

Erstellung eines E-Learning-Moduls zum
Thema Überlagerungen und
Fundamentalgruppe

Michela Riganti

3. September 2007

Inhaltsverzeichnis

I	Ziel und Inhalte	4
II	Theoretische Einführung in die Thematiken	5
1	Homotopie	5
2	Fundamentalgruppe	5
3	Windungszahl	8
4	Überlagerungen	9
5	Hochheben von Wegen	12
6	Monodromieoperation	14
7	Deckbewegungen	15
III	Fundamentalgruppe von \mathbb{C} ohne n Punkte	16
1	Homotopieäquivalenz zwischen X und einem Bouquet B	16
2	Die Isomorphie zwischen $\pi_1(X, x_0)$ und F_n	18
IV	Algorithmen und Anwendungen	23
1	Algorithmen	24
2	Applets	31
2.1	Applet 1: Gehören zwei Schleifen zur gleichen Homotopieklasse?	31
2.2	Applet 2: Nullomotopie vs Nullhomologie	33
2.3	Projekt für Applet 3: Universelle Überlagerung und Fundamentalgruppe	35

3	Input-Probleme und Lösungsvorschläge	36
V	Anlage und Kontakte	40

Teil I

Ziel und Inhalte

Das Ziel dieser Bachelor-Arbeit ist die Erstellung eines E-Learnings Moduls, das als Zusatzmittel für eine Vorlesung über Topologie ausgerichtet ist und speziell die Themen Fundamentalgruppe und Überlagerungen mittels interaktiven Fenstern behandelt.

Dieses Dokument enthält:

- Eine theoretische Einführung in die Thematiken.
- Einige Algorithmen zur Berechnung der Homotopieklasse einer Schleife in $\mathbb{C} \setminus \{x_1, \dots, x_n\}$.
- Vorschläge für drei Applets, die diese Algorithmen benutzen, mit einigen didaktischen Bemerkungen.
- Ein mit Windows kompatibles Programm, mit dem der Benutzer Beispiele sehen und konstruieren kann, um das Verständnis der Theorie auf anschauliche Weise zu verbessern.

Teil II

Theoretische Einführung in die Thematiken

1 Homotopie

Def 1. Seien X und Y zwei topologische Räume, $I := [0, 1]$ und $f, g : X \rightarrow Y$ zwei stetige Abbildungen.

- Eine **Homotopie von f nach g** ist eine stetige Abbildung $h : X \times I \rightarrow Y$ mit:

$$\left. \begin{array}{l} h(x, 0) = f(x) \\ h(x, 1) = g(x) \end{array} \right\} \forall x \in X$$

- Man sagt, dass f zu g **homotop** ist ($f \simeq_h g$), falls eine Homotopie von f nach g existiert.
- Sei $A \subset X$. Man nennt h **Homotopie relativ A** , falls $h(a, t) = f(a) = g(a) \forall t \in [0, 1], \forall a \in A$.

Def 2. Eine stetige Abbildung $f : X \rightarrow Y$ heisst **Homotopieäquivalenz** zwischen X und Y , wenn sie ein Homotopieinverses besitzt, d.h. eine stetige Abbildung $g : Y \rightarrow X$ mit:

$$\begin{array}{l} g \circ f \simeq Id_X \\ f \circ g \simeq Id_Y \end{array}$$

2 Fundamentalgruppe

Die Fundamentalgruppe ist ein algebraisches Objekt, das man zu jedem topologischen Raum X mit Basispunkt $x_0 \in X$ zuordnen kann.

Diese Gruppe beschreibt die topologischen Eigenschaften, die sich bei stetigen Verformungen nicht ändern. Durch Berechnung der Fundamentalgruppe kann man entscheiden, ob zwei topologische Räume sich ineinander stetig verformen lassen.

Def 3. Sei X ein topologischer Raum. Eine Weg von $x_0 \in X$ nach x_0 heisst **Schleife an x_0** .

Def 4. Ein geschlossener Weg heißt **nullhomotop** genau dann, wenn er homotop zu einem konstanten Weg ist.

Def 5. Sei $\Omega(X, x_0)$ die Menge der Schleifen um x_0 . Die **Fundamentalgruppe von X zum Basispunkt x_0** ist:

$$\pi_1(X, x_0) := \Omega(X, x_0) / \simeq$$

wobei \simeq die Homotopie relativ $\{0,1\}$ bezeichnet.

Bem 1. $\pi_1(X, x_0)$ ist eine Gruppe. Die Gruppenoperation ist durch:

$$[\sigma] * [\tau] := [\sigma * \tau]$$

definiert, wobei σ und τ zwei Schleifen um x_0 sind und

$$\sigma * \tau := \begin{cases} \sigma(2t), t \in [0, 1/2] \\ \tau(2t - 1), t \in [1/2, 1] \end{cases}$$

Satz 1. *Homotopieäquivalente Räume besitzen isomorphe Fundamentalgruppen.*

Beispiel 1. Sei $X = S^1$. Es gilt: $\pi_1(S^1, 1) \cong \mathbb{Z}$. Anschaulich sieht man, dass die Homotopieklasse einer Schleife σ um 1 nur von der Anzahl und Richtung der Drehungen um 0 hängt. Für einen formel Beweis, siehe [SKP, S. 51] oder [Hatcher, S. 29]

Bem 2. (Produkt) Sei $(x_0, y_0) \in X \times Y$. Dann gilt:

$$\pi_1(X \times Y, (x_0, y_0)) = \pi_1(X, x_0) \times \pi_1(Y, y_0)$$

Beispiel 2. (Die Fundamentalgruppe des Torus T^2)

Sei $x_0 \in T^2$. Da $T^2 \cong S^1 \times S^1$ folgt:

$$\pi_1(T^2, x_0) = \pi_1(S^1 \times S^1, (z_0, z_1)) = \pi_1(S^1, z_0) \times \pi_1(S^1, z_1) = \mathbb{Z} \times \mathbb{Z}$$

Diese Anwendung zeigt, wie die Zerlegung eines Raumes (in einfachere Räume) zur Berechnung der Fundamentalgruppe helfen kann. Eine weitere Möglichkeit, zur Vereinfachung der Berechnung ist, den anfänglichen Raum zu einem einfachen homotopieäquivalenten Raum zurückzuführen. Diese Methode benützt man zum Beispiel zur Berechnung der Fundamentalgruppe von \mathbb{C} ohne n Punkte.

Def 6. Ein **Bouquet von n Kopien von S^1** ist die Einpunktverheftung von n Kopien von S^1 an einem Punkt, mit der Quotiententopologie versehen.

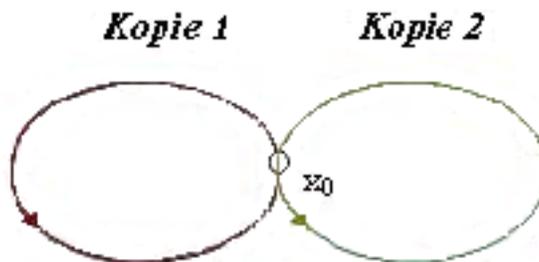
Def 7. Seien a_1, a_2, \dots, a_n paarweise verschiedene Symbole. Die Menge aller endlichen Wörter, die man anhand des Alphabets $\{a_1, \dots, a_n, a_1^{-1}, \dots, a_n^{-1}\}$ bilden kann und keine der Teilwörter $a_i a_i^{-1}$ oder $a_i^{-1} a_i$ enthalten, heisst **freie Gruppe über $\{a_1, a_2, \dots, a_n\}$** und wird mit F_n bezeichnet.

Satz 2. (*Bouquet*) Sei B der topologischer Raum, der aus einem Bouquet von n Kopien von S^1 besteht, indem man einen Punkt aus jeder Kopie zusammenklebt. $\pi_1(B, x_0)$ ist dann zur freien Gruppe F_n isomorph.

Beweis 1. Siehe [SKP, S. 51]

Beispiel 3. Sei X ein topologischer Raum, der aus zwei Kopien von S^1 , in einem Punkt zusammengeklebt, besteht. Die Fundamentalgruppe $\pi_1(X, x_0)$ ist dann $\forall x_0 \in S$ zur Freien Gruppe mit 2 Erzeuger a_1 und a_2 isomorph.

Abbildung 1: Bouquet für $n=2$



Bem 3. Aus diesem Beispiel folgt, dass die Fundamentalgruppe nicht abelsch ist.

