2 - 1;
 f = X*Y^2 - X;
 lden = { g2, g1 };
division ( f, lden);
<< "-----------------------------------" << endl;
-- example 2
lden = { g1, g2 };
division ( f, lden);
<< "-----------------------------------" << endl;
-- example 3
-- g1 = X*Y + 1;
-- g2 = Y + 1;
-- f = X*Y^2 + 1;
-- lden = {g1, g2};
-- division ( f, lden);
-- << "-----------------------------------" << endl;
-- example 4
-- g1 = X*Y - 1;
-- g2 = Y^2 - 1;
-- f = X^2*Y + X*Y^2 + Y^2;
-- lden = {g1, g2};
-- division ( f, lden);
-- << "-----------------------------------" << endl;
```

8.9 Macaulay2-Skript *(Gröbner Basis)*

```
-- Macaulay2
-- Computation of Groebner bases
```

```
-- << "-----------------------------------" << endl;
R = QQ [ X,Y, MonomialOrder => Lex];
-- example 1
g1 = X*Y + 1;
g2 = Y^2 - 1;
I1 = ideal (  g1, g2 );
G1 = gb I1;
<<  I1 << " has Groebner base  with respect to Lex with "<<  endl;
<< rank source gens G1 << " elements:" << endl;
<< transpose gens G1 << endl;
<< "-----------------------------------" << endl;
R = QQ [ X,Y, Z, MonomialOrder => Lex];
-- example 2
g3 = X^5 + Y^4 + Z^3 - 1;
g4 = X^3 + Y^2 + Z^2 - 1;
I2 = ideal (  g3, g4 );
G2 = gb I2;
<<  I2 << " has Groebner base  with respect to Lex with "<<  endl;
<< rank source gens G2 << " elements:" << endl;
<< transpose gens G2 << endl;
<< "-----------------------------------" << endl;
R = QQ [ X,Y, Z, MonomialOrder => GRevLex];
-- example 3
g5 = X^5 + Y^4 + Z^3 - 1;
g6 = X^3 + Y^2 + Z^2 - 1;
I3 = ideal (  g5, g6 );
G3 = gb I3;
<<  I3 << " has Groebner base  with respect to GRevLex with "<<  endl;
<< rank source gens G3 << " elements:" << endl;
<< transpose gens G3 << endl;
<< "-----------------------------------" << endl;
```

8.10 Macaulay2-Skript (Kern von Algebra-Morphismen)

```
-- Macaulay2
-- Twisted cubic curve
-- Define parametrization of twisted cubic curve by map
-- from affine line to affine space
<< "-------------------------------------------------------" << endl;
<< "Twisted cubic curve: Start " << endl;


-- Graph of twisted cubic curve
-- Monomial order for graph
```

```
R3 = QQ [ T, X,Y,Z, MonomialOrder => Lex ]
short = 0;
J = ideal ( X - T, Y -  T^2, Z - T^3 );
<< "Graph:" << J << endl;
<< "has Groebner base (Lex): " << gb (J) << endl;

-- Suitable elimination order for graph
R4 = QQ [ T, X,Y,Z, MonomialOrder => Eliminate 1 ]
short = 0;
J = substitute ( J, R4);
<< "Graph:" << J << endl;
<< "has Groebner base (Eliminate 1): " << gb (J) << endl;
<< endl;

-- Affine line
R1 = QQ[ T, MonomialOrder => Lex ]
-- Affine space
R2 = QQ [ X, Y, Z, MonomialOrder => Lex ]

phi = map ( R1, R2, { T, T^2, T^3 } )
<< phi << endl;
idealCubicCurve = kernel phi;

<<  "Twisted CubicCurve has ideal: " <<idealCubicCurve << endl;
<< "Is " <<  idealCubicCurve << " equal to " <<  ideal ( X^2 - Y, Z - X^3 ) << "? "
<< idealCubicCurve ==  ideal ( X^2 - Y, Z - X^3 ) << endl;

<< "Twisted cubic curve: End " << endl;
<< "-------------------------------------------------------" << endl;
```

8.11 Macaulay2-Skript (Komplement affiner Varietäten)

