

**Geometrie- und topologiebasierte Algorithmen zum
Vergleich spektroskopischer Daten in der
astronomischen Datenanalyse**

Diplomarbeit

Vorgelegt am 23. Juni 2014

Name: Mirko Westermeier
Studiengang: Diplom-Informatik

Gutachter: Prof. Dr. Jan Vahrenhold
Prof. Dr. Klaus Hinrichs

Institut für Informatik
Westfälische Wilhelms-Universität Münster

Inhaltsverzeichnis

Einleitung	3
1 Maschinelles Lernen in der astronomischen Datenanalyse	5
1.1 Überblick über eine Werkzeugkette	6
1.2 Methoden der maschinellen Klassifikation	8
1.2.1 Lokale Demokratie: die k nächsten Nachbarn	9
1.2.2 Eine Grenze ziehen: Support Vector Machines	10
1.3 Gütebeurteilung für die Klassifizierung	14
2 Die Fréchet-Distanz als Abstandsbegriff polygonaler Kurven	18
2.1 Einführung und Abgrenzung	18
2.2 Berechnung von Fréchet-Distanzen polygonaler Kurven	19
2.2.1 Das Freespace-Diagramm	20
2.2.2 Der Freespace als Schlauch	22
2.2.3 Der Freespace als Ellipse	27
2.2.4 Das Fréchet-Distanz-Entscheidungsproblem	34
2.2.5 Exakte Berechnung der Fréchet-Distanz	42
2.3 Die partielle Fréchet-Distanz	45
2.3.1 Qualitätsmaße für die partielle Fréchet-Distanz	46
2.3.2 Das deformierte Freespace-Diagramm	47
2.3.3 Ein Approximationsalgorithmus	48
3 Eine lokalisierte Version der Fréchet-Distanz	51
3.1 Formalisierung	52
3.2 Konkrete Berechnungen	55
3.2.1 Lösung des lokalisierten Entscheidungsproblems	57
3.2.2 Ermittlung kritischer Werte	59
3.3 Die Lokalisierung der Leine	63
3.3.1 Lokalisierung über die punktweise Abweichung	63
3.3.2 Ein divide-and-conquer-Ansatz zur Lokalisierung auf Basis der Fréchet-Distanz	64

4 Eine Implementierung von Fréchet-Distanz-Varianten für die astronomische Datenanalyse	70
4.1 Clastro: ein Framework zur Klassifikation astronomischer Objekte aus dem SDSS	71
4.1.1 Repräsentation astronomischer Objekte	71
4.1.2 Datentransport in Clastro	72
4.1.3 Training und Klassifikation	72
4.1.4 Die Benutzerschnittstelle	73
4.2 Implementierung von Fréchet-Distanz-Varianten in Clastro	74
4.2.1 Geometrische Grundlagen	74
4.2.2 Entscheidungsproblem und lokalisierte Fréchet-Distanz	77
4.2.3 Eine grafische Benutzerschnittstelle	80
5 Reflexion und Ausblick	82
Literaturverzeichnis	83

Symbolverzeichnis

Hier werden einige in dieser Ausarbeitung benutzte mathematische Notationen erklärt, die nicht in jedem Kontext als eindeutig definiert vorrausgesetzt werden können:

$\langle \cdot, \cdot \rangle$	Standard-Skalarprodukt (hier meist des \mathbb{R}^2)
E_2	Einheits-(2×2)-Matrix $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ (induziert die Abbildung $\text{id}_{\mathbb{R}^2}$)
I	Das Einheitsintervall $[0, 1]$
$\bigcup_{a,b} X(a, b)$	Verkürzte Schreibweise für $\bigcup_{a \in A, b \in B} X(a, b)$, falls aus dem Kontext A und B klar hervorgehen. Statt \bigcup auch für \bigcap , \sum , \prod , min und max
π_k	Die Projektion $\prod_{i=1}^n X_i \rightarrow X_k$, $(x_1, \dots, x_n) \mapsto x_k$ eines (endlichen) kartesischen Produkts auf eine Koordinatenachse mit $k \in \{1, \dots, n\}$.
$\mathfrak{P}(X)$	Potenzmenge der Menge X , also die Menge aller Teilmengen $Y \subset X$.