Lemma 1. Man kann trotzdem aus der Fundamentalgruppe eine abelsche Gruppe erhalten, wenn man $A := \pi_1/K(G)$ betrachtet, wo $K(G) := \{[a, b] \mid a, b \in G\}$ die Kommutatorgruppe bezeichnet. A ist dann zur ersten Homologiegruppe isomorph.

3 Windungszahl

Def 8. Sei α eine geschlossene, stückweise glatte Kurve, deren Bild den Punkt $z \in \mathbb{C}$ nicht enthält. Die **Windungszahl** von α bezüglich z ist definiert durch:

$$n(\alpha; z) := \frac{1}{2\pi i} \int_{\alpha} \frac{1}{\zeta - z} d\zeta$$

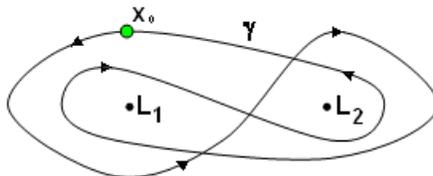
Def 9. Sei $G \subset \mathbb{C}$ ein Gebiet. Eine Schleife γ in G heisst **nullhomolog**, falls $n(\gamma, p) = 0 \forall p \in \mathbb{C} \setminus G$.

Lemma 2. *Jede nullhomotope Schleife ist nullhomolog.*

Beweis 2. Siehe [FB, S. 243]; das Lemma ist eine Folgerung des Cauchy-Integralsatz.

Bem 4. Die Umkehrung gilt nicht! (Siehe Abbildung 2)

Abbildung 2: Ein nullhomologer, aber nicht nullhomotoper Weg



Lemma 3. [FB, S. 241] *Die Windungszahl ist eine ganze Zahl.*

Bem 5. Jede Schleife in der komplexen punktierten Ebene \mathbb{C}^\bullet ist zu einer k -fach durchlaufenen Einheitskreislinie homotop. Die Umlaufzahl eines Wegs $\alpha \subset \mathbb{C}^\bullet$ entspricht der Anzahl Drehungen von α um $\{0\}$ im Gegenuhrzeigersinn.

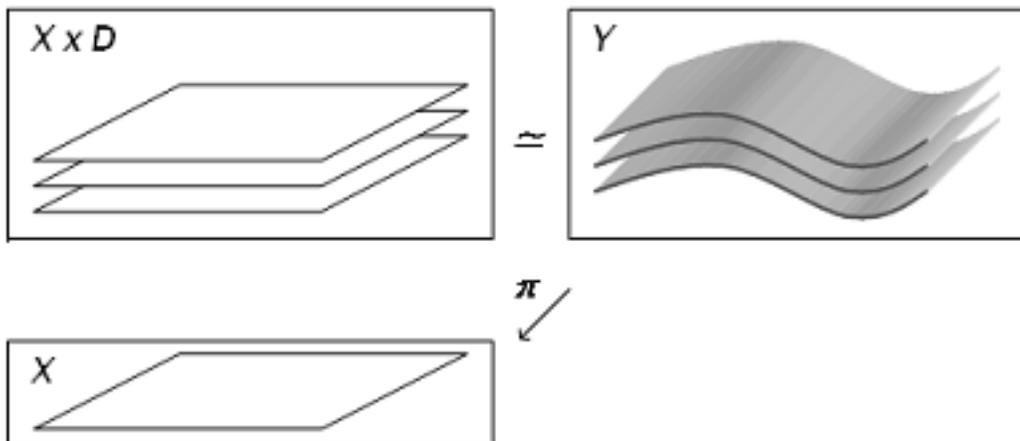
4 Überlagerungen

Def 1. Seien X und Y lokal wegzusammenhängende Räume. Eine **Überlagerung von X** ist eine surjektive, stetige Abbildung $\pi : Y \rightarrow X$ s.d.
 $\forall x \in X \exists U_x$ Umgebung von x , $\exists D_u$ diskreter Raum mit:

$$\begin{array}{ccc}
 U \times D_u & \xrightarrow{\cong} & \pi^{-1}(U) \xrightarrow{\subset} Y \\
 \text{pr}_i \downarrow & \swarrow \pi & \swarrow \pi \\
 U \ni x & \xrightarrow{\subset} & X
 \end{array}$$

Die Mengen $U \times \{d\}$ für $d \in D$ heissen **Blätter** der Überlagerung.

Abbildung 3: Überlagerung



Sprechweise: Oft bezeichnet man mit „Überlagerung“ den überlagerten Raum Y . Eine Überlagerung $\pi : Y \rightarrow X$ erlaubt, einige Eigenschaften des Grundraums X durch die Eigenschaften des überlagerten Raums Y , der üblicherweise einfacher ist, zu untersuchen.

Da es für einem Raum X normalerweise mehr als eine Überlagerung existiert, war es wichtig einen allgemeineren Begriff einzuführen, der alle Überlagerungen zusammenfasst: es handelt sich um die universellen Überlagerung.

Die „*Universalität*“ bezieht sich auf die erste Eigenschaft von Satz 1:

Def 2. Eine **universelle Überlagerung von X** ist eine Überlagerung $\pi : \tilde{X} \rightarrow X$ mit \tilde{X} einfach zusammenhängend.

Satz 1. 1. Falls \tilde{X} eine universelle Überlagerung von X ist, dann ist sie auch Überlagerung jeder anderen zusammenhängenden Überlagerung von X .

2. Jeder topologische Raum X , der:

- wegzusammenhängend
- lokal wegzusammenhängend
- semilokal einfach zusammenhängend

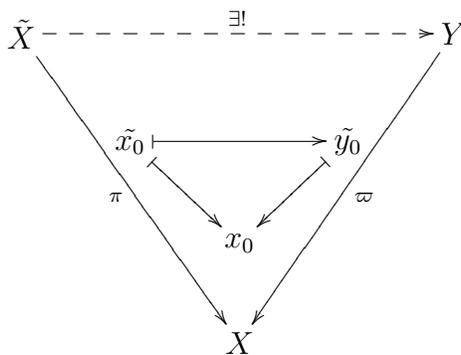
ist, besitzt eine bis auf Homöomorphismus eindeutig bestimmte universelle Überlagerung.

Beispiel 4. Einige klassische Beispiele:

- $p : \mathbb{R} \rightarrow S^1$
 $t \mapsto e^{it}$
- $\varpi : \mathbb{R}^2 \rightarrow S^1 \times S^1$
 $(t, u) \mapsto (e^{it}, e^{iu})$

Satz 2. Sei X ein einfach zusammenhängender topologischer Raum. Dann ist jede Überlagerung $\pi : Y \rightarrow X$ ein Homöomorphismus.

Satz 3. (Universelle Eigenschaft) Seien: $\pi : \tilde{X} \rightarrow X$ eine universelle Überlagerung von X , $\varpi : Y \rightarrow X$ eine Überlagerung, $\tilde{x}_0 \in \pi^{-1}(x_0)$ und $y_0 \in \varpi^{-1}(x_0)$.
 $\Rightarrow \exists! f : \tilde{X} \rightarrow Y$ stetig, mit :
 $f(x_0) = y_0$ und $\varpi \circ f = \pi$.