```
-- Macaulay2
-- Quotient of ideals
<< "-----------------------------------------------------------------------" << endl;
<<  "Quotient of ideals. Start" << endl;
R1 = QQ[ X,Y,Z, MonomialOrder => Lex ];
<< "Base ring: " << describe R1 << endl;
<< endl;
-- ideal of y-z plane union x-axis
I = ideal ( X*Y, X*Z );

-- ideal of x-axis
```

```
J =  ideal ( Y, Z );


<< I << ": " << J << " = " << I:J << endl;
<< endl;
<<  "Quotient of ideals. End" << endl;
<< "--------------------------------------------------------------------------" << endl;
```

8.12 Macaulay2-Skript *(Reguläre Abbildung zwischen projektiven Varietäten)*

```
-- Regular maps between projective varieties
<< "-----------------------------------------------------------------------------------------------" << endl;
<< "Twisted cubic. Start" << endl;
R1 = QQ [t0, t1];
R2 = QQ [x0, x1, x2, x3];
short = 0
phi1 = map ( R1, R2, { t0^3, t0^2*t1, t0*t1^2, t1^3 } )
idealTwistedCubic = kernel phi1;
<< "image of " << phi1 <<  " is defined by " << endl;
<< idealTwistedCubic << endl;
matr1 = matrix {{ x0, x1, x2 }, { x1, x2, x3}};
idealMinors1 =  minors (2, matr1);
<< "Is " << idealTwistedCubic << " equal to minors " << endl;
<<  idealMinors1 << "? " << idealTwistedCubic ==  idealMinors1 << endl;
<< "Twisted cubic. End" << endl;
<< "-----------------------------------------------------------------------------------------------" << endl;
<< "Veronese embedding. Start" << endl;
R3 = QQ [t0, t1, t2];
R4 = QQ [x0, x1, x2, x3, x4, x5];
short = 0
phi2 = map ( R3, R4, { t0^2, t0*t1, t0*t2, t1^2, t1*t2, t2^2 } )
idealVeroneseMap = kernel phi2;
<< "image of " << phi2 << " is defined by " << endl;
<< idealVeroneseMap << endl;
matr2 = matrix {{ x0, x1, x2, x3, x4}, { x1, x2, x3,  x4, x5}};
idealMinors2 =  minors (2, matr2);
<< "Is " << idealVeroneseMap << " equal to minors " << endl;
<<  idealMinors2 << "? " << idealVeroneseMap == idealMinors2 << endl;
<< "Veronese embedding. End" << endl;
<< "-----------------------------------------------------------------------------------------------" << endl;
```

8.13 Macaulay2-Skript *(Hilbert Polynom)*

```
-- Macaulay2
-- Hilbert polynomial
```

```
-- base field
KK = QQ;
-- Macaulay2
-- Hilbert polynomial
<< "Twisted cubic: Start" << endl;
R1 = KK [ T0, T1 ];
R2 = KK [ X0, X1, X2, X3 ];
phiCubic = map ( R1, R2, { T0^3, T0^2*T1, T0*T1^2, T1^3 } );
describe phiCubic;
idealCubic = kernel phiCubic;
coordinateRingCubic = R2 / idealCubic;
-- evaluate projective Hilbert polynomial
-- the values dim and degree refering to the projective variety
cubicHilbertPolynomial = hilbertPolynomial (coordinateRingCubic);
cubicDimension = dim (cubicHilbertPolynomial);
cubicDegree =  degree (cubicHilbertPolynomial);
cubicArithmeticGenus = (-1)^cubicDimension * ( (cubicHilbertPolynomial 0) - 1);

<< "Twisted cubic curve has" << endl;
<< "ideal: " << transpose mingens idealCubic << endl;
<< "in ring: " << describe R2 << endl;
<< "Hilbert polynomial: " << cubicHilbertPolynomial << endl;
<< "dimension: " << cubicDimension << endl;
<< "degree: " << cubicDegree << endl;
<< "arithmetic genus: " << cubicArithmeticGenus << endl;