Einleitung

Um mit der technischen Entwicklung in der heutigen Survey-Astronomie und dem damit verbundenen gigantischen Datenaufkommen umgehen zu können, ist der Einsatz automatisierter Verfahren zur Verarbeitung und Reduktion spektroskopischer Daten unverzichtbar. Durch den Einsatz vielfältiger Methoden aus dem maschinellen Lernen und der algorithmischen Geometrie haben sich die verwendeten Verfahren zu einem spannenden interdisziplinären Feld für die Zusammenarbeit von Astrophysikern und Informatikern entwickelt.

Ein wichtiges Ziel dieser Techniken ist es, die Klassifikation eines Objekts nur anhand seines Spektrums anhand bestimmter Merkmale durch Automatisierung so zu beschleunigen, dass die Daten überhaupt in zumutbarer Zeit verarbeitet werden können. Anstatt dabei einem erfahrenen Astrophysiker die Interpretation jedes Spektrums zuzumuten, werden Techniken entwickelt, innerhalb kürzester Zeit sehr gute Schätzungen über die Zugehörigkeit der Objekte bestimmten zu Klassen abzugeben.

In dieser Ausarbeitung werden Methoden vorgestellt, eine solche automatisierte Klassifizierung zu realisieren. Grundlagen des maschinellen Lernens anhand der Anordnung von Trainingsmengen von Spektren in einem Merkmalsraum werden in Kapitel 1 vorgestellt. Die Auswahl geeigneter Merkmale für die spezifische astronomische Fragestellung ist dabei für den Klassifizierungserfolg ein zentrales Problem. Die in dieser Arbeit diskutierten Methoden lassen sich ebenfalls als Merkmal zur Klassifikation einsetzen.

Das Spektrum eines astronomischen Objekts lässt sich durch die in der Praxis gegebene Diskretisierung als spezielle polygonale Kurve auffassen. Ein populäres Maß für die Ähnlichkeit polygonaler Kurven ist die *Fréchet-Distanz*, die man sich sinnbildlich als die kürzestmögliche Leine vorstellen kann, die ein Hund und ein Herrchen, die jeweils auf ihrer eigenen polygonalen Kurve vorwärts gehen, benötigen, um gemeinsam das Ende ihrer Wege zu erreichen. Dieser Distanzbegriff wird in Kapitel 2 ausführlich diskutiert. Da in der Literatur nur wenig spezifische Angaben darüber zu finden sind, welcher Gestalt das zur Berechnung der Fréchet-Distanz nötige Freespace-Diagramm genau ist, werden in einem Teil dieses Kapitels Algorithmen zur exakten Berechnung des gesamten Freespace-Diagramms vorgestellt.

Varianten der Fréchet-Distanz, die sich für die Verarbeitung spektroskopischer Daten besonders eignen, sind Gegenstand von Abschnitt 2.3 und Kapitel 3: Während die partielle Fréchet-Distanz sich besonders für Daten mit Ausreißern eignet, ist die lokalisierte Fréchet-Distanz imstande, bei der Messung von Ähnlichkeiten zwischen Spektren in einigen Bereichen besonders sensibel zu sein und in anderen größere Abweichungen zuzulassen. Die Definition einer solchen Variante ist motiviert durch eine tatsächlich realisierte Klassifikationsaufgabe sternbildender Galaxien: Die zum Training eingesetzten Beispielgalaxien haben die Eigenschaft, dass sich ihre Spektren in gewissen Wellenlängenbereichen stark ähneln, in anderen Bereichen aber stark unterschiedlich aussehen.

