



Stirnradverzahnung mit Freiform-Kontaktweg

Bachelor-Thesis

an der Universität Passau
Fakultät für Informatik und Mathematik

in Kooperation mit dem FORWISS Passau

vorgelegt von
Philipp Dühring

März 2015

Betreuer: Prof. Dr. rer. nat. Tomas Sauer

Danksagung

Ich danke zu aller erst meinem Betreuer, Herrn Prof. Dr. Tomas Sauer und seinem Mitarbeiter Dr. Florian Zimmermann für die umfassende Betreuung, die mir während dieser Arbeit zuteil wurde. Meine Fragen stießen immer auf ein offenes Ohr und wurden umgehend beantwortet.

Außerdem möchte ich mich bei meiner Partnerin Franziska Gonschorek bedanken, die es geschafft hat, mich in schwierigen Momenten der Ideenlosigkeit immer wieder aufzubauen und zu motivieren.

Inhaltsverzeichnis

Inhaltsverzeichnis	2
1 Vorwort	4
2 Einführung	5
2.1 Splines	5
2.1.1 Berechnung der Spline-Kurve	5
2.1.2 Eigenschaften von Splines	7
2.1.3 Ableitung der Spline-Kurve	9
2.2 Zahnräder	10
2.2.1 Stirnräder	10
2.2.2 Historisches	10
2.2.3 Begrifflichkeiten	12
3 Vom Spline zum fertigen Zahnrad	13
3.1 Umsetzung der Splines	13
3.1.1 Knotenvektor	13
3.1.2 Fixierung eines Punktes auf der Spline-Kurve	15
3.2 Berechnungen mit vorgegebener Übersetzung und Kontaktweg	17
3.3 Berechnungen mit vorgegebener Übersetzung und Profillinie	19
3.4 Für die Graphik - Konstruktion von Zahnfuß und -kopf	20
3.4.1 Zahnfuß	21
3.4.2 Zahnkopf	23
3.5 Das Programm	25
3.5.1 Entwicklungsumgebung	25
3.5.2 Aussehen und Funktion des GUI	25
4 Zusammenfassung und Ausblick	30
A Literatur	31
B Abbildungsverzeichnis	32

C	Anhang	34
C.1	Ausführliches Beispiel zur Spline-Berechnung	34
C.2	Ausführliches Beispiel zur Ableitung eines Spline	37
C.3	Nachbau einer Evolventenverzahnung	39
C.4	„Teufel-Zahn“ - Eine weniger sinnvolle Verzahnung	40

1 Vorwort

Diese Bachelorarbeit ist im Rahmen einer HiWi-Anstellung am FORWISS Passau entstanden. Der dort angestellte Dr. Florian Zimmermann entwickelte in seiner Dissertation eine neue Zahnradgeometrie für Stirnräder in Kooperation mit der Siemens AG.

Meine Aufgabe war es, ein JAVA Programm zu schreiben, welches aus eingegebenen Zahnradgrößen die benötigten Ausgabeformate liefert. Hierbei ist durch die Eingaben der Kontaktweg zwischen zwei Zahnflanken jedoch starr vorgegeben, nämlich durch eine Gerade und je zwei kongruenten Kreisbögen.

In dieser Bachelorarbeit wird nun die Fragestellung erläutert, was passiert, wenn man sich von der Vorgabe eines starren Kontaktweges löst und diesen durch Knoten und dazwischen stattfindender Spline-Interpolation darstellbar macht.

Die Berechnung des Kontaktweges und der daraus resultierenden Zahnflanken sollte aber nicht jedes Mal selbst aufwändig per Hand berechnet werden, sondern es soll auch ein Programm entstehen, welches es dem Benutzer ermöglicht, mit Hilfe einfacher Maus- und Tastatur-Interaktionen den Kontaktweg zu zeichnen und die mit den entsprechenden Daten erstellten Zahnflanken zu sehen.

2 Einführung

In diesem Kapitel werden einige grundlegende Begriffe und Vorgänge erklärt, die für das Verständnis dieser Arbeit von zentraler Bedeutung sind. Darunter fallen die in Kapitel 2.1 erklärten mathematischen Grundlagen zum Thema Splines. Kapitel 2.2 erläutert anschließend noch die Grundlagen zu Zahnrädern.

2.1 Splines

2.1.1 Berechnung der Spline-Kurve

Um den Kontaktweg Freihand zeichnen zu können, wird eine Kurvenkonstruktion benötigt, die es ermöglicht, dem Benutzer durch einfaches Setzen und Verschieben von Kontrollpunkten eine Kurve zu generieren. Dafür werden sogenannte Spline-Kurven verwendet. Folgende Definition sind angelehnt an [3].

Zuerst wird ein sogenannter Knotenvektor T mit Einträgen t_i erzeugt. Dieser hat die Form $T = [t_0, t_1, \dots, t_{n+k}]$ und muss folgende Eigenschaften erfüllen:

1. $t_i \in \mathbb{R}$ für $i = 0, \dots, n + k$
2. $t_0 \leq t_1 \leq \dots \leq t_{n+k}$
3. $t_j < t_{j+k+1}$ für $j = 0, \dots, n - 1$

Mit Hilfe dieses Knotenvektors, der als Eingangsgröße dient, kann nun eine rekursive Formel, die die sogenannte Basisfunktion der Spline-Kurve definiert, nach folgenden Schema für $t \in T$ aufgestellt werden:

Def. Spline Basisfunktion

$$N_{i,0}(t) = \begin{cases} 1, & \text{für } t_i \leq t < t_{i+1} \\ 0, & \text{sonst} \end{cases}$$
$$N_{i,k}(t) = \frac{t - t_i}{t_{i+k} - t_i} \cdot N_{i,k-1}(t) + \frac{t_{i+k+1} - t}{t_{i+k+1} - t_{i+1}} \cdot N_{i+1,k-1}(t)$$

In Abbildung 2.1 und 2.1 sind die Basisfunktionen von Grad 0 bis Grad 3 bei 5 Kontrollpunkten dargestellt.

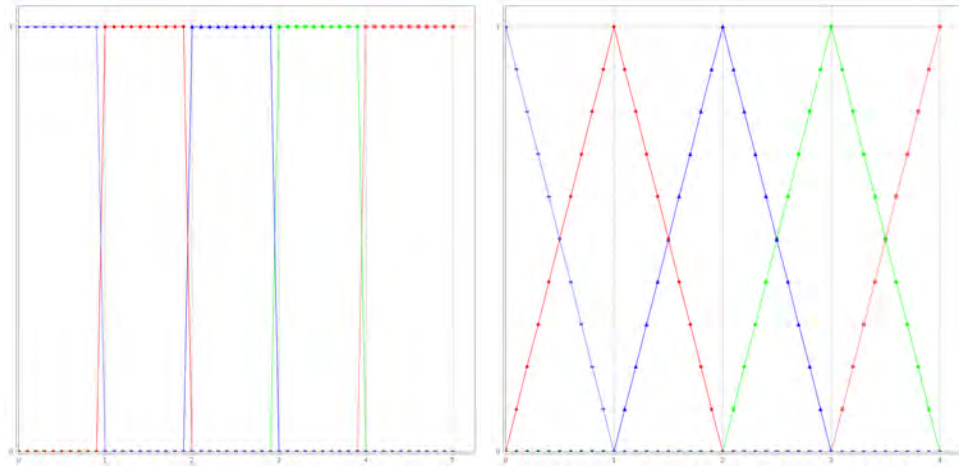


Abbildung 2.1: Spline Basisfunktionen von Grad 0 und Grad 1

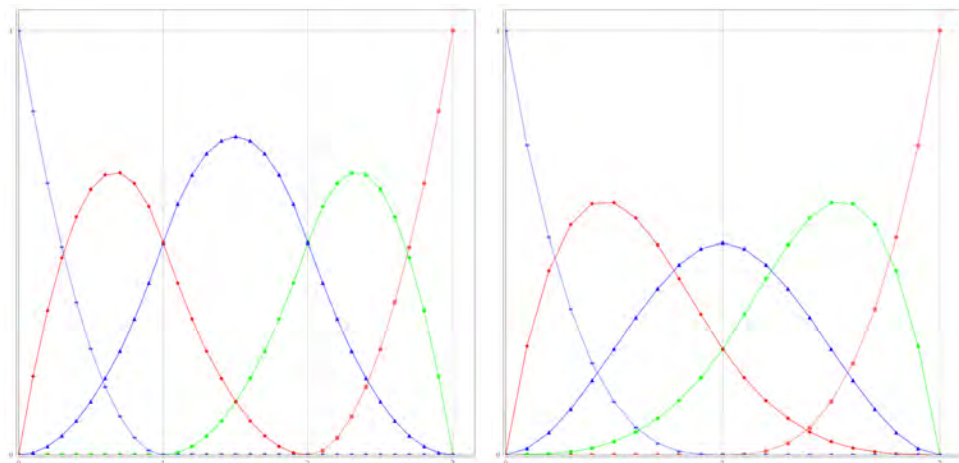


Abbildung 2.2: Spline Basisfunktionen von Grad 2 und Grad 3

Eine Spline-Kurve wird durch eine gegebene Anzahl von Kontrollpunkten festgelegt. Sie ist ein stückweises Polynom eines bestimmten Grades und wird durch die Anzahl der Kontrollpunkte, deren Koordinaten und obiger Basisfunktion samt Knotenvektor bestimmt.

Def. Spline

Seien P_0, P_1, \dots, P_{n-1} die n gegebenen Kontrollpunkte, auch de Boor Punkte¹ genannt, welche die Spline-Kurve definieren sollen. Sei weiterhin k der gewünschte Grad der Spline Kurve und $T = [t_0, t_1, \dots, t_{n+k}]$ ein Knotenvektor.

Dann lassen sich die Koordinaten eines Punktes $P(t)$, wobei t ein Eintrag aus T ist, durch eine lineare Kombination, nämlich der Funktion

$$P(t) = \sum_{i=0}^{n-1} N_{i,k}(t) \cdot P_i$$

bestimmen.

P_i ist der i -te Kontrollpunkt und ist wie alle Punkte im Rahmen dieser Arbeit von der Form (x_i, y_i) , also zweidimensional. $N_{i,k}(t)$ ist die Spline Basisfunktion, wie zuvor definiert.

2.1.2 Eigenschaften von Splines

Nennenswerte Eigenschaften von Splines und warum sie häufig zum Einsatz kommen wären:

- **Endpunktinterpolation**

Sollten im Knotenvektor T $(k + 1)$ -fache Randknoten gewählt werden, werden der linke und rechte äußere Kontrollpunkt interpoliert. Es gilt dann also:

$$P(t_0) = P_0 \text{ und } P(t_n) = P_{n-1}.$$

- **Konvexe Hülle**

Der Spline verläuft immer innerhalb der von den Kontrollpunkten definierten konvexen Hülle.

- **Nichtnegativität**

Es gilt stets $N_{i,k}(t) \geq 0$.

- **Teilung der Eins**

$$\sum_{i=0}^{n-1} N_{i,k}(\cdot|T) = 1.$$

¹Benannt nach dem deutsch-amerikanischen Mathematiker Carl-Wilhelm Reinhold de Boor, geb. 1937

Beispiel. In Abbildung 2.3 ist eine durch mein Programm generierte Spline-Kurve dargestellt. Die grünen Rechtecke sind die vom Benutzer gesetzten Kontrollpunkte und bilden zusammen mit ihren Verbindungslinien das Kontrollpolygon. Die grauen Kreise sind die errechneten Punkte auf der Kurve.