<< "Twisted cubic: End" << endl << endl;
<< "------------------------------------------------------------------------" << endl;
<< "Veronese surface: Start" << endl;
R1 = KK [ T0, T1, T2 ];
R2 = KK [ X0, X1, X2, X3, X4, X5 ];
phiVeronese = map ( R1, R2, { T0^2, T0*T1, T0*T2, T1^2, T1*T2, T2^2 } );
describe phiVeronese;
idealVeronese = kernel phiVeronese;
coordinateRingVeronese = R2 / idealVeronese;
veroneseHilbertPolynomial = hilbertPolynomial (coordinateRingVeronese);
veroneseDimension = dim (veroneseHilbertPolynomial);
veroneseDegree =  degree (veroneseHilbertPolynomial);
veroneseArithmeticGenus = (-1)^veroneseDimension * ( (veroneseHilbertPolynomial 0) - 1);

<< "Veronese surface has" << endl;
<< "ideal: " << transpose mingens idealVeronese << endl;
```

```
<< "in ring: " << describe R2 << endl;
<< "Hilbert polynomial: " << veroneseHilbertPolynomial << endl;
<< "dimension: " << veroneseDimension << endl;
<< "degree: " << veroneseDegree << endl;
<< "arithmetic genus: " << veroneseArithmeticGenus << endl;

<< "Veronese surface: End" << endl << endl;
<< "----------------------------------------------------------------------" << endl;
<< "Segre embedding: Start" << endl;
R1 = KK [ X0, X1 ];
R2 = KK [ Y0, Y1 ];
domainOfDefinition = R1 ** R2;
R3 = KK [ Z00, Z01, Z10, Z11 ];
phiSegre = map ( domainOfDefinition, R3, { X0*Y0, X0*Y1, X1*Y0, X1*Y1 } );
describe phiSegre;
idealSegre = kernel phiSegre;
coordinateRingSegre = R3 / idealSegre;
segreHilbertPolynomial = hilbertPolynomial (coordinateRingSegre);
segreDimension = dim (segreHilbertPolynomial);
segreDegree =  degree (segreHilbertPolynomial);
segreArithmeticGenus = (-1)^segreDimension * ( (segreHilbertPolynomial 0) - 1);

<< "Segre embedding has" << endl;
<< "ideal: " << transpose mingens idealSegre << endl;
<< "in ring: " << describe R3 << endl;
<< "Hilbert polynomial: " << segreHilbertPolynomial << endl;
<< "dimension: " << segreDimension << endl;
<< "degree: " << segreDegree << endl;
<< "arithmetic genus: " << segreArithmeticGenus << endl;

<< "Segre embedding: End" << endl << endl;
<< "----------------------------------------------------------------------" << endl;
<< "Elliptic Curve: Start" << endl;
R2 = KK [ X0, X1, X2 ];
idealEllipticCurve = ideal ( X0*X2^2 - X1^3 - X0^3 );
coordinateRingElliptic = R2 / idealEllipticCurve;
ellipticHilbertPolynomial = hilbertPolynomial (coordinateRingElliptic);
ellipticDimension = dim (ellipticHilbertPolynomial);
ellipticDegree =  degree (ellipticHilbertPolynomial);
ellipticArithmeticGenus = (-1)^ellipticDimension * ( (ellipticHilbertPolynomial 0) - 1);

<< "Elliptic Curve has" << endl;
```

```
<< "ideal: " << transpose mingens idealEllipticCurve << endl;
<< "in ring: " << describe R2 << endl;
<< "Hilbert polynomial: " << ellipticHilbertPolynomial << endl;
<< "dimension: " << ellipticDimension << endl;
<< "degree: " << ellipticDegree << endl;
<< "arithmetic genus: " << ellipticArithmeticGenus << endl;

<< "Elliptic Curve: End" << endl << endl;
<< "------------------------------------------------------------------" << endl;
```

Pari-Skripte

Ein Pari-Skript in der Datei <file> im Arbeitsverzeichnis von Pari wird mit dem Befehl

\r <file>

ausgeführt. Dabei wird der Name der Datei, ohne Hochkommata eingegeben.

8.14 Pari-Skript (Elliptische Kurve und Diskriminante)