Lemma 1. *Es folgt aus der universellen Eigenschaft, dass die universelle Überlagerung bis auf Isomorphie eindeutig bestimmt ist.*

Satz 4. *Seien: $\pi : \tilde{X} \rightarrow X$ die universelle Überlagerung von X , $x_0 \in X$ und $\tilde{x}_0 \in \pi^{-1}(x_0)$. Dann:*

- *$\text{Aut}(\tilde{X})$ operiert frei und transitiv auf $\pi^{-1}(x_0)$.*
- *Die Abbildung*

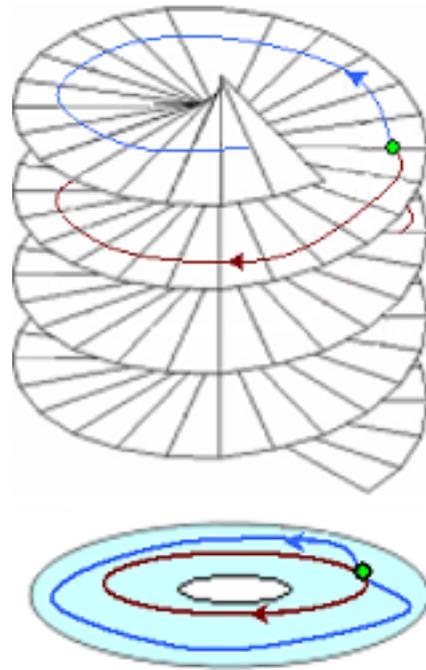
$$m : \text{Aut}(\tilde{X}) \rightarrow \pi_1(x_0)$$

$$f \mapsto [\sigma]$$

*für $f(\tilde{x}_0) = \tilde{x}_0 * [\sigma]$, definiert ein Gruppenhomomorphismus.*

5 Hochheben von Wegen

Abbildung 4: Hochheben

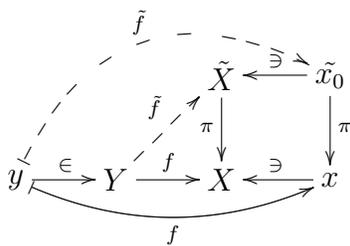


Wenn $\pi : Y \rightarrow X$ eine Überlagerung ist, kann man bei Vorgabe eines Startpunkts y_0 aus der Faser $\pi^{-1}(x_0)$ zu jedem Weg α in X mit Startpunkt x_0 einen Weg in Y zuordnen.

Def 1. Sei $\pi : Y \rightarrow X$ eine Überlagerung mit $\pi(x_0) = \pi(y_0)$. Ein Weg $\tilde{\alpha} : [0, 1] \rightarrow Y$ heisst **Hochhebung von α zum Anfangspunkt y_0** , wenn: $\pi \circ \tilde{\alpha} = \alpha$ und $\tilde{\alpha}(0) = y_0$.

Satz 1. (Hochhebung von $f : Y \rightarrow X$) Seien: $\pi : \tilde{X} \rightarrow X$ eine Überlagerung, $\tilde{x}_0 \in \tilde{X}$, $x_0 := \pi(\tilde{x}_0)$ und sei Y ein zusammenhängender, lokal- und einfach wegzusammenhängender Raum; $y \in Y$.

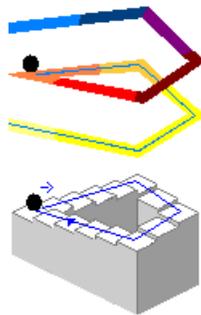
$\Rightarrow \forall f : (Y, y) \rightarrow (X, x_0)$ stetig $\exists! \tilde{f} : (Y, y) \rightarrow (\tilde{X}, \tilde{x}_0)$ stetig mit $\pi(\tilde{f}) = f$.



Kor 1. $\forall \sigma : [0, 1] \rightarrow X$ mit $\sigma(0) = x_0 \exists! \tilde{\sigma} : [0, 1] \rightarrow Y$ s.d. $\pi \circ \tilde{\sigma} = \sigma$ und $\tilde{\sigma}(0) = y_0$.

6 Monodromieoperation

Abbildung 5: Monodromie



Lemma 1. (*Monodromielemma*) Sei $\pi : Y \rightarrow X$ eine Überlagerung. Seien α und β zwei Wege in X , die homotop rel. $\{0,1\}$ sind. Seien $\tilde{\alpha}$ und $\tilde{\beta}$ Hochhebungen von α und β zum gleichen Anfangspunkt y_0 , dann gilt: $\tilde{\alpha}(1) = \tilde{\beta}(1)$.

Bem 1. Es folgt, dass die **Monodromieoperation**

$$y_0 * [\sigma] := \text{Endpunkt von } \tilde{\sigma} \text{ zum Anfangspunkt } y_0 \text{ gehoben}$$

eine wohldefinierte Operation von $\pi_1(X, x_0)$ von rechts auf dem Faser $\pi^{-1}(x_0)$ ist.

Satz 1. Seien $\pi : \tilde{X} \rightarrow X$ eine universelle Überlagerung, $x_0 \in X$ und $\tilde{x}_0 \in \pi^{-1}(x_0)$. Die Abbildung $b : [\sigma] \mapsto \tilde{x}_0 * [\sigma]$ definiert eine Bijektion zwischen \tilde{X} und den Raum der Homotopieklassen relativ $\{0,1\}$ von Wegen mit Anfangspunkt x_0 .

7 Deckbewegungen

Die Deckbewegungsgruppe ist eine Gruppe von Transformationen des überlagerten Raums Y , die zur Berechnung der Fundamentalgruppe des Basisraums X sehr nützlich ist.

Def 1. Sei $\pi : X \rightarrow Y$ eine Überlagerung. Ein Homöomorphismus $\varphi : Y \rightarrow Y$ heisst **Deckbewegung** (oder **Decktransformation**), falls das Diagramm

$$\begin{array}{ccc} Y & \xrightarrow{\varphi} & Y \\ & \searrow \pi & \swarrow \pi \\ & X & \end{array}$$

kommutiert.

Die Kommutativität des Diagramms besagt, dass φ Elemente, innerhalb derselben Faser $\pi^{-1}(x)$ abbildet.

Def 2. Die Gruppe $Deck(Y/X) := \{\text{Deckbewegungen } \varphi : Y \rightarrow Y\}$ heisst **Deckbewegungsgruppe** von π .

Beispiel 5. Sei $p : \mathbb{R} \rightarrow S^1$ die universelle Überlagerung von S^1 .
 $t \mapsto e^{it}$

Für jedes $k \in \mathbb{Z}$ ist die Translation $\varphi_k : \mathbb{R} \rightarrow \mathbb{R}$ eine Deckbewegung. Die Deckbewegungsgruppe ist: $Deck(\mathbb{R}/S^1) = \{\varphi_k | k \in \mathbb{Z}\}$.
 $t \mapsto t + k \cdot 2\pi$

Satz 1. Die Gruppe der Decktransformationen der universellen Überlagerung ist zur Fundamentalgruppe des Basisraums isomorph.

Satz 2. Seien $p : \tilde{X} \rightarrow X$ eine Überlagerung, D ihre Deckbewegungsgruppe, $x \in X$ und $F := p^{-1}(x) \subset \tilde{X}$.

Falls p eine n -blättrige Überlagerung für ein $n \in \mathbb{N}$ ist, so ist D endlich und $|D| = n$.

Satz 3. Sei $p : \tilde{X} \rightarrow X$ eine universelle Überlagerung mit X lokal wegzusammenhängend. Sei $x \in X$ und D die Deckbewegungsgruppe von p .
 $\Rightarrow \pi_1(X, x) \simeq D$.

Teil III

Fundamentalgruppe von \mathbb{C} ohne n Punkte

Der Zweck dieses Teils ist die Untersuchung von $\pi_1(X, x_0)$.

Sei $\Delta := \{L_1, \dots, L_n\}$, wobei $L_i \in \mathbb{C}$ paarweise verschieden sind. Wir interessieren uns nun für den topologischen Raum $X := \mathbb{C} \setminus \Delta$.

1 Homotopieäquivalenz zwischen X und einem Bouquet B

Wenn wir beweisen können, dass X zu einem Bouquet B von n Kopien von S^1 homotopieäquivalent ist, dann folgt, dass $\pi_1(X, x_0)$ zu F_n isomorph ist.

Lemma 1. *X und B sind homotopieäquivalent.*

Beweis 1. Wir werden eine Homotopieäquivalenz zwischen X und B finden.

Seien:

$K := (K_1, \dots, K_n)$ ein Raum, der aus der stetigen Umformung von n Kopien (S_1^1, \dots, S_n^1) von S^1 (an x_0 verklebt) besteht, mit der Eigenschaft, dass $\forall i = 1, \dots, n, L_i \in E_i$, wobei

$$E_i := \{x \in \mathbb{C} \mid \exists \alpha, \text{Weg von } x \text{ nach } L_i \text{ s.d. } (\alpha \cap K_i) = \emptyset\}$$

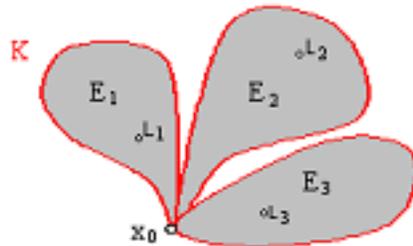
Wir wählen für jedes i ein K_i , s.d E_i sternförmig ist.