Der für dieses Beispiel verwendete Grad ist 3, wodurch die Kurve kubisch ist.

Ein exemplarisch vorgerechnetes Beispiel ist in Anhang C.1 zu finden.

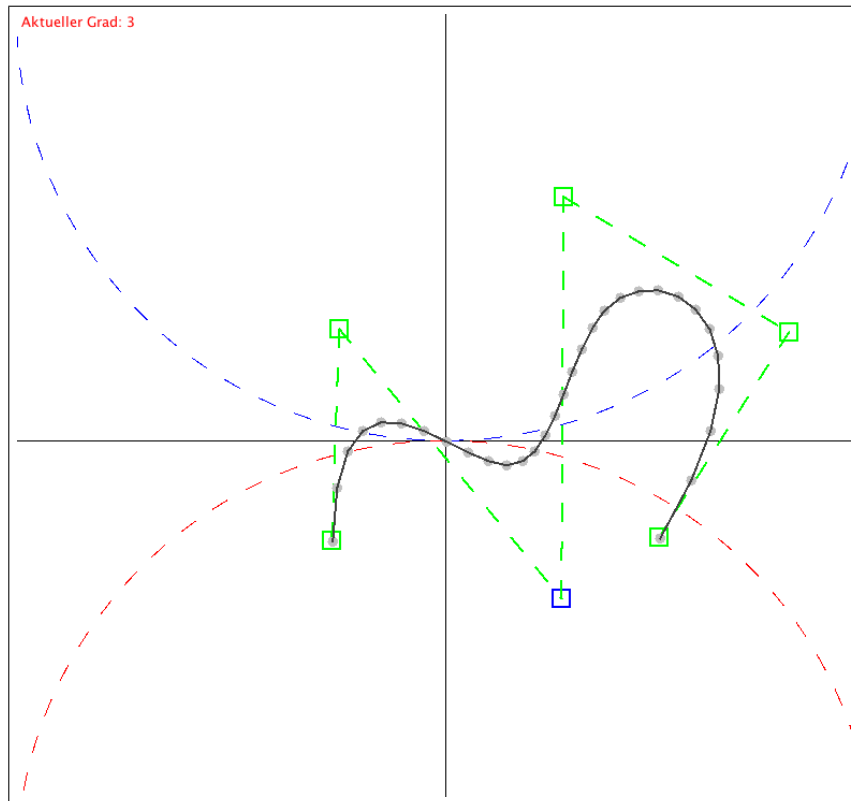


Abbildung 2.3: Grafik eines Spline

2.1.3 Ableitung der Spline-Kurve

Da für die Berechnung der Kontaktfunktion die Ableitung des Kontaktweges benötigt und dieser durch einen Spline dargestellt wird, muss die Ableitung eines Splines berechnet werden. Diese berechnet sich aus der Multiplikation der Kontrollknoten mit der abgeleiteten Vandermonde Matrix.

Sei hierfür $C \in \mathbb{R}^{2 \times n}$ die Matrix mit den x- und y-Koordinaten der Kontrollknoten. Die Ableitung der Spline-Kurve berechnet sich durch die Matrizenmultiplikation $CN_k^{(1)}(\cdot|T)$.

Def. Die Matrix $N_k^{(1)}(X|T) \in \mathbb{R}^{n \times \#X}$ ist die erste Ableitung der Matrix, welche als Einträge die Werte der entsprechenden Spline Basisfunktion enthält, und berechnet sich durch

$$N_k^{(1)}(X|T) = G_1 N_{k-1}(X|T_1)$$

Für diese Berechnung werden die beiden Matrizen G_1 und $N_{k-1}(X|T_1)$ benötigt. Diese werden nachfolgend definiert.

Die Matrix G_1 berechnet sich durch

$$G_1 = kD_n \Delta_k T_1$$

mit:

- k ist der Grad des Splines.
- D_n ist die Differenzenmatrix und bestimmt durch:

$$D_n = \begin{bmatrix} -1 & . & . & . \\ 1 & -1 & . & . \\ . & 1 & . & . \\ . & . & . & -1 \\ . & . & . & 1 \end{bmatrix} \in \mathbb{R}^{n \times n-1}.$$

- $\Delta_k T_1 = \text{diag}[(t_{j+k} - t_j)^{-1} : j = 1, \dots, n - 1] \in \mathbb{R}^{n-1 \times n-1}$.

Die Matrix $N_{k-1}(X|T_1)$ besitzt als Einträge den jeweiligen Wert der Spline Basisfunktion an der entsprechenden Position und wird wie folgt definiert:

$$N_{k-1}(X|T_1) = [N_{i,k-1}(x|T_1) : i = 0, \dots, n - 2, x \in X] \in \mathbb{R}^{n-1 \times \#X}$$

Da nun G_1 und $N_{k-1}(X|T_1)$ bekannt sind, kann die Ableitungsmatrix der ersten Ableitung $N_k^{(1)}(X|T)$ wie oben definiert berechnet werden und die Werte der Spline-Kurve mit Multiplikation der Kontrollknoten $CN_k^{(1)}(X|T)$ bestimmt werden.

Ein Beispiel zur Berechnung der Ableitung einer Spline-Kurve ist in Anhang C.2 zu finden.

2.2 Zahnräder

Ein Zahnrad ist ein Maschinenelement, das zur Übertragung zweier sich entgegengesetzter Drehungen aufeinander verwendet wird. Der Kontakt zwischen zwei Zahnrädern findet durch die aus dem Grundkörper hervorstehenden Zähne statt.

2.2.1 Stirnräder

In dieser Arbeit werden die Zahnräder allgemein auf Stirnräder speziell eingeschränkt. Der Grundkörper eines Stirnrades ist ein Zylinder, aus dem die Zähne herausragen. Bei einer Stirnradverzahnung sind im Gegensatz zu einer Kegelradverzahnung die Drehachsen beider Stirnräder parallel. Man spricht von einem Stirnradpaar.

Bei Stirnrädern kann eine Schräg- oder Geradverzahnung existieren. In dieser Arbeit wird nur der Fall einer Geradverzahnung betrachtet. Dafür wird ein Zahn in der Ebene konstruiert. Um nun einen dreidimensionalen Zahn für ein geradverzahntes Zahnrad zu erhalten, wird dieses Konstrukt n-mal gerade nach hinten versetzt.

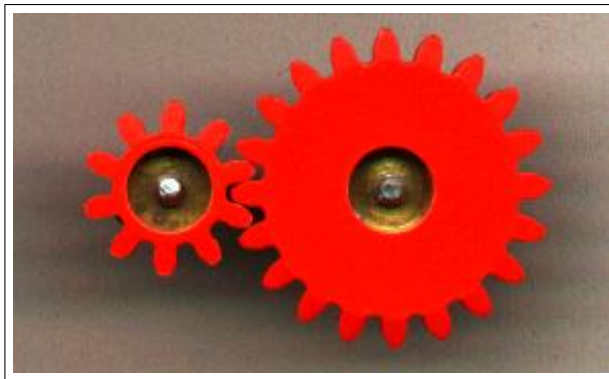


Abbildung 2.4: Ein Stirnradpaar

2.2.2 Historisches

Die im alten Ägypten erfundenen Göpel, die aus einem großen Holzrad mit Pflöcken bestanden, sind die geschichtlich älteste Form von Zahnrädern.[2] In Europa fanden die ersten Zahnräder im 9. Jahrhundert in Wind- und Wassermühlen ihren Einsatz.

Im 16. Jahrhundert wurden die Zahnräder erstmals aus Eisen geschmiedet, woraus allerdings eine wenig gleichmäßige Form der Zähne resultiert. Philippe de La Hire² war einer der Ersten, die eine genauere mathematische Untersuchung der Zahnräder durchführte. Die daraus resultierende verbesserte Form der Zähne sicherte den Einsatz der Zahnräder in Uhrwerken.

²Philippe de La Hire, französischer Mathematiker, 1640-1718

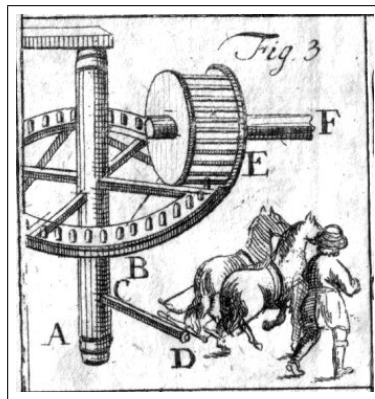


Abbildung 2.5: Skizze einer Göpelmühle

Zur Zeit der Industrialisierung (18. Jahrhundert) wurden die ersten Massen an Zahnrädern aus Metall gefertigt. Dies liegt an der Erfindung der Dampfmaschine von Thomas Newcomen³ 1712, für die sehr viele Zahnräder benötigt wurden.[1]

Die Form der Zähne und die Art der Verzahnung veränderten sich im Laufe der Geschichte.

So wurden aus ungenau geschmiedeten Zahnrädern immer präziser gefertigte Maschinenelemente. 1760 entwickelte Euler⁴ die Evolvente als Verzahnung und 1820 Joseph Woollams die Schrägverzahnung.

Heutzutage werden in der Zahnräderfertigung Verfahren angewandt, wie Gießen, Stanzen, 5-Achs-Fräsen, Wälzfräsen und viele mehr.



Abbildung 2.6: Wälzfräsen

³Thomas Newcomen, englischer Erfinder, 1663-1729

⁴Leonhard Euler, schweizer Mathematiker, 1707-1783

2.2.3 Begrifflichkeiten

Dieses Kapitel gibt einen kurzen Überblick über elementare Begriffe zu Zahnrädern. Sonstige Begriffe werden, falls benötigt, noch in den einzelnen Kapiteln genauer erklärt.

- **Achsabstand**

Dies ist der Abstand der Mittelpunkte beider Zahnräder. Er wird nachfolgend mit d bezeichnet.

- **Eingriffswinkel**

Bei einer normalen Evolventenverzahnung bezeichnet der Eingriffswinkel den Winkel zwischen der Horizontalen und der Evolvente.

- **Kontaktweg**

Der Kontaktweg eines Stirnradpaares ist die Kurve auf der beide Zahnräder überwiegend abwälzen. Dieser wird häufig als Evolvente, Zykloide oder auch Zykloevolvente gewählt, ist in dieser Arbeit jedoch frei formbar. Er wird nachfolgend mit k bezeichnet.

- **Modul**

Das Modul eines Zahnrades ist definiert durch $\frac{\text{Teilkreisdurchmesser}}{\text{Zahnanzahl}}$. Es wird nachfolgend mit mod bezeichnet.

- **Profillinie**

Die Profillinie gibt die Form eines Zahnes an. Ein Zahn besteht aus 2 Profillinien und einem Zahnkopf im 2-dimensionalen, welche für das Modell im 3-dimensionalen nach hinten versetzt werden.

- **Radius**

Die Angabe eines Radius ist bei der Zykloidverzahnung wichtig. Im Allgemeinen ist damit der Radius beider Zykloiden gemeint.

- **Übersetzungsverhältnis**

Das Übersetzungsverhältnis zweier Zahnräder berechnet sich durch

$$u = \frac{\text{Zahnanzahl}_{\text{Rad}}}{\text{Zahnanzahl}_{\text{Ritzel}}}$$

- **Wälzpunkt**

Der Wälzpunkt eines Stirnradpaares ist der Punkt auf dem beide Zahnräder hauptsächlich aufeinander abwälzen. Er wird nachfolgend mit c bezeichnet.