```
/* Pari
/* Elliptic curves in Weierstrass normal form y^2 = x^3 + A*x + B */
default(format, "f0.3" );
print ("------------------------------------------------------------------");
A = 1;
B = 0;
E = [ 0, 0, 0, A, B];
E = ellinit (E);
print ( "Elliptic curve:  y^2 = x^3 + ",E[4], "*x + ",E[5] );
print ( "Discr.: ", E.disc,);
print ( "j(E): ", E.j );
print ( "Root 1: ", E[14][1] );
print ( "Root 2: ", E[14][2] );
print ( "Root 3: ", E[14][3] );
print ( "Order of torsion group: ", elltors(E)[1] );
print ( "period tau: ", E[16]/E[15] );

p = [ E[14][1], 0];
print ( "Rational point p = ", p );
print ( "p + p = ", elladd(E, p, p) );
print ( p, " has order ", ellorder(E, p) );

print ("------------------------------------------------------------------");
A = 0;
B = 1;
E = [ 0, 0, 0, A, B];
E = ellinit (E);
print ( "Elliptic curve:  y^2 = x^3 + ",E[4], "*x + ",E[5] );
print ( "Discr.: ", E.disc,);
print ( "j(E): ", E.j );
print ( "Root 1: ", E[14][1] );
print ( "Root 2: ", E[14][2] );
print ( "Root 3: ", E[14][3] );
print ( "Order of torsion group: ", elltors(E)[1] );
```

```
print ( "period tau: ", E[16]/E[15] );
p = [ E[14][1], 0];
print ( "Rational point p = ", p );
print ( "2p = ", ellpow(E, p, 2) );
print ( p, " has order ", ellorder(E, p) );

q = [2, 3];
print (  q, " is on curve? ", ellisoncurve(E, q) );
print ( "2q = ", ellpow(E, q, 2) );
print ( "3q = ", ellpow(E, q, 3) );
print ( "4q = ", ellpow(E, q, 4) );
print ( "5q = ", ellpow(E, q, 5) );
print ( "6q = ", ellpow(E, q, 6) );
print ( q, " has order ", ellorder(E, q) );

print ("----------------------------------------------------------------------");
```

8.15 Pari-Skript *(Elliptische Kurve mit positivem Rang)*

```
/* Pari
/* Elliptic curve with positive rank
/* Weierstrass normal form y^2 = x^3 + A*x + B */
default(format, "f0.3" );
print ("----------------------------------------------------------------------");
A = -7;
B = 25/4;
E = [ 0, 0, 0, A, B];
E = ellinit (E);
print ( "Elliptic curve:  y^2 = x^3 + ",E[4], "*x + ",E[5] );
print ( "Discr.: ", E.disc,);
print ( "j(E): ", E.j );
print ( "Root 1: ", E[14][1] );
print ( "Root 2: ", E[14][2] );
print ( "Root 3: ", E[14][3] );
print ( "Order of torsion group: ", elltors(E)[1] );
print ( "period tau: ", E[16]/E[15] );



{
   for ( x = -3, 1000,
            s = ellordinate ( E, x);
            if ( length (s),
                                 y = s[1];
```

```
                        print ( [x,    y    ] )  ;
                        print ( [-x, -y-1 ] ) ;
        )
  );
}
print ("-----------------------------------------------------------------");
```

8.16 Pari-Skript (Tori und elliptische Kurven)

```
/* Pari */
/* Tori and corresponding elliptic curves in Weierstrass normal form y^2 = x^3 + A*x + B */
default(format, "f0.6" );
print ("-----------------------------------------------------------------");
/* create lattice */
omega_1 = 1;
omega_2 = -1/2 + (1/2)*I;
l = [ omega_1 , omega_2 ];
tau = omega_2/omega_1;
print ( "Torus belonging to lattice with periods");
print ("Omega_1: " polcoeff ( l, 1) );
print ("Omega_2: " polcoeff ( l, 2) );
print ( "j of lattice: ", ellj ( tau ) ) ;
print();

/* attach elliptic curve to torus */
wp_torus =  ellwp (l);
eisenstein_2 = 1/3 * polcoeff ( wp_torus, 2 );
eisenstein_3 = 1/5 * polcoeff ( wp_torus, 4 );
A_comp = -15 * eisenstein_2;
B_comp = -35 * eisenstein_3;
print ("Attached elliptic curve has Weierstrass normal form Y^2=X^3 + A*X + B with");
print ("A: ", A_comp);
print ("B: ", B_comp);