Desweiteren definieren wir eine Einbettung $\iota : B \rightarrow X$ durch Identifizierung von S_i^1 mit K_i , und zeigen durch Konstruktion einer Homotopieinversen, dass ι eine Homotopieäquivalenz ist.

- Sei h die Homotopieäquivalenz zwischen dem Bouquet B und den Raum K . Die Schleifen K_1, \dots, K_n entsprechen also $h(S_1^1), \dots, h(S_n^1)$
- $E := \bigcup_{i=1}^n E_i$

Wir betrachten die folgende Funktionen:

Abbildung 6: die oben beschriebene Mengen



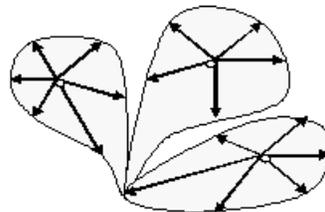
- Sei $p_i : (\bar{E}_i \setminus L_i) \rightarrow K_i$ die Abbildung, die im Zeitintervall $[0, 1]$ ein Punkt von $\bar{E}_i \setminus L_i$ in seiner radialen Projektion auf K_i umformt. Man kann p_i so wählen, dass es eine Homotopieäquivalenz ist, weil $\bar{E}_i \setminus L_i$ zu $D^1 \setminus \{0\} := \{x \in \mathbb{C}^\circ \mid |x| \leq 1\}$ homotopieäquivalent ist und

$$q(t, x) := (1 - t) \cdot x + t \cdot \frac{x}{\|x\|}$$

eine Homotopieäquivalenz zwischen $D^1 \setminus \{0\}$ und S^1 ist.

- $p : \bar{E} \rightarrow K$ mit $p(z) := p_i(z), z \in \bar{E}_i \setminus L_i$

Abbildung 7: $p(z)$



- Sei $\phi : (\mathbb{C} \setminus E) \rightarrow K$ eine starke Deformationsretraktion von $\mathbb{C} \setminus E$ zu K .
- $\chi : (\mathbb{C} \setminus \Delta) \rightarrow K$

$$\chi(z) := \begin{cases} p(z), & z \in \bar{E} \\ \phi(z), & z \notin \bar{E} \end{cases}$$

Aus der Tatsache, dass

$$\forall z_0 \in K \lim_{z \rightarrow z_0} (\phi(z)) = z_0 = p(z_0)$$

können wir die Stetigkeit von χ schliessen.

Desweiteren gilt:

$$\begin{aligned} \chi \circ \iota &\simeq Id_B \text{ und} \\ \iota \circ \chi &\simeq Id_X \end{aligned}$$

χ ist also eine Homotopieinverse von ι . \square

Deswegen kann man mit $\chi(\gamma)$, anstatt γ , und mit dem Raum K , anstatt $\mathbb{C} \setminus D$ arbeiten.

Der grosse Vorteil dieser Vereinfachung ist, dass die Fundamentalgruppe von K bekannt ist, und dass es einfacher ist, für jede Schleife die entsprechende Homotopieklasse zu finden.

2 Die Isomorphie zwischen $\pi_1(X, x_0)$ und F_n

Wir interessieren uns für das folgende

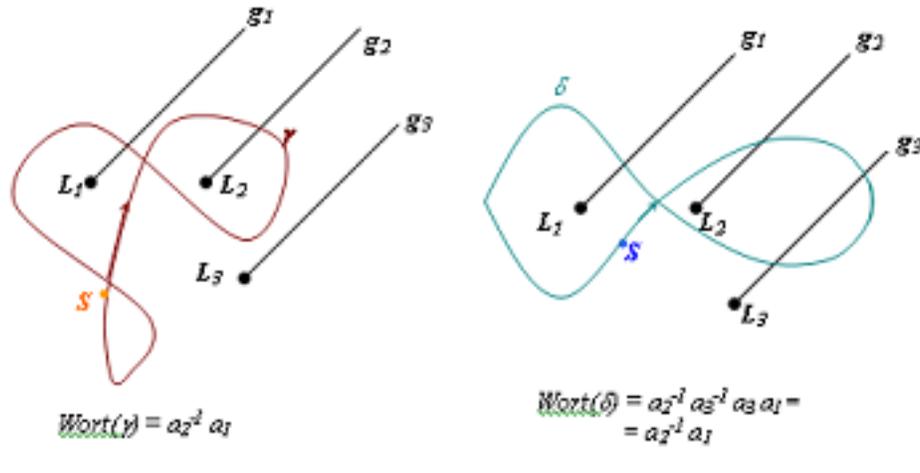
Verfahren 1. *Konstruiere n parallele Halbgeraden (g_1, \dots, g_n) , die von den n Löchern ausgehen, s.d. jede Halbgerade genau ein Loch enthält. Man beginnt dann den Weg γ zu durchlaufen und jedes Mal, dass man die Halbgerade g_i im Uhr- bzw. Gegenuhrzeigersinn schneidet, schreibt man a_i , bzw a_i^{-1} . (Siehe Abbildung 8)*

Bem 1. (Optimale Konfiguration) Die Steigung der Halbgeraden kann aus \mathbb{R} (überabzählbar) gewählt werden und die Anzahl von Konfigurationen, die nicht anpassen (mit zwei verschiedenen Löcher auf derselben Gerade) ist endlich (höchstens $2^{\frac{n(n-1)}{2}}$). Daraus kann man die Existenz einer optimalen Konfiguration von parallele Halbgeraden (d.h. g_i geht aus L_i und enthält kein weiteres Loch) schliessen.

Anschaulich, besagt das so erhaltene Wort , wie viele Drehungen und mit welchem Sinn und Reihenfolge die Schleife um die Löcher macht, und ob sie unten oder oben von L_i durchgeht.

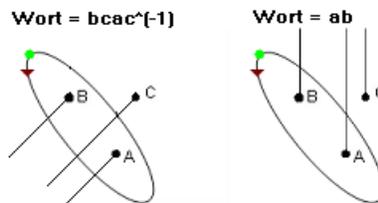
Zur Vereinfachung bezeichnen wir für die folgende Beispiele mit A, B, C, \dots

Abbildung 8: Verfahren 1



die Löcher L_1, L_2, L_3, \dots , und mit a, b, c, \dots die Buchstaben a_1, a_2, a_3, \dots . Man kann leicht sehen, dass dieses Wort, von der gewählten Konfiguration von parallelen Halbgeraden abhängt. Abbildung 9 zeigt wie man durch Drehung der Halbgeraden ein anderes Wort erhält.

Abbildung 9: Isomorphe, aber verschiedene Worte



Lemma 2. Für jedes $n \in \mathbb{N}$ gibt es, bis auf Isomorphien, genau eine freie Gruppe mit n Erzeuger.

Wir wollen nun zeigen, dass Verfahren 1 ein Isomorphismus zwischen $\pi_1(X, x_0)$ und F_n beschreibt:

Bem 2. Es gilt:

- Das aus Verfahren 1 erhaltene Wort gehört zu F_n .

- Für je zwei Schleifen γ und δ um x_0 gilt: $Wort(\gamma * \delta) = Wort(\gamma) \circ Wort(\delta)$.
- Schleifen, die zur gleichen Homotopieklasse gehören, führen zu äquivalenten Wörtern.
- Man kann für jeden Buchstaben a_i eine entsprechende Schleife konstruieren, die nur das Loch L_i umkreist.
- Zwei Schleifen, die zu äquivalenten Wörtern führen, sind homotop.

Aus diesen Gründen kann man schliessen, dass Verfahren 1 ein Isomorphismus zwischen $\pi_1(X, x_0)$ und F_n beschreibt.

Darum gibt es ein Isomorphismus zwischen den zwei „Sprachen“, die aus zwei verschiedene Konfigurationen entstehen. Für die in Abbildung 9 gewählte Konfigurationen ist der Isomorphismus durch die Bilder der Erzeugenden eindeutig bestimmt:

$$\begin{aligned} a &\mapsto b^{-1}a^{-1}c^{-1}acb \\ b &\mapsto b \\ c &\mapsto b^{-1}a^{-1}cab \end{aligned}$$

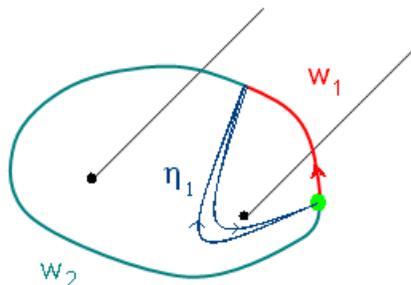
Bem 3. Die Homotopieklasse einer Schleife γ um x_0 besitzt ein Repräsentant, der eine Zusammensetzung von Schleifen, die genau ein Loch umkreisen und genau eine Halbgerade schneiden, ist.