3 Vom Spline zum fertigen Zahnrad

Da nun die grundlegenden Begrifflichkeiten geklärt sind, können wir den Hauptteil dieser Arbeit betrachten. Es wird erklärt, wie aus einem Spline, den man erstellt hat, ein fertiges Zahnrad wird. Dabei werden in Kapitel 3.1 Vorgänge dargelegt, die man für die Umsetzung der Splines in diesem Fall beachten muss. Kapitel 3.2 und 3.3 erläutern anschließend die Berechnung beider Profillinien. Diese sind angelehnt an [4].

Für die graphische Darstellung der Zahnräder im Programm wird zusätzlich noch ein Zahnfuß und -kopf konstruiert. Dieses Vorgehen erklärt Kapitel 3.4 und in Kapitel 3.5 wird das Programm, welches die graphische Oberfläche für den Benutzer darstellt genauer gezeigt.

3.1 Umsetzung der Splines

Wie setzt man Splines um? Dafür wird zuerst einmal eine gültige Wahl eines Knotenvektors benötigt. Kapitel 3.1.1 erläutert meine Wahl und den Beweis dafür, dass diese korrekt ist. Zudem muss für eine korrekte Berechnung der Profillinien gewährleistet werden, dass der Kontaktweg durch den Wälzpunkt verläuft. Wie diese Fixierung eines Punktes vonstattengeht, erklärt Kapitel 3.1.2.

3.1.1 Knotenvektor

Für die Berechnung von Spline-Kurven wird, wie im Einführungskapitel schon erklärt, ein Knotenvektor benötigt. Dieser dient lediglich als Eingangsgröße, denn es gibt keine vorgeschriebene Form, solange keine der Anforderungen verletzt wird.

In dieser Arbeit konstruiere ich den Knotenvektor folgendermaßen, wobei k der Grad und n die Anzahl an Kontrollpunkten ist:

$$t_i = \begin{cases} 0, & \text{für } 0 \leq i \leq k \\ i - k, & \text{für } k < i < n \\ n - k, & \text{für } n \leq i \leq n + k \end{cases}$$

Beispiel. Bei 5 Kontrollknoten und einem Grad 2, wäre $T = [0, 0, 0, 1, 2, 3, 3, 3]$

Lemma. Die obige Konstruktion des Knotenvektors erfüllt die Voraussetzungen und ist somit eine gültige Wahl dafür.

Beweis. Folgendes ist zu zeigen:

1. Anzahl an Elementen
2. Zugehörigkeit der Elemente zu \mathbb{R}
3. $t_0 \leq t_1 \leq \dots \leq t_{n+k}$
4. $t_j < t_{j+k+1}$ für $j = 0, \dots, n-1$

Zu 1: Da $0 \leq i \leq n+k$ ist die Anzahl an Elementen $n+k+1$.

Zu 2: Da n und k beide aus \mathbb{N}_0 und i nur ganzzahlige Werte zwischen 0 und $n+k$ annimmt, ist die Anforderung erfüllt, da somit $0 \in \mathbb{R}$, $i-k \in \mathbb{R}$, $n-k \in \mathbb{R}$.

Zu 3: Man betrachte zuerst die drei Zweige der Abbildungsvorschrift separat.

Für den ersten gilt: $t_i = t_{i+1} = 0$ und $0 \leq 0$.

Für den zweiten gilt: $t_i = i-k$ und $t_{i+1} = i+1-k$. Es gilt: $i-k \leq i+1-k \Leftrightarrow i \leq i+1$. Dies ist erfüllt, da i ein fortlaufender Index ist.

Für den dritten Zweig gilt: $t_i = t_{i+1} = n-k$ und $n-k \leq n-k \Leftrightarrow 0 \leq 0$.

Nun bleiben noch die beiden Übergangsstellen zu betrachten.

Die erste ist nach Definition von t_k auf t_{k+1} . Es gilt: $t_k \leq t_{k+1} \Leftrightarrow 0 \leq k+1-k \Leftrightarrow 0 \leq 1$.

Die zweite Übergangsstelle ist von t_{n-1} auf t_n . Es gilt: $t_{n-1} \leq t_n \Leftrightarrow n-1-k \leq n-k \Leftrightarrow n-1 \leq n$.

Somit erfüllt diese Konstruktion von T die Anforderung $t_0 \leq t_1 \leq \dots \leq t_{n+k}$.

Zu 4: Betrachte zuerst $j = 0, \dots, k$. Somit sind alle $t_j = 0$ nach Definition. Da $t_{k+1} = 1$ wie oben gezeigt und für die Konstruktion $t_0 \leq t_1 \leq \dots \leq t_{n+k}$ gilt, ist für diese j die Anforderung erfüllt.

Betrachte nun $j = k+1, \dots, n-1$. Diese t_j fallen alle in den zweiten Zweig der Konstruktion, es gilt also: $t_j = j-k$.

Angenommen t_{j+k+1} liegt auch im zweiten Zweig, ist $t_{j+k+1} = j+k+1-k = j+1$. Da $k > 0$ ist $j-k < j+1$.

Angenommen t_{j+k+1} liegt im dritten Zweig, ist $t_{j+k+1} = n-k$. $t_j < t_{j+k+1} \Leftrightarrow j-k < n-k \Leftrightarrow j < n$. Dies gilt, da $j = 0, \dots, n-1$.

Somit ist die Anforderung $t_j < t_{j+k+1}$ für $j = 0, \dots, n$ durch diese Konstruktion von T erfüllt. T nach obiger Konstruktionsvorschrift gebildet erfüllt also alle 4 Anforderungen und ist somit eine gültige Wahl für den Knotenvektor. \square

3.1.2 Fixierung eines Punktes auf der Spline-Kurve

An den Kontaktweg eines Zahnrades wird die Anforderung gestellt, einen Punkt in der Mitte zwischen beiden Wälzkreisen auf dem Wälzpunkt c zu besitzen, um korrekte Berechnungen durchführen zu können. Deswegen muss ein Punkt der Spline-Kurve fixiert werden. Welchen Punkt des Spline man fixiert, ist im Prinzip irrelevant. Damit aber die vor der Fixierung gezeichnete Form grundlegend erhalten bleibt, sollte der Punkt gewählt werden, der von Anfang an am nächsten zum Wälzpunkt liegt.

Um den Punkt zu ermitteln, der fixiert wird, wird sobald die Anzahl an Kontrollknoten um eins größer ist als der Grad des Spline, der Spline vorerst normal ohne Fixierung vorberechnet. Danach wird der Punkt auf der Spline-Kurve, dessen Abstand zum Wälzpunkt am kleinsten ist, also denjenigen, für dessen x- und y-Koordinate

$$\sqrt{(x_{Knoten} - x_{Mittelpunkt})^2 + (y_{Knoten} - y_{Mittelpunkt})^2}$$

minimal ist und das t_x aus T aus dem er sich berechnet ermittelt.

Um diesen Punkt zentrieren zu können, muss eine Verschiebung eines Kontrollknoten stattfinden. Dafür ist im Programm ein vor-eingefügter Kontrollknoten im Mittelpunkt, der speziell für diesen Zweck vorgesehen ist und vom Benutzer selbst nicht verschoben werden kann.

Sei dieser Kontrollpunkt P_j und sei $P_{Mittelpunkt}$ ein Punkt, dessen Koordinaten dem des Wälzpunktes entsprechen. Dann lässt sich P_j durch eine Umstellung der allgemeinen Formel berechnen:

$$P_j = \frac{P_{Mittelpunkt} - \sum_{i \neq j} N_{i,k}(t_x) \cdot P_i}{N_{j,k}(t_x)}$$

Beispiel. In Abbildung 3.1 sieht man einmal eine Spline-Kurve ohne und in Abbildung 3.2 mit einem fixierten Punkt im Zentrum. Der blaue rechteckige Punkt ist der vom Benutzer nicht verschiebbare Kontrollpunkt, durch dessen errechnete Position die Fixierung stattfindet.

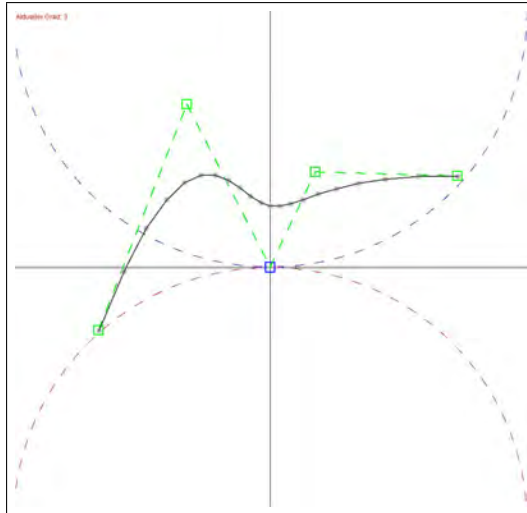


Abbildung 3.1: Spline ohne Fixierung

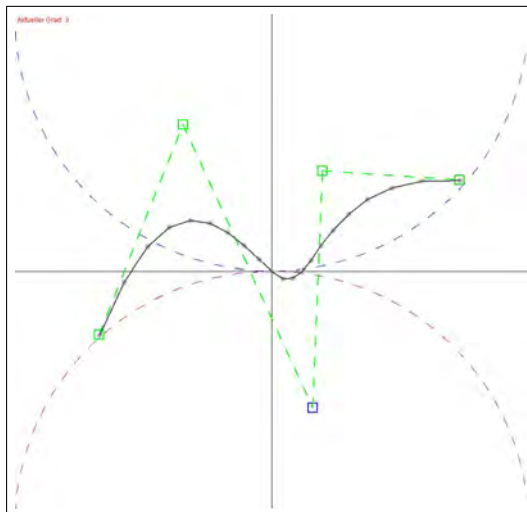


Abbildung 3.2: Spline mit Fixierung

3.2 Berechnungen mit vorgegebener Übersetzung und Kontaktweg

Im ersten Fall der Berechnungen ist der Kontaktweg k durch Erstellung eines Splines gegeben. Der Kontaktweg liegt somit diskret als zweidimensionale Punkte in \mathbb{R}^2 in Abhängigkeit eines Parameters t aus dem Knotenvektor T vor.

Zudem ist k' , die Ableitung des Kontaktweges, bekannt. Beide wurden durch das Setzen von Kontrollpunkten errechnet und sind durch die Berechnung Matrizen in $\mathbb{R}^{2 \times \#X}$. $\#X$ ist die Anzahl von Werten, an denen der Spline ausgewertet wurde.

Man möchte die beiden Zahnflanken f und g berechnen, die das Zahnrad definieren.

Dazu extrahiert man zuerst einmal eine Spalte, die zu einem $t \in T$ gehört, aus dem sie berechnet wurde. Es gilt also für den Kontaktweg:

$$k : t \in T \mapsto k(t) = \begin{pmatrix} k_x(t) \\ k_y(t) \end{pmatrix}.$$

Und für die Ableitung:

$$k' : t \in T \mapsto k'(t) = \frac{d}{dt}k(t) = \begin{pmatrix} k'_x(t) \\ k'_y(t) \end{pmatrix}.$$

Des Weiteren lassen sich im Programm das Modul mod , die Zähnezahzahl des Ritzel n und das Übersetzungsverhältnis u eingeben.