/* compare with elliptic curve */
A = 1;
B = 0;
E = [ 0, 0, 0, A, B];
E = ellinit (E);
print ();
print ( "Elliptic curve:  Y^2 = X^3 + ",E[4], "*X + ",E[5] );
print ( "has period: ", E[16]/E[15] );
print ( "j of curve: ", E.j  ) ;
```

```
print ("-----------------------------------------------------------------");
/* create lattice */
omega_1 = 1;
omega_2 = -1/2 + 0.288675*I;
l = [ omega_1 , omega_2 ];
tau = omega_2/omega_1;
print ( "Torus belonging to lattice with periods");
print ("Omega_1: " polcoeff ( l, 1) );
print ("Omega_2: " polcoeff ( l, 2) );
print ( "j of lattice: ", ellj ( tau ) ) ;
print();

/* attach elliptic curve to torus */
wp_torus =  ellwp (l);
eisenstein_2 = 1/3 * polcoeff ( wp_torus, 2 );
eisenstein_3 = 1/5 * polcoeff ( wp_torus, 4 );
A_comp = -15 * eisenstein_2;
B_comp = -35 * eisenstein_3;
print ("Attached elliptic curve has Weierstrass normal form Y^2=X^3 + A*X + B with");
print ("A: ", A_comp);
print ("B: ", B_comp);

/* compare with elliptic curve */
A = 0;
B = 1;
E = [ 0, 0, 0, A, B];
E = ellinit (E);
print ();
print ( "Elliptic curve:  Y^2 = X^3 + ",E[4], "*X + ",E[5] );
print ( "has period: ", E[16]/E[15] );
print ( "j of curve: ", E.j  ) ;

print ("-----------------------------------------------------------------");
/* create lattice */
omega_1 = 1;
omega_2 = 0.713227*I;
l = [ omega_1 , omega_2 ];
tau = omega_2/omega_1;
print ( "Torus belonging to lattice with periods");
print ("Omega_1: " polcoeff ( l, 1) );
print ("Omega_2: " polcoeff ( l, 2) );
```

```
print ( "j of lattice: ", ellj ( tau ) ) ;
print();

/* attach elliptic curve to torus */
wp_torus =  ellwp (l);
eisenstein_2 = 1/3 * polcoeff ( wp_torus, 2 );
eisenstein_3 = 1/5 * polcoeff ( wp_torus, 4 );
A_comp = -15 * eisenstein_2;
B_comp = -35 * eisenstein_3;
print ("Attached elliptic curve has Weierstrass normal form Y^2=X^3 + A*X + B with");
print ("A: ", A_comp);
print ("B: ", B_comp);

/* compare with elliptic curve */
A = -7;
B = 25/4;
E = [ 0, 0, 0, A, B];
E = ellinit (E);
print ();
print ( "Elliptic curve:  Y^2 = X^3 + ",E[4], "*X + ",E[5] );
print ( "has period: ", E[16]/E[15] );
print ( "j of curve: ", E.j , " rounded as ", 1.0 * E.j) ;