Man kann ein Repräsentant dieser Art wie folgt finden (siehe Abbildung 10):

Verfahren 2. Man zerlegt die Schleife in Wege (w_1, \dots, w_m) , die genau eine Halbgerade schneiden. Sei $p_i \forall i = 1, \dots, m - 1$ der Endpunkt von w_i . Der Raum $\mathbb{C} \setminus \{g_1, \dots, g_n\}$ ist wegzusammenhängend. Darum kann man nach jedem w_i für $i = 1, \dots, m - 1$ eine nullhomotope Schleife η_i , die von p_i zum Basispunkt und dann wieder zu p_i geht, zusammensetzen. Die Schleife $w_1 * \eta_1 * \dots * w_{m-1} * \eta_{m-1} * w_m$ hat die gewünschte Eigenschaft.

Jede nichttriviale Homotopieklasse in $\pi_1(X, x_0)$ hat also die Form $[w_1 * \eta_1 * w_2 * \eta_2 * \dots * w_{m-1} * \eta_{m-1} * w_m]$ für $m \leq n$ und lässt sich darum als Zusammensetzung $[\sigma_1] \circ [\sigma_2] \circ \dots \circ [\sigma_m]$ schreiben, wobei $\sigma_1, \dots, \sigma_m$ Schleifen sind, die genau ein Loch einmal umkreisen und genau eine Halbgerade

Abbildung 10: Die Schleife $w_1 * \eta_1 * w_2$ ist zu $w_1 * w_2$ homotop und besteht aus zwei Schleifen, die je ein Loch umkreisen.



schneiden.

Aus diesem Grund kann man einen (von den Halbgeraden abhängigen) Isomorphismus von $\pi_1(X, x_0)$ nach F_n definieren, indem man zur Homotopieklasse der Schleifen, die nur das Loch L_i einmal umkreisen, den Buchstaben a_i in F_n zuordnet.

Das ist genau der Isomorphismus, der in Verfahren 1 beschrieben wird.

Bem 4. Die Umlaufzahl von γ um L_i ist genau dann gleich null, wenn für jede optimale Konfiguration von Halbgeraden in Verfahren 1 die Anzahl von Buchstaben a_i gleich der Anzahl von Buchstaben a_i^{-1} ist.

Dies erlaubt die Umlaufzahl aus dem von Verfahren 1 konstruierten Wort zu berechnen.

Wenn man sich für die Berechnung der Umlaufzahl nicht interessiert, kann man eine einfachere Version von Verfahren 1 betrachten, die keine Drehung der Halbgeraden benötigt:

Verfahren 3. *Konstruiere für jedes Loch L_i :*

1. *Eine vertikale, nach oben gerichtete Halbgerade, falls diese kein weiteres Loch enthält*
2. *Eine vertikale, nach oben gerichtete Strecke von L_i nach dem ersten getroffenen Loch, der auf derselben Halbgerade liegt, falls die vertikale Halbgerade die von L_i ausgeht weitere Löcher enthält.*

Für jedes $i = 1, \dots, n$ bezeichne s_i die aus L_i konstruierte Halbgerade oder Strecke.

Man beginnt dann den Weg γ zu durchlaufen und jedes Mal, dass man die Halbgerade oder Strecke s_i im Uhr- bzw. Gegenuhrzeigersinn schneidet, schreibt man a_i , bzw. a_i^{-1} .

Bem 5. Bemerkung 4 gilt für Verfahren 3 nicht, weil es nicht für jede Schleife möglich ist, ein Repräsentant der Homotopieklasse zu finden, der eine Zusammensetzung von Schleifen ist, die genau ein Loch umkreisen und genau eine Strecke schneiden.

Teil IV

Algorithmen und Anwendungen

Sei $\Delta := \{L_1, \dots, L_n\} \subset \mathbb{C}$. Wir interessieren uns für den topologischen Raum $X := \mathbb{C} \setminus \Delta$.

In diesem Teil werden einige Algorithmen beschrieben, die die Fundamentalgruppe von X zum Basispunkt x_0 untersuchen.

Speziell, beantworten diese Algorithmen die Frage:

Wann liegen zwei Schleifen in der gleichen Homotopieklasse?

Aus Teil 3, Verfahren 1 und Verfahren 3 kann man zwei Algorithmen ableiten, die als Input eine Schleife um x_0 und als output ein entsprechendes Wort aus der freien Gruppe F_n haben.

Mittels dieser zwei Algorithmen, kann man entscheiden, ob zwei Schleifen in der gleichen Homotopieklasse liegen.

Algorithmus 1 entspricht Verfahren 3 und ist einfacher, da er keine Drehung benötigt.

Algorithmus 1 entspricht Verfahren 1 und, obwohl er eine Drehung der Halbgeraden enthält, er hat das Vorteil, dass man aus dem so erhaltenen Wort die Umlaufzahlen der Schleife leicht lesen kann.

1 Algorithmen

Algorithm 1: Wort, erste Version

Input:

- n verschiedene Löcher L_1, \dots, L_n
- Schleife γ als Folge z_0, z_1, \dots, z_N von benachbarten Pixeln (d.h. z_i und z_{i+1} besitzen mindestens eine gemeinsame Ecke) und $z_0 = z_N$. γ darf nicht durch ein Loch gehen.

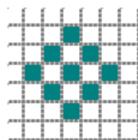
Output: Element aus F_n , das zur Homotopieklasse von γ in $\pi_1(X, x_0)$ isomorph ist. Die Isomorphie hängt ausschliesslich von der Konfiguration der Löcher ab.

```
1 Sortiere die Löcher von oben nach unten nach der y-Koordinate;
2 Erhalte  $(L_{i_1}, L_{i_2}, \dots, L_{i_n})$ ;
   /* Wir arbeiten mit dem Koordinatensystem, das  $L_1$  als
   Ursprung und nach oben gerichtete y-Achse hat. */
3  $x(L_{i_k}) :=$  x-Koordinate von  $L_{i_k}$ ;
4  $y(L_{i_k}) :=$  y-Koordinate von  $L_{i_k}$ ;
5  $x(z_j) :=$  x-Koordinate von  $z_j$ ;
6  $y(z_j) :=$  y-Koordinate von  $z_j$ ;
   /* Suche die Durchschnitte von  $\gamma$  mit  $s_1, \dots, s_n$  und schreibe
   die entsprechende Buchstaben */
7 bool StreckeGefunden = false;
8 for  $(j = 1, \dots, N)$  do
9   for  $k = 1, \dots, n$  and  $(\neg \text{StreckeGefunden})$  do
10    if  $x(L_{i_k}) = x(z_j)$  and  $y(z_j) > y(L_{i_k})$  then
11      Schreibe Buchstabe( $z_{m-1}, z_m, z_{m+1}, s_{i_k}$ );
12      StreckeGefunden = true;
13    end
14  end
15 end
```

Bem 1. Das in Teil 3 eingeführte Verfahren 1 hat bei der Implementierung das folgende Problem:

Löcher werden durch ein Pixel dargestellt und bei der Drehung hat man wegen Abschätzung nur endlich viele Winkeln zur Verfügung.

Abbildung 11: Konfiguration von Löcher, für die wegen Abschätzung keine optimale Drehung existiert



Wenn es mehrere Löcher gibt die zu nahe beieinander liegen, kann es passieren, dass keine Drehung existiert, so dass jede Folge von benachbarten Pixel, die eine Halbgeraden darstellt, genau ein gedrehtes Loch (durch einen Pixel dargestellt) enthält.

Um dieses Problem zu vermeiden, genügt es die n Löcher so zu wählen, dass zwischen je zwei Löcher ein Abstand von mindestens n Pixel ist.

Algorithm 2: Wort, zweite Version

Input:

- n verschiedene Löcher L_1, \dots, L_n , die die Bedingung aus Bemerkung 1 erfüllen.
- Schleife γ als Folge z_0, z_1, \dots, z_N von benachbarten Pixeln (d.h. z_i und z_{i+1} besitzen mindestens eine gemeinsame Ecke) und $z_0 = z_N$. γ darf nicht durch ein Loch gehen.