Mit Hilfe dieser Eingaben lässt sich der Achsabstand d beider Zahnräder berechnen. Da mod definiert ist, als

$$\frac{\text{Teilkreisdurchmesser}}{\text{Zahnanzahl}}$$

erhält man den Teilkreisdurchmesser des Ritzel durch

$$d_1 = mod \cdot n.$$

Die Zähnezahzahl des Rad ist $u \cdot n$ und dessen Teilkreisdurchmesser somit

$$d_2 = mod \cdot u \cdot n.$$

Folglich ist der Achsabstand die Summe beider Teilkreisdurchmesser, es gilt also für d :

$$d = \frac{d_1 + d_2}{2}.$$

Nun muss der Wälzpunkt c bestimmt werden, an dem beide Zahnräder überwiegend aufeinander abwälzen. Das Zahnrad wird so konstruiert, dass es auf der x-Achse zentriert liegt. Für den x-Wert gilt demnach $c_x = 0$.

3 Vom Spline zum fertigen Zahnrad - 3.2 Berechnungen mit vorgegebener Übersetzung und Kontaktweg

Da der Mittelpunkt des Ritzel im Nullpunkt liegt, ist der y-Wert des Wälzpunktes beim Wert des halben Teilkreisdurchmessers des Ritzel, also bei $c_y = \frac{d}{u+1}$. Für die Koordinaten von c gilt also:

$$c = \begin{pmatrix} c_x \\ c_y \end{pmatrix} = \begin{pmatrix} 0 \\ \frac{d}{u+1} \end{pmatrix}.$$

Für die Errechnung beider Profillinien wird die Kontaktfunktion $T_k(t)$ benötigt. Dafür wird zuerst die Ableitung $T'_k(t)$ gebildet und diese anschließend integriert. Es gilt:

$$T'_k(t) = \frac{c_y k'_y(t) - k_x(t) k'_x(t) - k_y(t) k'_y(t)}{c_y k_x(t)} = \frac{k'_y(t) - c_y^{-1} k^T(t) k'(t)}{k_x(t)}.$$

Nun erhält man durch Integration $T_k(t)$

$$T_k(t) = \int_{t_0}^t T'_k(\sigma) d\sigma = \int_{t_0}^t \frac{k'_y(\sigma) - c_y^{-1} dk^T(\sigma) k'(\sigma)}{k_x(\sigma)} d\sigma,$$

wobei $t_0 \in T$.

Da eine Anforderung an den Kontaktweg ist, durch den Wälzpunkt zu laufen und dieser wie oben definiert ist, erkennt man, dass bei diesem Punkt eine Singularität in der Ableitung der Kontaktfunktion auftritt, da der Nenner 0 wird.

Im Programm wird diese Singularität umgangen, indem dieser Wert durch das arithmetische Mittel des vorangegangenen und nachfolgenden Wertes approximiert wird.

Die beiden Profillinien f und g lassen sich jetzt mit Hilfe der Kontaktfunktion berechnen. Es gilt:

$$f(T_k(t)) = \begin{pmatrix} k_x \cos(T_k(t)) + k_y \sin(T_k(t)) \\ -k_x \sin(T_k(t)) + k_y \cos(T_k(t)) \end{pmatrix}$$

$$g(T_k(t)) = \begin{pmatrix} k_x \cos(T_k(t)) + (k_y - d) \sin(T_k(t)) \\ -k_x \sin(T_k(t)) + (k_y - d) \cos(T_k(t)) \end{pmatrix}.$$

3.3 Berechnungen mit vorgegebener Übersetzung und Profillinie

Dieser Fall findet im Programm keine Anwendung, da es nicht gebräuchlich ist, aus einer Profillinie die andere zu errechnen, sondern der Fall gegeben ist, wie in Kapitel 3.2. Trotzdem werden die Formeln und das Vorgehen zu dieser Art von Berechnung zum Zwecke der Vollständigkeit nachfolgend kurz erwähnt. Ein detailliertes Vorgehen ist nachzulesen in [4].

Im zweiten Fall der Berechnung ist die Übersetzung und eine Profillinie durch einen Spline gegeben. Die Profillinie liegt somit diskret als zweidimensionale Punkte in \mathbb{R}^2 in Abhängigkeit des Knotenvektors T vor. Weiterhin ist die Ableitung der Profillinie bekannt. Beide werden wieder durch das Setzen von Kontrollpunkten und die daraus resultierenden Berechnungen definiert.

Man möchte nun damit zuerst den Kontaktweg k und anschließend die andere Profillinie bestimmen.

Sei in Fall 1 diese vorgegebene Profillinie f . Dann kann wieder eine Spalte aus der Matrix zu einem gewissen $t \in T$ extrahiert werden. Für f gilt also:

$$f : t \in T \mapsto f(t) = \begin{pmatrix} f_x(t) \\ f_y(t) \end{pmatrix} = f_r(t) \begin{pmatrix} \cos(f_\varphi(t)) \\ \sin(f_\varphi(t)) \end{pmatrix}.$$

Und für deren Ableitung:

$$f' : t \in T \mapsto f'(t) = \frac{d}{dt}f(t) = \begin{pmatrix} f'_x(t) \\ f'_y(t) \end{pmatrix} = f'_r(t) \begin{pmatrix} \cos(f'_\varphi(t)) \\ \sin(f'_\varphi(t)) \end{pmatrix}.$$

Auch in diesem Fall lässt sich der Wälzpunkt c mit Hilfe der Werte von mod , n und u analog zu Kapitel 3.2 berechnen, es gilt also:

$$c = \begin{pmatrix} 0 \\ \frac{d}{u+1} \end{pmatrix}.$$

Nun lässt sich die Kontaktfunktion $T_k(t)$ berechnen durch:

$$T_k(t) = \pm \arcsin(c_y^{-1} f_r(t) \cos(f_\varphi(t) - f'_\varphi(t))) - f'_\varphi(t).$$

Mit Hilfe der Kontaktfunktion kann nun der Kontaktweg k und daraus resultierend die andere Profillinie g berechnet werden.

$$k(T_k(t)) = \begin{pmatrix} f_x \cos(T_k(t)) - f_y \sin(T_k(t)) \\ f_x \sin(T_k(t)) + f_y \cos(T_k(t)) \end{pmatrix}.$$

$$g(T_k(t)) = f_r \begin{pmatrix} \cos(f_\varphi T_k(t)) \\ \sin(f_\varphi T_k(t)) \end{pmatrix} - d \begin{pmatrix} \sin(T_k(t)) \\ \cos(T_k(t)) \end{pmatrix}.$$

Sei nun in Fall 2 nicht f , sondern g und dessen Ableitung durch den Spline vorgegeben. Es gilt also analog für g :

$$g : t \in T \mapsto g(t) = \begin{pmatrix} g_x(t) \\ g_y(t) \end{pmatrix} = g_r(t) \begin{pmatrix} \cos(g_\varphi(t)) \\ \sin(g_\varphi(t)) \end{pmatrix}.$$

Und für die Ableitung:

$$g' : t \in T \mapsto g'(t) = \frac{d}{dt}g(t) = \begin{pmatrix} g'_x(t) \\ g'_y(t) \end{pmatrix} = g'_r(t) \begin{pmatrix} \cos(g'_\varphi(t)) \\ \sin(g'_\varphi(t)) \end{pmatrix}.$$

Der Wälzpunkt c wird wie vorher berechnet. Für die Kontaktfunktion gilt:

$$T_k(t) = \pm \arcsin \left(\frac{c_y^{-1}}{-u} g_r(t) \cos(g_\varphi(t) - g'_\varphi(t)) \right) - g'_\varphi.$$

Das weitere Vorgehen zur Bestimmung des Kontaktweges k und der Berechnung der anderen Profillinie f ist nun analog zum vorherigen Vorgehen, als f gegeben war.

3.4 Für die Graphik - Konstruktion von Zahnfuß und -kopf

Im Programm werden die berechneten Profillinien f und g am Ende so dargestellt, dass die Zahnräder ersichtlich sind. Damit diese Anzeige klarer wird und nicht nur aus den beiden Flanken besteht, werden sowohl eine Ellipse als Zahnfuß, als auch als Zahnkopf automatisch konstruiert und eingefügt.

Diese dienen nur der Optik und hängen in ihrer Berechnung nicht von etwaigen Eingaben ab. So ist der in Kapitel 3.4.1 konstruierte Zahnfuß im Programm immer viertel so tief wie breit, unabhängig von der Größe des Zahnrades. Der in Kapitel 3.4.2 ist ein Kreissegment, der die beiden äußeren Punkte von f und g miteinander verbindet.

3.4.1 Zahnfuß

Damit beide Profillinien verbunden sind, wird im Zahnfußbereich eine Ellipse als Konstruktion verwendet.

Um diesen Zahnfuß mit diskreten ellipsenhafte angeordneten Punkten zu konstruieren, benötigt man die beiden Fußpunkte. Seien diese f_1 als Fußpunkt der ersten Profillinie und f_2 als Fußpunkt der zweiten Profillinie. Außerdem sei n_E die gewünschte Anzahl an diskreten Punkten. Ein größerer Wert von n_E sorgt für eine stärkere Glätte der Ellipse.

Zuerst wird die Hälfte der Ellipse in der passenden Größe um den Nullpunkt konstruiert. Da beide Flanken ihre Fußpunkte jedoch nicht auf der x-Achse haben, muss diese anschließend noch verschoben und gedreht werden.

Als Erstes zur Konstruktion der halben Ellipse:

Zuerst wird der Abstand beider Fußpunkte benötigt. Dieser errechnet sich aus

$$d_F = \sqrt{(f_{1,x} - f_{2,x})^2 + (f_{1,y} - f_{2,y})^2}.$$

Die halbe Ellipse soll achsensymmetrisch zur y-Achse, mit beiden Endpunkte auf der x-Achse und nach oben offen liegen. Dies ist dargestellt in Abbildung 3.3.

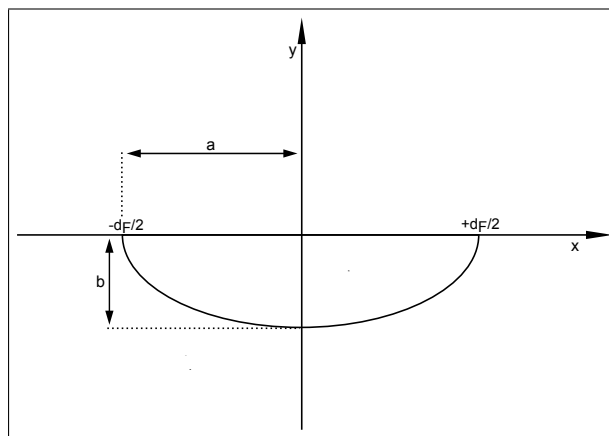


Abbildung 3.3: Ellipse für Konstruktion

Nun wird um den Nullpunkt ein gleichverteilter Vektor erstellt, jeweils von $-\frac{d_F}{2}$ bis $+\frac{d_F}{2}$ mit n_E Elementen.

Zur Berechnung der y-Werte mit Hilfe der allgemeinen Ellipsenformel, werden noch die beiden "Radien" a und b benötigt.

Hierbei a ist einfach der halbe Fußpunktabstand, also $a = \frac{d_F}{2}$ und b wird durch $b = a \cdot p$ angegeben, wobei p das gewünschte Verhältnis Höhe zu Breite ist. Innerhalb des Programms ist $p = 0.5$.

Da nun die x-Koordinaten der Ellipse feststehen, berechnen sich die y-Koordinaten durch Umstellung der allgemeinen Formel der Ellipse, nämlich

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1 \Leftrightarrow y = \pm \sqrt{b^2 \cdot \left(1 - \frac{x^2}{a^2}\right)}.$$

Da die Hälfte der Ellipse nach unten konvex sein muss, gilt:

$$y = -\sqrt{b^2 \cdot \left(1 - \frac{x^2}{a^2}\right)}.$$

Die halbe Ellipse liegt nun korrekt dimensioniert um den Mittelpunkt vor und muss nur noch gedreht und an einen der beiden Zahnfußpunkte verschoben werden.