Im Rahmen meiner Diplomarbeit wurde ein Teil der vorgestellten Algorithmen in Clastro, einem Framework zur Klassifikation astronomischer Objekte, implementiert. Es geht dabei darum, zu einer spezifischen Klasse von astronomischen Objekten ein repräsentatives Spektrum zu finden und die Distanz der Spektren von bisher unklassifizierten Objekten als Merkmal für die automatisierte Klassifikation nutzbar zu machen. In Kapitel 4 wird die Architektur und Arbeitsweise des Frameworks kurz vorgestellt, bevor auf die Implementierungsdetails eingegangen wird.

Ich bedanke mich herzlich bei allen Personen, die mich bei der Implementierung und bei der schriftlichen Ausarbeitung unterstützt haben. Prof. Dr. Vahrenhold gilt mein Dank für seine intensive Unterstützung im Rahmen der Betreuung dieser Diplomarbeit und für inspirierende Ideeninjektionen. Christian Scheffer danke ich dafür, dass er jederzeit bereit war, meine geometrischen Ideen mit mir zu diskutieren und für seine Fähigkeit, Gedanken präzise zu formulieren. Außerdem danke ich Andreas Thom für die konstruktiven Rückmeldungen im Implementierungsteil und für seine offenherzige Motivation. Und ohne die Unterstützung meiner Eltern Gabi Hillmann und Michael Westermeier wäre ich völlig aufgeschmissen.

Mirko Westermeier
Juni 2014

Kapitel 1

Maschinelles Lernen in der astronomischen Datenanalyse

Wenn man einem erfahrenen Astronomen das Spektrum¹ eines Himmelsobjekts zeigt, kann dieser daraus augenblicklich wertvolle Informationen zur Art und Beschaffenheit dieses Objekts erkennen. Obwohl diese Spektren teilweise sehr unterschiedlich aussehen, kann der Astronom doch anhand wesentlicher Merkmale, die fast allen Objekten einer Art gemein sind, eine Klassifikation vornehmen.

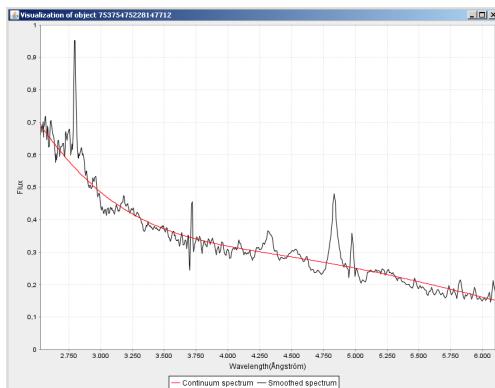


Abbildung 1.1: Das Spektrum (im Bild schwarz) eines beispielhaften Quasars. Ein intuitiv erfassbares Merkmal ist die über den Wellenlängenbereich abfallende Intensität der Strahlung. Weitere charakteristische Merkmale sind die ausgeprägten breiten Peaks. Das rote Kontinuum ist eine um Emissionen und Absorptionen bereinigte Version des Spektrums. Die nach rechts hin abfallende Intensität lässt sich direkt als die Intensität am Beginn und am Ende des Kontinuums ablesen.

Die Fähigkeit, wesentliche Merkmale von unwesentlichen zu unterscheiden, beruht auf langer Erfahrung und guter Kenntnis der astrophysikalischen Vorgänge, die zu bestimmten Formen in einem Spektrum führen. Die Breite mancher Peaks im

¹ Ein elektromagnetisches Spektrum wird in diesem Kontext betrachtet als der Graph einer (diskreten) Funktion, die Wellenlängen aus einem bestimmten Bereich die Intensität der zugehörigen elektromagnetischen Strahlung zuordnet.

Spektrum von Quasaren etwa hängt unmittelbar mit der Rotationsgeschwindigkeit ihrer Akkretionsscheibe zusammen.

Obwohl ein geübter Mensch dabei in kurzer Zeit zu guten Klassifikationsergebnissen kommen kann, macht doch das Geschwindigkeitspotential heutiger Computer gemeinsam mit Ergebnissen im maschinellen Lernen in der Astroinformatik (z. B. [GPT⁺¹⁰]) berechtigte Hoffnung, bei geschickter Vorgehensweise noch deutlich schneller und vor allem automatisiert zu guten Ergebnissen kommen zu können.