Output: Element aus F_n , das zur Homotopieklasse von γ in $\pi_1(X, x_0)$ isomorph ist. Die Isomorphie hängt nur von der Konfiguration der Halbgeraden ab.

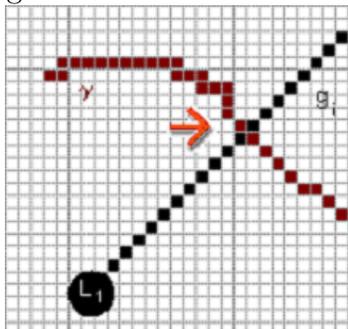
```
1 Sortiere die Löcher von oben nach unten nach der y-Koordinate;
2 Erhalte  $(L_{i_1}, L_{i_2}, \dots, L_{i_n})$ ;
   /* 1. Konstruktion von  $g_1, \dots, g_n$  */
3  $\vec{r}_1 := (0, 1)$ ;
4  $\forall i = 1, \dots, n$  define  $g_i := L_i + t \cdot \vec{r}_1, t \in \mathbb{R}, t \geq 0$ ;
   /* 2. Falls nötig Schleife und Löcher drehen */
5 Wende Algorithmus "Drehung( $\gamma, L_1, \dots, L_n$ )" an;
   /* 3. Fixiere das Koordinatensystem, s.d. die Ursprung mit
       $L_1$  und die y-Achse nach oben gerichtet ist. Dann suche
      die Durchschnitte von  $\gamma$  mit  $g_1, \dots, g_n$  */
6  $x(L_i) :=$  x-Koordinate von  $L_i$ ;
7  $y(L_i) :=$  y-Koordinate von  $L_i$ ;
8  $x(z_k) :=$  x-Koordinate von  $z_k$ ;
9  $y(z_k) :=$  y-Koordinate von  $z_k$ ;
10  $z_{N+1} := z_1$ ;
11 for  $m = 1, \dots, N$  do
12   for  $s = 1, \dots, n$  do
13     if  $(y(z_m) > y(L_s))$  and  $(x(z_m) = x(L_s))$  then
14       /* 4. Wort schreiben (durch Anwendung von
15         Algorithmus " $Buchstabe(z_{k-1}; z_k; z_{k+1}; g_i)$ ") */
16       Schreibe  $Buchstabe(z_{m-1}, z_m, z_{m+1}, g_s)$ ;
17     end
18   end
19 end
```

Um eine Konfiguration von n parallelen Halbgeraden zu konstruieren, kann

man die Löcher und die Schleife festhalten, und die Halbgeraden bis zur optimalen Konfiguration drehen lassen. Dieses Verfahren führt aber zum folgenden Problem:

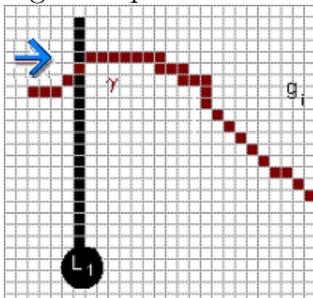
Weil die Schleife sowie die gedrehte Halbgeraden als Folge von Pixeln gespeichert sind, kann die folgende Situation erscheinen (siehe Abbildung 12):

Abbildung 12: Problematischer Durchschnitt



Eine mögliche Lösung zu diesem Problem ist die Halbgeraden fix (z.B. vertikal) zu halten und, falls nötig, Schleife und Löcher um L_1 zu drehen. In diesem Fall besitzt jede Schleife, die die Halbgerade überquert, mindestens einen mit der Halbgeraden gemeinsamen Pixel (siehe Abbildung 13).

Abbildung 13: Optimales Durchschnitt



Algorithm 3: *Drehung*(γ, L_1, \dots, L_n)

Input:

- n verschiedene Löcher L_1, \dots, L_n
- Schleife γ um ein gegebenes x_0
- Halbgeraden g_1, \dots, g_n

Output: Gedrehte Löcher und Schleife, s.d. jede Halbgerade genau ein Loch enthält.

```
1  $D(\gamma, L_i, \dots, L_n)$ ;  
2  $\alpha := 1 \text{ rad}$ ;  
3 for  $i = 1, \dots, n$  do  
4   for  $j = 1, \dots, n$  do  
5     if  $j \neq i$  then  
6       if  $L_i \in g_j$  then  
7          $\gamma =$ Drehung mit Winkel  $\alpha$  von  $\gamma$  um  $L_1$ ;  
8          $\forall k \in \{1, \dots, n\}$ ;  
9          $L_k =$ Drehung mit Winkel  $\alpha$  von  $L_k$  um  $L_1$ ;  
10         $\vec{r} := (0, 1)$ ;  
11         $g_k := L_k + t \cdot \vec{r}, t \in \mathbb{R}, t \geq 0$ ;  
12         $D(\gamma, L_1, \dots, L_n)$ ;  
13        return;  
14      end  
15    end  
16  end  
17 end
```

Algorithm 4: *Buchstaben*(z_k, g_i)

Input: $z_{k-1}, z_k, z_{k+1}, g_i, L_i$ wobei z_k auf g_i liegt

Output: Buchstaben:

- a_i , wenn es in z_k eine Überquerung im Gegenuhr Sinn gibt
- a_i^{-1} , wenn es in z_k eine Überquerung im Uhr Sinn gibt
- Leeres Wort, wenn es in z_k keine Überquerung gibt.

/* Berechnung von B: */

```
1 if  $y(z_{k-1}) > y(L_i)$  then
2 |   Schreibe  $a_i^{-1}$ ;
3 end
4 if  $y(z_{k+1}) > y(L_i)$  then
5 |   Schreibe  $a_i$ ;
6 end
```

Man wendet zur Vereinfachung die Kürzungsregeln an, die aus den Relationen $a_i a_i^{-1} = \text{Leeres Wort} = a_i^{-1} a_i$ folgen.

Algorithm 5: Vereinfachung

Input:

- $Wort(\gamma)$ mit γ Schleife um x_0
- **int** $b \in \{0, \dots, W.size()\}$

Output: Vereinfachung von $Wort(\gamma)$.

```

/* Wir bezeichnen  $a^{-1}$  mit  $-a$  und arbeiten mit dem String
    $W$  (das so bezeichnete  $Wort(\gamma)$ ) */
1 Vereinfachung(string  $W$ , int  $b$ )
2 if ( $W = \text{Leeres Wort}$  or ( $W.size() < 3$ ) ) then
3   | return  $W$ ;
4 end
5 for ( $i := b$ ;  $i < W.size()$ ;  $++i$ ) do
6   | if ( $W[i] = '-'$ ) then
7     | if ( $i < (W.size() - 2)$  and  $W[i + 1] = W[i + 2]$ ) then
8       | /* Kürze Teilwort vom Typ  $'-aa'$  Weg */
9       | VereinfachtesWort =  $W$  ohne  $W[i], W[i + 1], W[i + 2]$ ;
10      | return Vereinfachung(VereinfachtesWort,  $max(0, i - 2)$ );
11     | end
12     | if ( $i \neq 0$  and  $W[i - 1] = W[i + 1]$ ) then
13       | /* Kürze Teilwort vom Typ  $'a - a'$  Weg, wenn das
14         | Symbol vor dem ersten Buchstabe kein  $'-'$  ist
15         | */
16       | if ( $(i < 2)$  or ( $i > 1$  and  $W[i - 2] \neq '-'$ )) then
17         | VereinfachtesWort =  $W$  ohne  $W[i - 1], W[i], W[i + 1]$ ;
18         | return Vereinfachung(VereinfachtesWort,  $max(0, i - 2)$ );
19       | end
20     | end
21   | end
22 end
23 return  $W$ ;

```

2 Applets

Algorithmus 1 erlaubt nun einige Applets zu konstruieren, die verschiedene Probleme behandeln.

2.1 Applet 1: Gehören zwei Schleifen zur gleichen Homotopieklasse?

Das erste Applet, welches Algorithmus 1 benützt, ist das folgende:
Der Benutzer hat die Möglichkeit aus der gegebenen Fläche n Punkte (L_1, \dots, L_n) herauszunehmen, und zwei Schleifen γ und δ mit Anfangspunkt x_0 im neuen Raum mit der Maus zu zeichnen.