Zur Drehung und Verschiebung der halben Ellipse:

Um die Ellipse zu drehen, werden ihre Koordinaten, die als Matrix vorliegen, mit der Drehmatrix

$$M_{dreh} = \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{pmatrix}$$

multipliziert.

Für die richtige Drehung sei $\alpha = \tan\left(\frac{|f_{1,y} - f_{2,y}|}{|f_{1,x} - f_{2,x}|}\right)$.

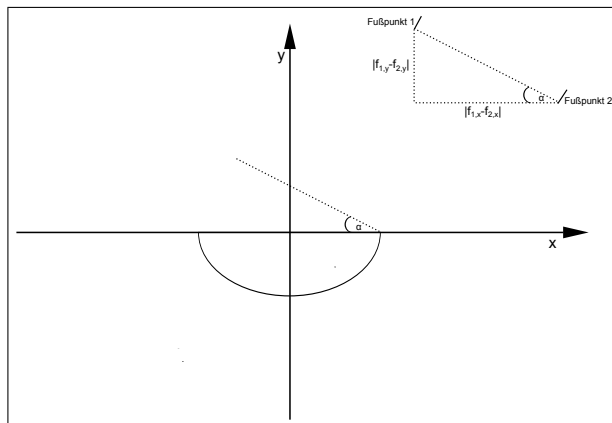


Abbildung 3.4: Drehung der Ellipse

Mit dieser Matrix multipliziert ist der Ellipsenabschnitt richtig gedreht und muss nur noch an den Fußpunkt verschoben werden, damit er korrekt als Zahnfuß platziert ist.

Dies erfolgt durch lineare Transformationen der einzelnen x- und y-Koordinaten.

3.4.2 Zahnkopf

Anders als beim Zahnfuß, wird beim Zahnkopf keine Ellipse beliebiger Tiefe, sondern ein Kreissegment konstruiert und zwischen die beiden Profillinien eingepasst, welches als Radius den halben Kopfkreisdurchmesser des Zahnrades hat. Hierfür werden Polarkoordinaten verwendet. Der Zahnkopf wird sofort passend an die richtige Stelle konstruiert und muss nicht wie der Zahnfuß noch verschoben werden.

Seien dafür zk_1, zk_2 die beiden Flankenspitzen eines Zahns. Da das Kreissegment auch hier aus diskreten Punkten besteht und die Zahnräder relativ klein werden können, muss ein sinnvoller Abstand der Punkte gewählt werden. In dieser Arbeit wird deshalb festgelegt, dass alle 0.1 Grad ein Punkt gesetzt werden soll.

Da wir uns hier im Bogenmaß befinden, rechnen wir die 0.1 Grad in Abhängigkeit von π um. Es gilt:

$$2\pi = 360^\circ \Rightarrow \frac{\pi}{1800} = 0.1^\circ.$$

Sei also $fac = \frac{\pi}{1800}$.

Weiterhin seien r die Entfernung vom Nullpunkt bis zur Zahnspitze von zk_1 . r lässt sich leicht bestimmen, da dies einfach der halbe Durchmesser des Kopfkreises des Zahnrades ist.

Dieser Radius ist der selbe wie der, des zu konstruierenden Kreissegments. Die einzelnen Größen sind dargestellt in Abbildung 3.5.

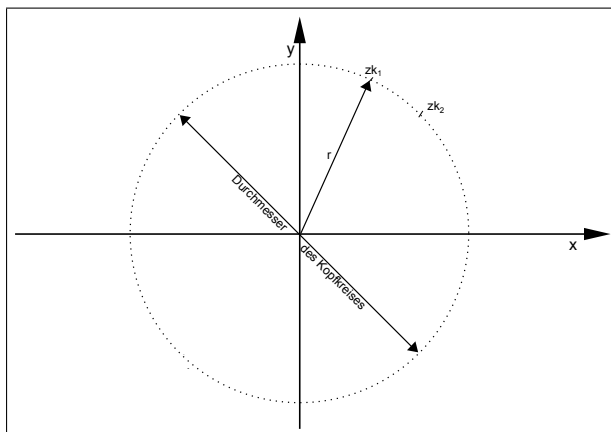


Abbildung 3.5: Schematische Darstellung

Im Falle von Polarkoordinaten werden die Punkte auf einem Kreis berechnet mit Hilfe der Formeln

3 Vom Spline zum fertigen Zahnrad - 3.4 Für die Graphik - Konstruktion von Zahnfuß und -kopf

$$x_K = x_M + r \cdot \sin(\varphi),$$

$$y_K = y_M + r \cdot \cos(\varphi),$$

wobei x_K und y_K die Koordinaten auf dem Kreis, x_M und y_M die Koordinaten des Kreismittelpunktes, r der Radius des Kreises und φ ein Winkel im Bogenmaß zwischen 0 und 2π ist.

Zur Berechnung benötigt man außerdem die beiden Winkel φ_S und φ_E , die als Winkel zwischen zwei Vektoren mit Hilfe des Skalarproduktes definiert sind als:

$$\varphi_S = \arccos \left(\frac{\begin{pmatrix} zk_{2,x} \\ zk_{2,y} \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 0 \end{pmatrix}}{\left| \begin{pmatrix} zk_{2,x} \\ zk_{2,y} \end{pmatrix} \right| \cdot \left| \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right|} \right)$$

$$\varphi_E = \arccos \left(\frac{\begin{pmatrix} zk_{1,x} \\ zk_{1,y} \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 0 \end{pmatrix}}{\left| \begin{pmatrix} zk_{1,x} \\ zk_{1,y} \end{pmatrix} \right| \cdot \left| \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right|} \right)$$

Beide Winkel sind in schematisch zu sehen in Abbildung 3.6.

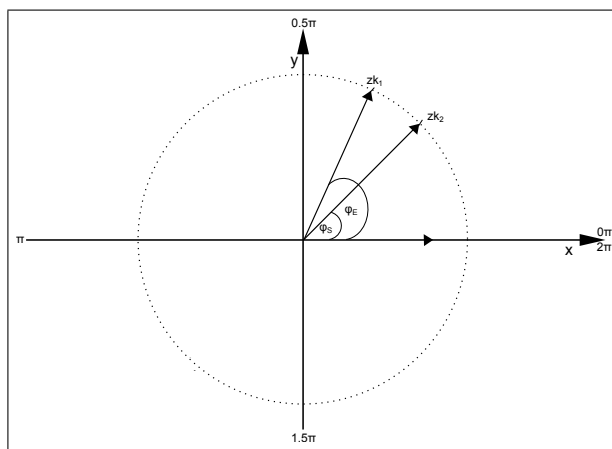


Abbildung 3.6: Schematische Darstellung von φ_S und φ_E

Nun berechnen sich die x- und y-Koordinaten des Kreissegments zwischen zk_1 und zk_2 durch:

$$x_k = r \cdot \sin(\varphi),$$

$$y_k = r \cdot \cos(\varphi),$$

wobei $\varphi \in [\varphi_S, \varphi_E]$.

Im Programm nimmt φ diskrete im Abstand fac an.

3.5 Das Programm

Dieser Abschnitt gibt Einblick in das Programm, welches es dem Benutzer ermöglicht, die Zahnräder zu erstellen. In Kapitel 3.4.1 werden kurz die Daten zur Entwicklung des Programms dargelegt. Kapitel 3.4.2 gibt Überblick über die verschiedenen Elemente im Programm und deren Funktionen.

3.5.1 Entwicklungsumgebung

Unten aufgeführt sind für die Entwicklung des Programms relevante Daten.

- Entwicklungsumgebung: Eclipse Luna von Oracle
- Java Compiler: Compliance Level 1.6
- Programm entwickelt auf einem MacBook Air unter OS X 10.9.4 mit Java Runtime Environment 1.7.0_67

3.5.2 Aussehen und Funktion des GUI

Beim Starten des Programms erscheint sofort das Hauptfenster, welches durch Tabs in die verschiedenen Bereiche gegliedert ist. Diese werden nachfolgend genauer dargestellt.

Die Tabbar

Abbildung 3.7 zeigt die übergeordnete Tabbar des Programms. Es gibt jeweils einen Tab für die Eingaben, die zu tätigen sind, das Zeichnen des Kontaktweges, die Anzeige für den Plot des Kontaktweges mit Profillinien und Wälzkreisen und die Darstellung der Zahnräder. Durch Auswahl der einzelnen Tabs ist es dem Benutzer möglich zu den eben genannten Funktionen zu springen.



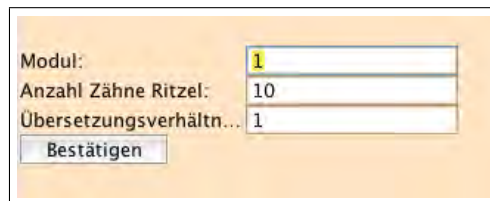
Abbildung 3.7: Tabbar des Programms

Das Eingabefenster

In Abbildung 3.8 sieht man den ersten Tab. Dies dient dazu, die Eingaben zu tätigen, die benötigt werden, um den Wälzpunkt zu berechnen, also Modul, Anzahl der Zähne des Ritzel und das Übersetzungsverhältnis. Sind diese Eingaben getätigt und bestätigt, wird c berechnet. Dieses Fenster muss als erstes befüllt werden, da sonst mit dem später gezeichneten Spline keine Berechnungen durchgeführt werden können.

Sollte dieses nicht befüllt sein und trotzdem ein anderer Tab ausgewählt werden, erscheint im Falle von „Kontaktweg“ und „Zahnräder“ die Meldung „Bitte zuerst Eingabemaske befüllen“, im Falle von „Plot“ bleibt dieser einfach leer.

Nach Klick auf den „Bestätigen“-Button, wird c berechnet und die anderen Tabs können genutzt werden.



Modul:	1
Anzahl Zähne Ritzel:	10
Übersetzungsverhältn...	1

Bestätigen

Abbildung 3.8: Eingabefelder des Programms

Die Zeichenfläche

Das zweite Fenster des Programms stellt die Zeichenfläche für den Kontaktweg dar. Wie in Abbildung 3.9 dargestellt, ist ein einfaches Koordinatenkreuz gegeben, auf dessen Schnittpunkt der Wälzpunkt liegt und beide Wälzkreise sind schemenhaft vorgezeichnet.

Mit der linken Maustaste können Knoten für das Kontrollpolygon (grün und rechteckig) hinzugefügt, mit der rechten Maustaste gelöscht werden. Mit dem Mausekranz kann der Grad des Spline erhöht, bzw. erniedrigt werden. Mit der Taste „n“ können alle Knoten, gelöscht werden. Das Löschen und Verschieben der Knoten schließt natürlich den blauen Kontrollknoten aus.

Der Grad wird oben links angezeigt und ist nach unten durch 1 und nach oben durch 10 beschränkt. Die Beschränkung nach unten erfolgt aufgrund der Differenzierbarkeit. Die Beschränkung nach oben ist nicht zwingend erforderlich und kann jederzeit aufgehoben werden, meiner Ansicht nach werden aber nie Grade größer 10 benötigt. Der voreingestellte Grad ist 3.

Sind dem Grad entsprechend Knoten vorhanden, wird der Kontaktweg vorberechnet. Da dieser durch den Wälzpunkt laufen muss, wird nach dieser Vorbereitung eine Korrektur wie in Kapitel 3.1.2 erklärt durchgeführt. Anschließend werden die Punkte des Kontaktwegs gezeichnet (grau und rund).