Die Notwendigkeit einer solchen automatisierten Klassifikation von Spektren ergibt sich schon aus der immensen und ständig weiter wachsenden Datenmenge² in der heutigen Survey-Astronomie und dem Wunsch, interessante Daten möglichst schnell von uninteressanten unterscheiden zu können.

Das Standardwerkzeug für die automatische Klassifizierung ist das überwachte maschinelle Lernen, das im Data Mining eine große Rolle spielt. Dabei wird u. A. versucht, Vorhersagen für eine Klassifizierung eines unbekannten Objekts zu treffen auf der Basis von Klassifizierungen bereits bekannter Objekte innerhalb der sogenannten Trainingsmenge. Dies geschieht anhand bestimmter Merkmale, die man aus den bekannten und unbekannten Objekten extrahieren kann und die benutzt werden, um einen Klassifizierer zu trainieren (vgl. [HTF11, Introduction]). Dabei ist die Wahl der Merkmale, anhand derer sich allein oder in Kombination Objekte der Trainingsmenge aus verschiedenen Klassen besonders gut unterscheiden lassen, entscheidend für eine gute Klassifizierung.

In diesem Kapitel wird eine Werkzeugkette vorgestellt, die auf der Grundlage von Daten aus dem Sloan Digital Sky Survey gemeinsam mit astronomischem Expertenwissen eine automatisierte Klassifizierung „neuer“ Spektren unter Verwendung der oben beschriebenen Methoden des überwachten maschinellen Lernens ermöglicht. Um einen guten Überblick über die Arbeitsweise zu geben, werden die dazu verwendeten Techniken ebenfalls vorgestellt, ohne dabei zu sehr in die Tiefe zu gehen. In Kapitel 4 wird auf konkrete Implementierungsdetails des dazu verwendeten Frameworks eingegangen.

1.1 Überblick über eine Werkzeugkette

Spektren aus dem SDSS liegen häufig in einem in der Astronomie verbreiteten Datenformat für multidimensionale Datenmengen vor, in sogenannten FITS-Dateien³. Diese Dateien umfassen meist die Datensätze für mehrere Himmelsobjekte, die gleichzeitig in einem Spektrographen aufgenommen wurden⁴. Da diese Gruppen von Objekten im Allgemeinen nicht nach den für uns relevanten Klassen angeordnet sind, ist ein solcher Dateisystemzugriff sehr unhandlich. Die Daten können aber automatisiert in ein für die hier relevanten Anwendungszwecke geeignetes Datenbanksystem überführt werden.

Im Wesentlichen besteht ein Datensatz, der ein Himmelsobjekt der hier verwendeten Datenbank repräsentiert, aus einer eindeutigen bereits im SDSS vergebenen Iden-

²Beim Sloan Digital Sky Survey (SDSS) etwa hatte das Data Release 10 aus dem Jahr 2013 ein Gesamtvolume von ca. 70TB (sdss3.org/dr10/data_access/volume.php) wohingegen das sechs Jahre ältere Data Release 6 noch ein Gesamtvolume von unter 17TB hatte (sdss2.org/dr6). Es handelt sich dabei um eine breit angelegte Durchmusterung von Objekten aus einem Viertel des Himmels durch Aufnahme der Intensitäten bei fünf Wellenlängenbereichen (Photometrie) und eventuell anschließende detaillierte Spektroskopie.

³Akronym für: *Flexible Image Transport System*, 1981 von der NASA entwickelt. Details: fits.gsfc.nasa.gov

⁴Der BOSS-Spektrograph etwa nimmt in einem Arbeitsschritt die Spektren von 1000 Himmelsobjekten auf. Details: www.sdss3.org/instruments/boss_spectrograph.php

tifikationsnummer, einigen Metadaten etwa zur Position am Himmel und zur Rotverschiebung sowie dem Spektrum selbst, welches aufgrund seiner variablen Länge im CSV-Format vorliegt. Außerdem lässt sich mit Hilfe zweier für jedes Spektrum eingetragener Koeffizienten die genaue Wellenlänge für jede Stelle im Spektrum ermitteln.