Alternativ kann der Benutzer nur über die Anzahl von Löchern entscheiden. Das kann eine Benutzung des Programms, die zu Schätzungsfehlern oder zu langen Wartezeiten führen kann, vermeiden (zum Beispiel wenn zu viele oder zu nahe Löcher gewählt werden).

Das Output-Fenster zeigt die Antwort zur Frage: *Sind γ und δ homotop?* (ja oder nein) und die von Algorithmus 1 konstruierte Halbgeraden, zeichnet nochmals die Schleifen und jedes Mal, dass eine Schleife eine Halbgerade g_i trifft, wird der entsprechende Buchstabe in dem entsprechenden Wort geschrieben.

Die folgende Aufgaben illustrieren einerseits die Vor- und Nachteile, wenn eine Frage wie 'Sind γ und δ homotop?' einem Computer gestellt wird, der keine Intuition besitzt aber der Sätze und Algorithmen schnell anwenden kann und helfen andererseits den Begriff von Homotopie veranschaulicht zu verstärken.

1. Zeichnen Sie zwei Schleifen γ und δ , die homotop, und zwei, die nicht homotop sind. Was stellen die Buchstaben a_i und a_i^{-1} dar?
2. Zeichnen Sie zwei Schleifen γ und δ , so dass $\text{Wort}(\gamma)$ und $\text{Wort}(\delta)$ die gleichen Buchstaben gleich viele Male enthalten (z.B. $aac^{-1}bb$ und $abc^{-1}ab$), aber γ und δ nicht homotop sind.
3. Zeichnen Sie eine Schleife, die zur schon gegebenen Schleife σ homotop ist.

Bei der letzten Frage sollte eine komplizierte Schleife σ gewählt werden. Für einige topologische Eigenschaften könnte man gerne denken, dass die Intuition schon genügt, und eine mathematisch strenge Theorie mit formelle Sätze und Beweise gar nicht benötigt wird.

Der Benutzer soll durch diesen Fragen sehen, dass auch für einen so intuitiven Begriff wie Homotopie in einem ganz einfachen Raum mit der natürlichen Topologie versehen, klar gegliederte Sätze nützlich sind, und dass die Intuition Grenzen hat.

Algorithm 6: *Homotopietester*

Input:

- $Wort(\gamma)$ und $Wort(\delta)$, wobei γ und δ Schleifen um x_0 sind
- Löcher L_1, \dots, L_n

Output: Antwort zur Frage: ' $\gamma \simeq \delta$ '

- 1 Wende Algorithmus "Vereinfachung" auf $Wort(\gamma)$ und $Wort(\delta)$ an;
 - 2 **if** (*vereinfachtes* $Wort(\gamma)$) = (*vereinfachtes* $Wort(\delta)$) **then**
 - 3 | return: true (γ und δ gehören zur gleichen Homotopieklasse)
 - 4 **else**
 - 5 | return: false (γ und δ gehören nicht zur gleichen Homotopieklasse)
-

2.2 Applet 2: Nullomotopie vs Nullhomologie

Dieses Applet soll über die Nichtkommutativität der Fundamentalgruppe aufmerksam machen.

Es erlaubt, die Fundamentalgruppe mit einer kommutativen Gruppe zu vergleichen. Konkret werden in diesem Beispiel Nullhomologie und nullhomotopie verglichen.

Der Grundraum ist schon gegeben und enthält 2 Löcher L_1, L_2 . Den Benutzer fragt man einige Schleifen zu zeichnen, die gewisse Eigenschaften besitzen.

Die Umlaufzahl um L_i für $i = 1, 2$ kann man dann wie folgt berechnen:

Sei für $k = 1, \dots, n$ $Y_k := \mathbb{C} \setminus L_k$. Die Fundamentalgruppe $\pi_1(Y_k, x_0)$ ist für jedes k zu \mathbb{Z} isomorph und darum abelsch. In diesem Fall sind Homotopie- und Homologiegruppe isomorph, was uns erlaubt, den Algorithmus „Wort“ zu benutzen.

Man kann also die Umlaufzahl einer Schleife δ um jedes Loch wie folgt berechnen:

- Möglichkeit 1: $Wort(\delta, L_i)$ für jedes i berechnen, dann $n(\gamma, L_i) = \#(a_i)$.
- Möglichkeit 2: Falls man $Wort(\delta, L_1, \dots, L_n)$ kennt, kann man den folgenden Algorithmus anwenden:

Algorithm 7: Umlaufzahl

Input: $Wort(\delta)$ mit δ in $\mathbb{C} \setminus \{L_1, \dots, L_n\}$ um x_0

Output: Umlaufzahlen $n(L_i, \delta)$ für $i = 1, \dots, n$

```
1 for  $i = 1, \dots, n$  do  
2   |  $n(L_i, \delta) := \#(a_i)$  in  $W - \#(a_i^{-1})$  in  $W$ ;  
3 end
```

Einige mögliche Fragen:

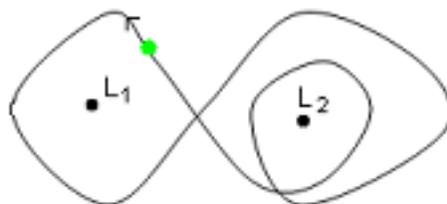
1. Welche Eigenschaften einer Schleife kann man aus der Umlaufzahl sehen? Welche zusätzlichen Informationen über die Schleife kann man aus seiner Homotopieklasse finden?
2. Zeichnen Sie eine Schleife γ mit $n(\gamma, L_1) = k$ und $n(\gamma, L_2) = (k, m \in \{-3, -2, \dots, 2, 3\})$. Was ist die entsprechende Homotopieklasse?
3. Zeichnen Sie eine Schleife, die nullhomolog, aber nicht nullhomotop ist.

- Setzen Sie mit der gegebenen Schleife β eine Weitere Schleife zusammen, so dass die resultierende Schleife als Wort ein Element aus der Kommutatorgruppe $K(F_2) := \{ghg^{-1}h^{-1} | g, h \in F_2\}$ hat. Was erhält man als Windungszahl?

Mögliche Antworten und Bemerkungen:

- Die Umlaufzahl sagt aus, wie oft eine Schleife um einem Punkt läuft, aber nicht in welcher Reihenfolge. Diese Information kann man aus der Homotopieklasse gewinnen.
- Siehe Abbildung 14

Abbildung 14: Schleife mit Windungszahlen 1, bzw. -2

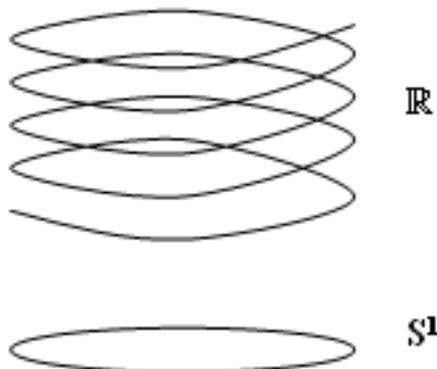


- Siehe Abbildung 2 (Lösung für $k = 1, m = -2$)
- Die so vervollständigte Schleife wird nullhomolog sein (siehe Kap. I).

2.3 Projekt für Applet 3: Universelle Überlagerung und Fundamentalgruppe

Eine weitere Möglichkeit, Algorithmus 1 anzuwenden, ist um sich die Beziehung zwischen der universellen Überlagerung und Fundamentalgruppe zu veranschaulichen.

Abbildung 15: \mathbb{R} als universelle Überlagerung von S^1



Man betrachtet als Grundraum $\mathbb{C} \setminus \{0\}$, der homotopieäquivalent zu S^1 ist, und \mathbb{R} als Universelle Überlagerung von S^1 versehen. Als Funktion betrachtet man:

$$\begin{aligned} \exp : \mathbb{R} &\rightarrow S^1 \\ t &\mapsto e^{it} \end{aligned}$$

Das Input-Fenster zeigt den Grundraum und gibt die Möglichkeit, eine Schleife α zu zeichnen. Als Output bekommt man die Hochhebung $\tilde{\alpha}$ von α zu \mathbb{R} und ihre Umlaufzahl.

Dieses Applet bringt ein konkretes Beispiel von Berechnung der Fundamentalgruppe eines Raumes mittels seiner universellen Überlagerung.