Sollte die Eingabemaske vorher nicht befüllt worden sein, erscheint die Meldung „Bitte zuerst Eingabemaske befüllen“ anstatt des Feldes zum Zeichnen.

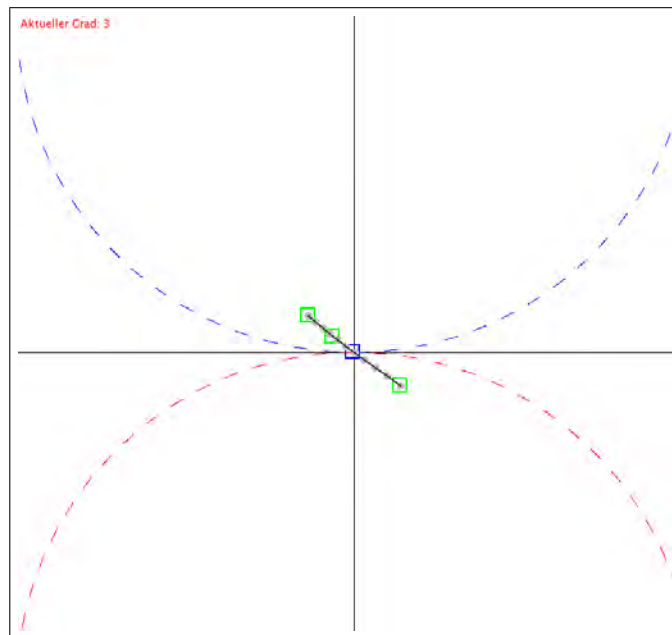


Abbildung 3.9: Zeichenpanel des Programms

Der Plot

Nachdem der Spline gezeichnet wurde, sieht man unter dem Tab „Plot“ nach den Berechnungen den Kontaktweg (grün dargestellt) und die Profillinien (blau und rot dargestellt) zusammen. Diese Ausgabe ist in Abbildung 3.10 zu sehen. Hier sind auch die Wälzkreise entsprechend der Eingabe korrekt skaliert und mit den selben Farben der entsprechenden Profillinie einem Zahnrad zugeordnet. Die Darstellung erfolgt in einem Koordinatensystem.

Auf diesem Fenster können die Knoten nicht verschoben werden, es dient lediglich der Ansicht der Berechnungen. Der Plot-Tab besitzt eine Zoom-Funktion, sodass es möglich ist, sich die Kurven genauer zu betrachten.

Zudem erscheint nach Rechtsklick auf den Plot ein Kontextmenü, in dem unter anderem das Bild kopiert, gedruckt oder als PNG, SVG oder PDF gespeichert werden kann.

Dieser Plot-Tab beruht auf der freien Java Bibliothek JFreeChart.

Sollte die Eingabemaske vorher nicht befüllt worden sein oder wurde kein Kontaktweg gezeichnet, bleibt dieser Tab einfach leer.

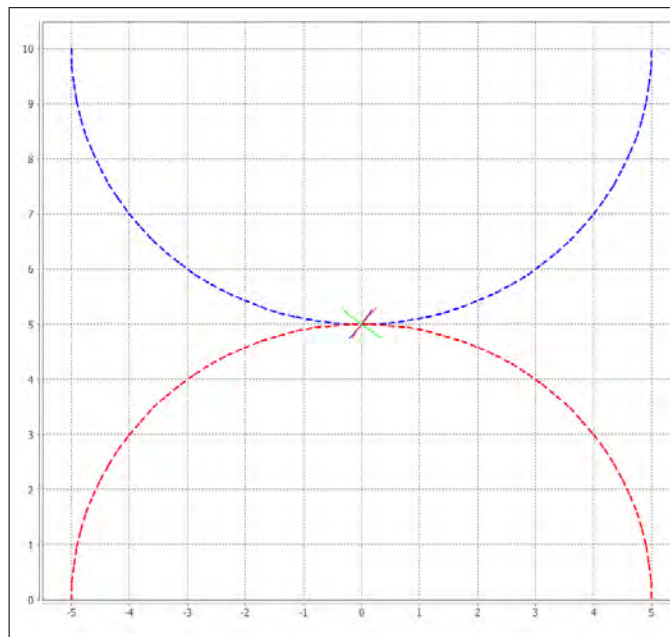


Abbildung 3.10: Plot des Programms

Die Anzeige der Zahnräder

Da man sich mit dem Plot relativ wenig unter einem Zahnrad vorstellen kann, sind im nächsten Fenster, wie man in Abbildung 3.11 die Profillinien zu beiden Zahnrädern zusammengesetzt.

Allerdings wurde ja nur ein Zahn je Zahnrad berechnet, für die Anzeige benötigt man aber alle und alle korrekt gedreht.

Sei dafür n die Anzahl an Zähnen des jeweiligen Zahnrades. Der vorhandene Zahn muss nun noch $(n - 1)$ -mal um den Mittelpunkt des Zahnrades gedreht werden.

Der Winkel zwischen einem Zahn und dem nächsten berechnet sich im Bogenmaß einfach durch

$$\alpha = \frac{2 \cdot \pi}{n}.$$

Die Drehung erfolgt nun iterativ, indem der erste Zahn und danach immer der zuletzt berechnete mit der allgemeinen Drehmatrix

$$\begin{pmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{pmatrix}$$

multipliziert wird.

Zudem erfolgte die Berechnung der Profillinien so, dass der Kontakt zwischen beiden Zahnradern vertikal stattfindet, wie im Plot zu sehen. Auf der Anzeige findet dieser aber horizontal statt, also werden beiden Zahnräder dementsprechend gedreht. Der Wälzpunkt c liegt immer im Zentrum des Tabs, unabhängig der Größe der Zahnräder, da dies der interessante Bereich der Darstellung ist.

Sollte dieser Tab ausgewählt werden, ohne vorher die Eingabemaske zu befüllen, erscheint die Meldung „Bitte zuerst Eingabemaske befüllen“.

Sollte er ausgewählt werden und die Eingaben wurden getätigt, aber kein Kontaktweg gezeichnet, erscheint „Bitte zuerst Kontaktweg erstellen“.

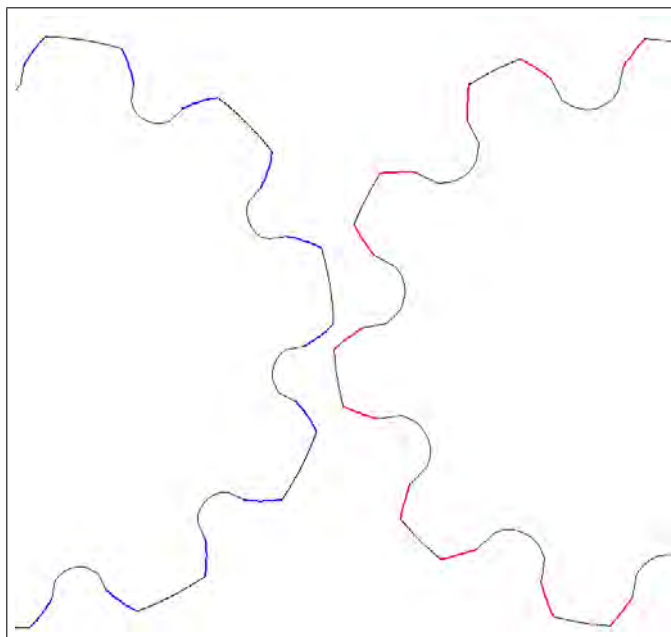


Abbildung 3.11: Zahnradanzeige des Programms

4 Zusammenfassung und Ausblick

Nachdem in dieser Arbeit das Thema behandelt wurde, wie ein Zahnrad aussieht, wenn der Kontaktweg frei formbar ist, bleibt zu überlegen, ob diese Methode für die Produktion Sinn macht. Es wurden Zahnräder erstellt, nachdem der Kontaktweg mit Hilfe einer Spline-Kurve gezeichnet wurde und daraus resultierend die Profillinien errechnet wurden. Die Frage jedoch ist, ob dieses Verfahren sinnvolle, was bedeutet, nützliche Zahnräder für allgemeine Anwendungen kreiert, oder ob die so generierten Zahnräder mehr Nachteile als Vorteile haben.

Meiner Ansicht nach hat diese Form der Konstruktion keinen Vorteil, außer den Freiheiten bei der Erstellung des Kontaktweges, gegenüber den schon genutzten Kontaktwegen einer Evolvente, für Verzahnungen mit konstanter Krafrichtung, die unempfindlich bezüglich des Achsabstandes ist, oder einer Zykloide, welche zwar empfindlicher gegenüber dem Achsabstand ist, dafür aber für Verzahnungen mit veränderlicher Krafrichtung genutzt wird.

Den Kontaktweg als Spline zu konstruieren, erzeugt zwar neue Formen von Zähnen, jedoch entstehen dadurch keine neuen Pluspunkte.

Eine weitere interessante Form der Zahnradkonstruktion wären Zahnräder mit asymmetrischen Zähnen, bei denen also eine Profillinie länger ist als die andere oder vielleicht eine der beiden mit einer Evolvente und die Zweite mit einer Zykloide als Kontaktweg erzeugt wurde.

So oder so gibt es im Bereich der Zahnräder noch vieles, was erforscht werden kann.

A Literatur

- [1] F.M. Feldhaus. *Die Technik*. Hrsg. von F.M. Feldhaus. R. Löwit Verlag Wiesbaden, 1914.
- [2] Albert Neuburger. *Die Technik des Altertums*. Hrsg. von Albert Neuburger. R. Voigländer Verlag, Leipzig, 1919.
- [3] Prof. Dr. Tomas Sauer. *Einführung in die Numerische Mathematik*. 2013.
- [4] Dr. Florian Zimmermann. *Explizite Berechnung kugelkonjugierter Kegelradpaare durch das sphärische Verzahnungsgesetz*. 2012.

B Abbildungsverzeichnis

2.1	Spline Basisfunktionen von Grad 0 und Grad 1	
	Quelle: Philipp Dühning	6
2.2	Spline Basisfunktionen von Grad 2 und Grad 3	
	Quelle: Philipp Dühning	6
2.3	Grafik eines Spline	
	Quelle: Philipp Dühning	8
2.4	Ein Stirnradpaar	
	Quelle: Wikipedia, Benutzer Honina	10
2.5	Skizze einer Göpelmühle	
	Quelle: Adam Meltzers, Neu verbesserte Mühlenbaukunst, Erster Teil, 3. Auflage, Merseburg, 1805	11
2.6	Wälzfräsen	
	Quelle: Wikipedia, Benutzer Kolossos	11
3.1	Spline ohne Fixierung	
	Quelle: Philipp Dühning	16
3.2	Spline mit Fixierung	
	Quelle: Philipp Dühning	16
3.3	Ellipse für Konstruktion	
	Quelle: Philipp Dühning	21
3.4	Drehung der Ellipse	
	Quelle: Philipp Dühning	22
3.5	Schematische Darstellung des Zahnkopfes	
	Quelle: Philipp Dühning	23
3.6	Schematische Darstellung φ_S, φ_E	
	Quelle: Philipp Dühning	24
3.7	Tabbar des Programms	
	Quelle: Philipp Dühning	25
3.8	Eingabefelder des Programms	
	Quelle: Philipp Dühning	26
3.9	Zeichenpanel des Programms	
	Quelle: Philipp Dühning	27

3.10	Plot des Programms	
	Quelle: Philipp Dühning	28
3.11	Zahnradanzeige des Programms	
	Quelle: Philipp Dühning	29
C.1	Plot und Zahnradarstellung der Evolventenverzahnung	
	Quelle: Philipp Dühning	39
C.2	Kontaktweg des „Teufel-Zahns“	
	Quelle: Philipp Dühning	40
C.3	Plot des „Teufel-Zahns“	
	Quelle: Philipp Dühning	41
C.4	Die „Teufel-Zähne“	
	Quelle: Philipp Dühning	42

C Anhang

C.1 Ausführliches Beispiel zur Spline-Berechnung

Dieses Kapitel führt die Berechnung eines Spline mit konkreten Werten vor. Seien folgende Punkte in \mathbb{R}^2 gegeben:

- $(0, 0)$
- $(-2, 1)$
- $(-1, 2)$
- $(1, 3)$
- $(2, 2)$

Damit ist die Anzahl an Punkten $n = 5$. Sei weiterhin ein kubischer Grad vorgegeben, also $k = 3$.