print ("-----------------------------------------------------------------");
```

9 Literatur

Kommutative Algebra

[AM1969] *Atiyah, M. F.; Macdonald, I. G.*: Introduction to Commutative Algebra. Addison Wesley, Reading 1969

[BW1998] *Becker, Thomas; Weispfenning, Volker*: Gröbner Bases: A Computational Approach to Commutative Algebra. Springer, Berlin et al. 1998

[Buc1998] *Buchberger, Bruno*: Introduction to Gröbner Bases. In: *Buchberger, Bruno; Winkler, Franz* (Hrsg.): Gröbner Bases and Appplications. Cambridge University Press, Cambridge 1998

[Eis1995] *Eisenbud, David*: Commutative algebra with a view toward algebraic geometry. Springer, New York 1995

Algebraische Geometrie

[CLO1997] *Cox, David; Little, John; O'Shea, Donal*: Ideals, Varieties, and Algorithms. Springer, New York 1997

[CLO1998] *Cox, David; Little, John; O'Shea, Donal*: Using Algebraic Geometry. Springer, New York 1998

[Die1974] *Dieudonné, Jean*: Cours de Géométrie Algébrique. Presses Universitaires de France, 1974

[EH2000] *Eisenbud, David; Harris, Joseph*: The Geometry of Schemes. Springer, New York 2000

[Har1992] *Harris, Joseph*: Algebraic Geometry. A first course. Springer, New York 1992

[Has 1977] *Hartshorne, Robin*: Algebraic Geometry. Springer, New York et al. 1977

[Mum1976] *Mumford, David*: Algebraic Geometry I. Complex Projective Varieties. Springer Berlin et al. 1976

[Rei 1988] *Reid, Miles*: Undergraduate Algebraic Geometry. Cambridge University Press, Cambridge 1988

[Sch2003] *Schenck, Henry*: Computational Algebraic Geomery. Cambridge University Press, Cambridge 2003

Zahlentheorie

[CS1986] *Cornell, Gary; Silverman, Joseph* (Hrsg.): Arithmetic Geometry. Springer, New York 1986

[CSS1995] *Cornell, Gary; Silverman, Joseph; Stevens, Glenn* (Hrsg.): Modular Forms and Fermat's Last Theorem. Springer, New York 1997

[Fal1984] *Faltings, Gerd*: Die Vermutungen von Tate und Mordell. Jahresber. Deutsch. Math.-Verein. Vol. 86 (1984), p. 1-13

[Fal1995] *Faltings, Gerd*: Der Beweis der Fermat-Vermutung durch R. Taylor und A. Wiles. Mitteilungen der Deutsch. Math.-Verein. Heft 2 (1995), p. 6-8

[For1996] *Forster, Otto*: Algorithmische Zahlentheorie. Vieweg, Braunschweig/Wiesbaden 1996

[Fre1984] *Frey, Gerhard*: Elementare Zahlentheorie. Vieweg, Braunschweig/Wiesbaden 1984

[IR1982] *Ireland, Kenneth; Rosen, Michael*: A Classical Introduction to modern Number Theory. Springer, New York 1982

[Maz1993] *Mazur, Barry*: On the Passage from Local to Global in Number Theory. Bulletin of the American Mathematical Society, Vol. 29 (1993)

[Ser1973] *Serre, Jean-Pierre*: A Course in Arithmetic. Sprnger. New York 1973

[Sil1986] *Silvermann, Joseph*: The Arithmetic of Elliptic Curves. Springer, New York 1986

Riemannsche Flächen

[Ahl1966] *Ahlfors, Lars*: Complex Analysis. An Introdution to the Theory of Analytic Functions of one Complex Variable. McGraw-Hill, Tokyo 1966

[For1977] *Forster, Otto*: Riemannsche Flächen. Springer, Berlin 1977

[Gun1972] *Gunning, Robert*: Vorlesungen über Riemannsche Flächen. Bibliographisches Institut, Heidelberg 1972

[Wey1913] *Weyl, Hermann*: Die Idee der Riemannschen Fläche. Nachdruck der Originalausgabe von 1913. Teubner, Leipzig 1977

Tools

9.1 Surf

Surf: http://surf.sourceforge.net/doc/manual.html

9.2 Singular

SINGULAR: http://www.singular.uni-kl.de

[GP2002] *Greuel, Gert-Martin; Pfister, Gerhard*: A Singular Introduction to Commutative Algebra. Springer, Berlin et al. 2002

9.3 Macaulay2

Macaulay2: http://www.math.uiuc.edu/Macaulay2

[EGS2002] *Eisenbud, David; Grayson, Daniel; Stillman, Michael; Sturmfels, Bernd* (Eds.): Computations in Algebraic Geometry with Macaulay2. Springer, Berlin et al. 2002

9.4 Pari

PARI. http://www.parigp-home.de

[COH1993] *Cohen, Henri*: A course in Computational Algebraic Number Theory. Springer, Berlin et al. 1993

[COH2000] *Cohen, Henri*: Advanced topics in Computational Algebraic Number Theory. Springer, Berlin et al. 2000