Der Endpunkt, des zu \mathbb{R} hochgehobenes Weges, kann durch eine kleine Änderung von Algorithmus 3 berechnet werden und zwar, durch Multiplikation der erhaltenen Umlaufzahl mit 2π .

3 Input-Probleme und Lösungsvorschläge

Die oben beschriebenen Algorithmen verlangen, dass die Schleife als eine Folge von benachbarten Pixeln gespeichert wird. Es kann aber passieren (siehe unten), dass die Eingaben vom Benutzer zwei sukzessive weit entfernte Pixel enthält.

Das Ziel dieses Abschnitts ist, solche unvollständige Eingaben sinnvoll zu vervollständigen, so dass das Programm mit der so vervollständigten Eingaben arbeiten kann.

Problem 1. *Die eingegebene Scheife geht durch ein Loch.*

Lösungsvorschlag 1. Man kann einfach eine Fehlermeldung ausgeben: Löcher sind ein wichtiger Bestandteil des Grundraumes und es ist aus didaktischen Gründen wichtig, auf dem Unterschied gegenüber des Raumes \mathbb{C} hinzuweisen.

Problem 2. *In manchen Applets fragt man den Benutzer, eine Schleife mit der Maus zu zeichnen. Da es schwierig ist, die Kurve am selben pixel zu beenden wo man sie begonnen hat, kann deswegen eine offene Kurve als Eingabe entstehen. Man kommt zum gleichen Problem, wenn die Schleife zu schnell gezeichnet wird: die gespeicherte Pixel sind oft nicht benachbart.*

Lösungsvorschlag 2. Man kann die Eingabe durch Strecken vervollständigen, indem man zwischen je zwei Pixel z_i und z_{i+1} eine Folge von benachbarte Pixel einfügt, die die Strecke $\overline{z_i z_{i+1}}$ annähern.

Algorithm 8: Vervollständigung

Input: Pixel A und B **Output:** Annäherung der Strecke \overline{AB} durch eine Folge von benachbarte Pixel

```
1  $MIN := B$  ;
2  $MAX := A$  ;
3  $A.x := x$ -Koordinate von  $A$  ;
4  $A.y := y$ -Koordinate von  $A$  ;
5  $B.x := x$ -Koordinate von  $B$  ;
6  $B.y := y$ -Koordinate von  $B$  ;
7 if  $A.x = B.x$  /* Wenn  $A$  und  $B$  auf derselben vertikalen
   Gerade liegen */
8 then
9   if  $A.y < B.y$  then
10     $MIN := A$ ;
11     $MAX := B$ ;
12   for  $i = MIN.y + 1; i < MAX.y; ++ i$  do
13     $\lfloor$  Pixel  $(A.x, i)$  einfügen;
14 else
15    $m := \frac{(B.y - A.y)}{B.x - A.x}$ ;
16   if  $|A.x - B.x| > |A.y - B.y|$  then
17     if  $A.x < B.x$  then
18        $MIN := A$ ;
19        $MAX := B$ ;
20     for  $(i := MIN.x + 1; i < MAX.x; ++ i)$  do
21        $\lfloor$  Pixel  $(i, MIN.y + \lfloor (i - MIN.x)m \rfloor)$  einfügen;
22   else
23     if  $A.y < B.y$  then
24        $MIN := A; MAX := B$ ;
25     for  $(j := MIN.y + 1; j < MAX.y; ++ j)$  do
26        $\lfloor$  Pixel  $(MIN.x + \lfloor \frac{(j - MIN.y)}{m} \rfloor, j)$  einfügen;
```

Alternative Lösung

Eine Strecke ist eindeutig durch ihr Anfangs- und Endpunkt bestimmt. Darum ist es nicht nötig alle Pixel der Strecke zu speichern. Hier unten wird dieser Annäherungsversuch beschrieben.

Lösungsvorschlag 2a: Man kann die offene „Kurve“ durch eine Strecke (LineTo) vervollständigen. Der Computer speichert also den Anfangs- und Endpunkt und zeichnet auf dem Bildschirm eine Strecke dazwischen. Wichtig ist, dass die Pixel der gezeichneten Strecke aber nicht schon als Pixel der Schleife gespeichert sind. Die Strecke ist trotzdem durch die zwei gespeicherten Pixel eindeutig bestimmt.

Problem 3. Überquert die mit LineTo gezeichnete Strecke \overline{AB} die Halbgerade g_i ?

Lösungsvorschlag 3. Man benützt den folgenden

Algorithm 9: Strecke

```
/* Dieses Algorithmus bezieht sich zu einer schon
   optimalen Konfiguration, d.h mit  $n$  vertikale
   Halbgeraden, die je genau ein Loch enthalten */
```

Input:

- Anfangs- und Endpunkte A und B ($A \neq B$) der Strecke
- Loch L_i
- Halbgerade g_i

Output: Antwort zur Frage "Überquert die Strecke \overline{AB} die Halbgerade g_i ?"

```
/* Wir arbeiten im Koordinatensystem, das  $A$  als Ursprung
   und eine y-Achse, die parallel zu  $g_i$  ist, hat */
```

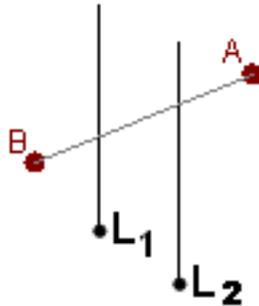
```
1  $r := \|\vec{AL}_i\|$  ;
2  $\vec{P} := \frac{\vec{AB}}{\|\vec{AB}\|} \cdot r$  ;
3 if  $y$ -Koordinate von  $L_i < y$ -Koordinate von  $\vec{P}$  then
4   | return true;
5 end
6 else
7   | return false;
8 end
```

Problem 4. *Geht die im Lösungsvorschlag 3 konstruierte Strecke durch ein Loch?*

Lösungsvorschlag 4. Das Vorgehen ist zu Algorithmus 3 ähnlich: man testet, ob y-Koordinate von L_i = y-Koordinate von \vec{P} .

Problem 5. *Eine mit dem im Lösungsvorschlag beschriebenen Verfahren konstruierte Strecke \bar{AB} schneidet mehr als eine Halbgerade (siehe Abbildung 5: man muss entscheiden, welche der zwei Halbgeraden früher überquert wird)*

Abbildung 16: L_2 wird früher überquert



Lösungsvorschlag 5.

Algorithm 10: Sortieren

/ Man arbeitet mit dem Koordinatensystem, das L_1 als Ursprung und g_1 als positive y-Achse hat. */*

Input: A, B , überquerte Halbgeraden h_1, \dots, h_k , $k < n + 1$

Output: Ordnung, in welcher h_1, \dots, h_k überquert werden.

```

1 if x-Koord. von  $A <$  x-Koord. von  $B$  then
2   | Sortiere  $h_1, \dots, h_k$  von links nach rechts (gemäss der x-Koordinate)
3 end
4 if x-Koord. von  $A >$  x-Koord. von  $B$  then
5   | Sortiere  $h_1, \dots, h_k$  von rechts nach links (gemäss der x-Koordinate)
6 end

```

/ Es macht keinen Sinn, den Fall x-Koord. von $A =$ x-Koord. von B zu betrachten, weil es dann unmöglich ist, dass \bar{AB} mehr als eine Halbgerade schneidet. */*

Teil V

Anlage und Kontakte

Diese Bachelor-Arbeit wurde vom Projekt LEMUREN unterstützt. Sie bildet eine Grundlage zur Entwicklung neuer E-Learning Ressourcen im Rahmen des Projekts LEMUREN.

Weitere Informationen: <http://www.lemuren.math.ethz.ch>

Kontakt: lemuren@math.ethz.ch.”

Die Algorithmen, die in dieser Arbeit beschrieben werden, sind in einem C++ Windows-Programm von Julien Carron implementiert (Bemerkungen an: carron_julien@hotmail.com) (siehe Anlage: ***Homotopy detector***).

Literatur

- [SKP] Prof. Pink, R., *Topologie, Vorlesungs-Skript*, ETHZ, Sommersemester 2006.
- [J] Jänich, N., *Topologie*, Springer, Deutschland, 2005, 8.Auflage.
- [Hatcher] Hatcher, A., *Algebraic Topology*, Cambridge University, 2002.
- [FB] Freitag, E., Busam, R. *Funktionentheorie 1*, Springer, Berlin, 2006, Vierte Auflage.