Nun gilt es den Knotenvektor zu bestimmen. Es ist die Formel gegeben

$$t_i = \begin{cases} 0, & \text{für } 0 \leq i \leq k \\ i - k, & \text{für } k < i < n \\ n - k, & \text{für } n \leq i \leq n + k \end{cases}$$

Damit folgt:

$$T = [0, 0, 0, 0, 1, 2, 2, 2, 2]$$

Für einen Wert $t \in T$ gilt nun

$$P(t) = \sum_{i=0}^{n-1} N_{i,k}(t) \cdot P_i = \sum_{i=0}^4 N_{i,3}(t) \cdot P_i$$

Nun müssen die Werte der Basisfunktionen $N_{0,3}(t), N_{1,3}(t), N_{2,3}(t), N_{3,3}(t), N_{4,3}(t)$ bestimmt werden.

Sei für dieses Beispiel die Stelle $t = 1 \in T$. Explizit vorgerechnet wird nur das Beispiel $N_{3,3}(1)$, die anderen folgen dem Schema analog.

Betrachten wir dafür die Rekursionsformel

$$N_{i,0}(t) = \begin{cases} 1, & \text{für } t_i \leq t < t_{i+1} \\ 0, & \text{sonst} \end{cases}$$

$$N_{i,k}(t) = \frac{t - t_i}{t_{i+k} - t_i} \cdot N_{i,k-1}(t) + \frac{t_{i+k+1} - t}{t_{i+k+1} - t_{i+1}} \cdot N_{i+1,k-1}(t)$$

Die Formel lautet mit den Werten eingesetzt:

$$\begin{aligned} N_{3,3}(1) &= \frac{1 - t_3}{t_{3+3} - t_3} \cdot N_{3,2}(1) + \frac{t_{3+3+1} - 1}{t_{3+3+1} - t_{3+1}} \cdot N_{4,2}(1) \\ &= \frac{1 - 0}{2 - 0} \cdot N_{3,2}(1) + \frac{2 - 1}{2 - 1} \cdot N_{4,2}(1) \\ &= \frac{1}{2} \cdot N_{3,2}(1) + N_{4,2}(1) \end{aligned}$$

Betrachten wir $N_{3,2}(1)$. Es gilt eingesetzt:

$$\begin{aligned} N_{3,2}(1) &= \frac{1 - t_3}{t_5 - t_3} \cdot N_{3,1}(1) + \frac{t_6 - 1}{t_6 - t_4} \cdot N_{4,1}(1) \\ &= \frac{1 - 0}{2 - 0} \cdot N_{3,1}(1) + \frac{2 - 1}{2 - 1} \cdot N_{4,1}(1) \\ &= \frac{1}{2} N_{3,1}(1) + N_{4,1}(1) \end{aligned}$$

Betrachten wir $N_{3,1}(1)$. Es gilt eingesetzt:

$$\begin{aligned} N_{3,1}(1) &= \frac{1 - t_3}{t_4 - t_3} \cdot N_{3,0}(1) + \frac{t_5 - 1}{t_5 - t_4} \cdot N_{4,0}(1) \\ &= \frac{1 - 0}{1 - 0} \cdot N_{3,0}(1) + \frac{2 - 1}{2 - 1} \cdot N_{4,0}(1) \\ &= N_{3,0}(1) + N_{4,0}(1) \end{aligned}$$

$N_{3,0}(1) = 0$, da $t_3 \leq 1 < t_4 \Leftrightarrow 0 \leq 1 < 1$ nicht erfüllt.

$N_{4,0}(1) = 1$, da $t_4 \leq 1 < t_5 \Leftrightarrow 1 \leq 1 < 2$ erfüllt.

Damit folgt:

$$N_{3,1}(1) = N_{3,0}(1) + N_{4,0}(1) = 1$$

$$N_{3,2}(1) = \frac{1}{2} + N_{4,1}(1)$$

Betrachten wir $N_{4,1}(1)$. Es gilt eingesetzt:

$$\begin{aligned} N_{4,1}(1) &= \frac{1-t_4}{t_5-t_4} \cdot N_{4,0}(1) + \frac{t_6-1}{t_6-t_5} \cdot N_{5,0}(1) \\ &= \frac{1-1}{2-1} \cdot N_{4,0}(1) + \frac{2-1}{2-2} \cdot N_{5,0}(1) = 0 \end{aligned}$$

Damit folgt:

$$\begin{aligned} N_{3,2}(1) &= \frac{1}{2} + N_{4,1}(1) = \frac{1}{2} \\ N_{3,3}(1) &= \frac{1}{4} + N_{4,2}(1) \end{aligned}$$

Betrachten wir $N_{4,2}(1)$. Es gilt eingesetzt:

$$\begin{aligned} N_{4,2}(1) &= \frac{1-t_4}{t_6-t_4} \cdot N_{4,1}(1) + \frac{t_7-1}{t_7-t_5} \cdot N_{5,1}(1) \\ &= \frac{1-1}{2-1} \cdot N_{4,1}(1) + \frac{2-1}{2-2} \cdot N_{5,1}(1) = 0 \end{aligned}$$

Damit folgt:

$$N_{3,3}(1) = \frac{1}{4}$$

Um jetzt den Wert des Spline bei $t = 1$ zu berechnen, müssen nach dem selben Schema $N_{0,3}(1), N_{1,3}(1), N_{2,3}(1), N_{4,3}(1)$ berechnet werden und dann mit den entsprechenden P_i multipliziert aufsummiert werden. Es gilt analog berechnet:

- $N_{0,3}(1) = 0$
- $N_{1,3}(1) = 0.25$
- $N_{2,3}(1) = 0.5$
- $N_{3,3}(1) = 0.25$
- $N_{4,3}(1) = 0$

Folglich hat der Spline bei $t = 1$ den Wert:

$$P(1) = \sum_{i=0}^4 N_{i,3}(1) \cdot P_i$$

Dies ergibt berechnet $(-0.75, 2)$.

C.2 Ausführliches Beispiel zur Ableitung eines Spline

Seien für dieses Beispiel die selben Eckdaten gegeben, wie bei dem voran gegangenen. Dann berechnen wir zuerst die Matrix G_1 durch $G_1 = kD_n\Delta_k T_1$.

Für D_n gilt: D_n ist die Differenzenmatrix in $\mathbb{R}^{n \times n-1}$, hat hier also die Dimension 5×4 und folgende Form:

$$D_n = D_5 = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \in \mathbb{R}^{5 \times 4}$$

$\Delta_k T_1$ wird folgendermaßen gebildet:

$$\Delta_k T_1 = \Delta_3 T_1 = \text{diag}[(t_{j+3} - t_j)^{-1} : j = 1, \dots, 4] \in \mathbb{R}^{n-1 \times n-1}$$

Die einzelnen Werte t_j sind im Knotenvektor zu finden. Durch einfaches Nachrechnen erhält man:

$$\Delta_3 T_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0 \\ 0 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \in \mathbb{R}^{4 \times 4}$$

Da nun alles bekannt ist, lässt sich G_1 durch einfache Matrizenmultiplikation nach obiger Formel errechnen. Die Dimensionen der Grundmatrizen passen überein, es gilt also für die Matrix G_1 :

$$G_1 = 3 \cdot \begin{bmatrix} -1 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0 \\ 0 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$G_1 = \begin{bmatrix} -3 & 0 & 0 & 0 \\ 3 & -1.5 & 0 & 0 \\ 0 & 1.5 & -1.5 & 0 \\ 0 & 0 & 1.5 & -3 \\ 0 & 0 & 0 & 3 \end{bmatrix} \in \mathbb{R}^{5 \times 4}$$

Für die Berechnung der Ableitung wird die Matrix $N_k^{(1)}(X|T) = G_1 N_{k-1}(X|T_1)$ benötigt. G_1 wurde oben schon berechnet, wie sieht nun $N_{k-1}(X|T_1)$ aus?

Für diese gilt nach Definition:

$$N_{k-1}(X|T_1) = [N_{i,k-1}(x|T_1) : i = 0, \dots, n-2, x \in X] \in \mathbb{R}^{n-1 \times \#X}$$

In unserem Beispiel wollen wir sie wegen höherer Genauigkeit im Intervall $[0; 2]$ diskret alle 0.5 Punkte auswerten. Damit hat sie eine Dimension von 4×5 und folgende Gestalt:

$$\begin{bmatrix} N_{0,2}(0) & N_{0,2}(0.5) & N_{0,2}(1) & N_{0,2}(1.5) & N_{0,2}(2) \\ N_{1,2}(0) & N_{1,2}(0.5) & N_{1,2}(1) & N_{1,2}(1.5) & N_{1,2}(2) \\ N_{2,2}(0) & N_{2,2}(0.5) & N_{2,2}(1) & N_{2,2}(1.5) & N_{2,2}(2) \\ N_{3,2}(0) & N_{3,2}(0.5) & N_{3,2}(1) & N_{3,2}(1.5) & N_{3,2}(2) \end{bmatrix} \\ = \begin{bmatrix} 1 & 0.25 & 0 & 0 & 0 \\ 0 & 0.625 & 0.5 & 0.125 & 0 \\ 0 & 0.125 & 0.5 & 0.625 & 0 \\ 0 & 0 & 0 & 0.25 & 1 \end{bmatrix}$$

Nun kann $N_k^{(1)}$ durch Multiplikation $G_1 N_{k-1}(X|T_1)$ berechnet werden und es gilt:

$$N_k^{(1)} = \begin{bmatrix} -3 & -0.75 & 0 & 0 & 0 \\ 3 & -0.1875 & -0.75 & -0.1875 & 0 \\ 0 & 0.75 & 0 & -0.75 & 0 \\ 0 & 0.1875 & -0.75 & 0.1875 & -3 \\ 0 & 0 & 0 & 0.75 & 3 \end{bmatrix}$$

Abschließend berechnen sich die Koordinaten der Ableitung mittels

$$CN_k^{(1)}$$

wobei C die Matrix der Kontrollpunkte ist also die Form

$$C = \begin{bmatrix} 0 & -2 & -1 & 1 & 2 \\ 0 & 1 & 2 & 3 & 2 \end{bmatrix}$$

aufweist.

Die Ableitung des Spline hat somit folgende x- und y-Koordinaten:

$$Spline_{der} = CN_k^{(1)} = \begin{bmatrix} -6 & -0.1875 & 2.25 & 2.8125 & 3 \\ 3 & 1.875 & 1.5 & 0.375 & -3 \end{bmatrix}$$

C.3 Nachbau einer Evolventenverzahnung

Im Programm der Bachelorarbeit ist es weder möglich den Winkel, noch die Längen der beiden Evolvententeilstücke konkret einzugeben. Lediglich Modul, Übersetzungsverhältnis und die Zähnezahzahl lassen sich eingeben. Beim Winkel und der Länge wird das Bild des Plots als Anhaltspunkt für eine sinnvolle Evolventenverzahnung genommen. Die Nachbildung leichte Abweichungen enthalten.

Die Eingabedaten sind:

- **Modul:** 10
- **Zähne Ritzel:** 15
- **Übersetzungsverhältnis:** 3

Dies ist der Plot der Profillinien und die Zahnradarstellung:

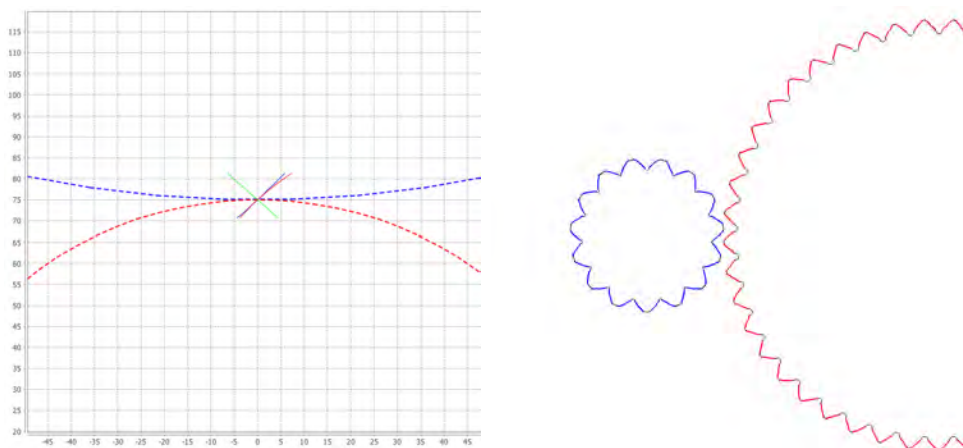


Abbildung C.1: Plot und Zahnradarstellung der Evolventenverzahnung

C.4 „Teufel-Zahn“ - Eine weniger sinnvolle Verzahnung

Natürlich ist es bei dieser Form der Zahngenerierung durch die gegebenen Freiheiten nicht nur möglich, einen sinnvollen Kontaktweg, z.B. ähnlich dem einer Evolvente zu erstellen, sondern allerhand Unsinn als Kontaktweg darzustellen.

Nachfolgend ist so ein Beispiel gegeben. Wie sieht wohl der Zahn eines Zahnrades aus, dessen Kontaktweg einen Bogen macht, sozusagen zu einem x-Wert zwei y-Werte besitzt. Anhand der Screenshots ist zu sehen, dass dies wohl eher keine geeignete Art der Verzahnung wäre. Die Eingaben sind für dieses Beispiel:

- **Modul:** 1
- **Zähne Ritzel:** 10
- **Übersetzungsverhältnis:** 1

Der Kontaktweg ist wie folgt gezeichnet:

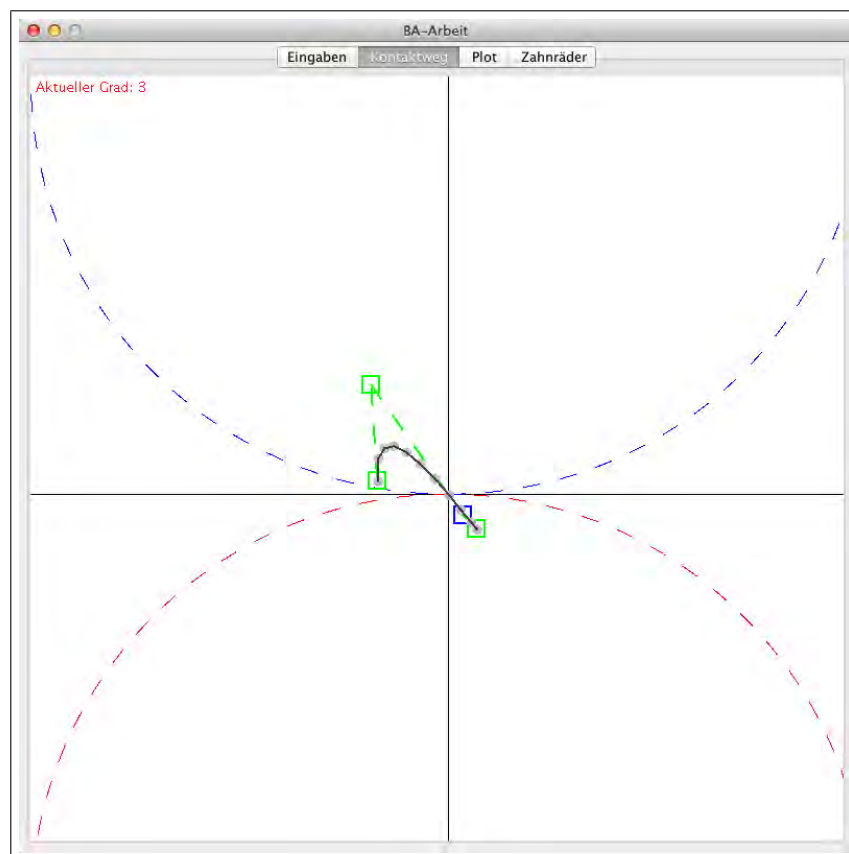


Abbildung C.2: Kontaktweg des „Teufel-Zahns“

Daraus resultiert folgender Plot:

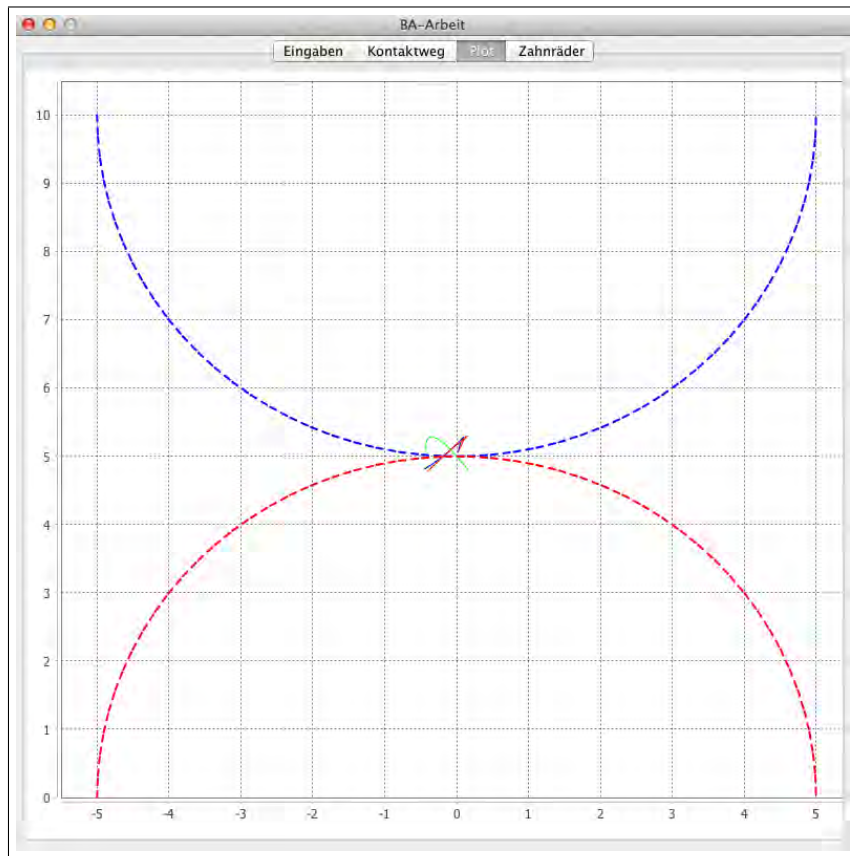


Abbildung C.3: Plot des „Teufel-Zahns“

Und die Zähne sehen wie folgt aus:

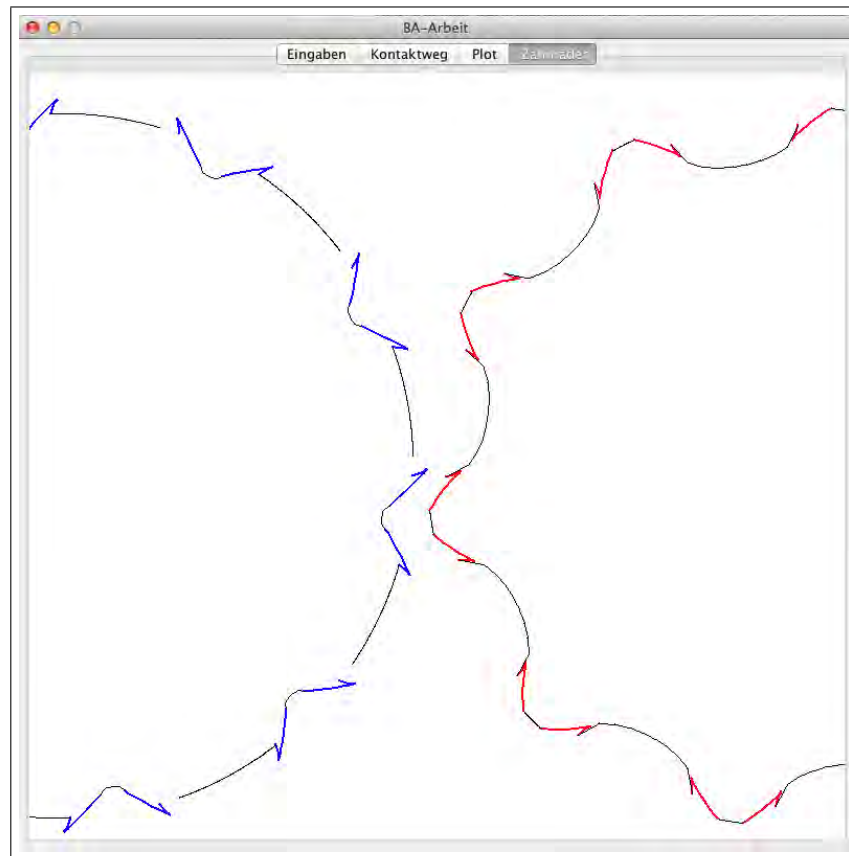


Abbildung C.4: Die „Teufel-Zähne“

Man sieht woher der Name „Teufel-Zahn“ kommt und warum diese Verzahnung relativ unsinnig ist. Die Zähne laufen beim Ritzel im Zahnkopf so spitz zu, dass diese beim Abrollen auf ein anderes Zahnrad einfach abbrechen würden, vertragen also keine Last.

Somit sieht man, dass man allerhand unsinnige Zahnräder erstellen kann, wenn man dem Kontaktweg so viele Freiheiten gibt, wie es z.B. durch die Erstellung mit einem Spline der Fall ist.

Erklärung

Name, Vorname des
Studierenden:

Dühning, Philipp

Universität Passau,
Fakultät für Informatik und Mathematik

Hiermit erkläre ich, dass ich die Arbeit selbstständig verfasst, noch nicht anderweitig für Prüfungszwecke vorgelegt, keine anderen als die angegebenen Quellen oder Hilfsmittel benützt, sowie wörtliche und sinngemäße Zitate auch als solche gekennzeichnet habe.

.....
(Datum)

.....
(Unterschrift des Studierenden)