Der Import aus der Datenbank erzeugt Himmelsobjekte repräsentierende Objekte im Speicher, deren Spektrum einfach und effizient handhabbar ist. So lässt sich etwa über alle Werte des Spektrums iterieren, an jeder Stelle lässt sich die Wellenlänge unter Berücksichtigung oder Nichtberücksichtigung der Rotverschiebung ermitteln und umgekehrt ist das Auslesen der Intensität zu einer gegebenen Wellenlänge in konstanter Zeit möglich.

Aufgrund der dankenswerten Vorarbeit eines Astrophysik-Experten ist eine Menge von Spektren bekannt, deren einzelne Exemplare für eine spezielle Klasse von Himmelsobjekten als beispielhaft gelten können. Ebenso existiert eine Menge von Spektren, die davon mehr oder weniger stark abweichen und von diesem Experten klar als nicht zugehörig zu der speziellen Klasse von Himmelsobjekten eingestuft wurden. Gemeinsam dient diese Menge von Spektren als Trainingsmenge.

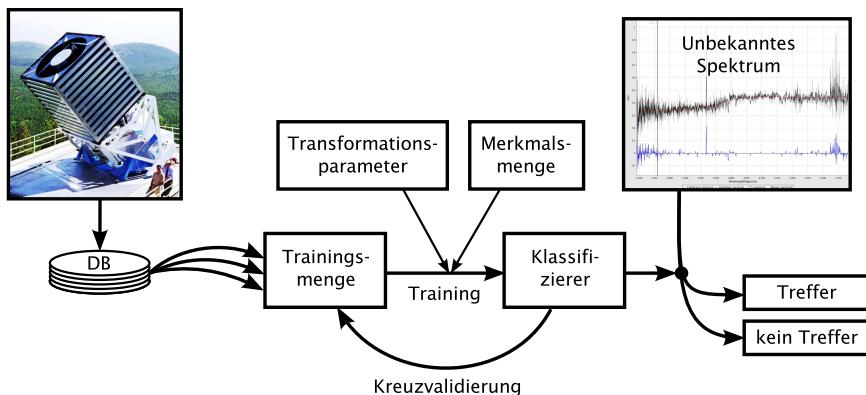


Abbildung 1.2: Ein schematischer Überblick über die Komponenten der hier beschriebenen Werkzeugkette. Die FITS-Dateien aus dem SDSS werden konvertiert in einem lokalen Datenbanksystem verwaltet, ein Teil dieser Spektren wurde von einem Astrophysiker als Trainingsmenge klassifiziert. Gemeinsam mit Parametern und einer geeigneten Merkmalsmenge kann daraus ein Klassifizierer trainiert werden, der ein unbekanntes Spektrum als zugehörig zu der gerade relevanten Klasse von Himmelsobjekten auszeichnet oder nicht. Mit Hilfe der Kreuzvalidierung kann die Güte des Klassifizierers gegenüber Teilmengen seiner eigenen Trainingsmenge gemessen werden.

Die Auswahl einer möglichst charakteristischen Menge von Merkmalen, anhand derer sich die Objekte der Trainingsmenge besonders gut unterscheiden, wird auf der Basis von Expertenwissen erarbeitet. Wie genau aufgrund dieser Merkmale die Klassifizierung eines neuen Objekts geschieht, wird in den folgenden Abschnitten diskutiert.

Aufgrund des aufwendigen Entstehungsprozesses einer guten Trainingsmenge für bestimmte Klassen von Himmelsobjekten beschränken sich in dieser Werkzeugkette die Variationsmöglichkeiten für Klassifizierer auf die Auswahl von Merkmalen und die Parameter gewisser Simplifizierungsmethoden, die die Spektren vor Extraktion der Merkmale durchlaufen. In der Werkzeugkette existieren Verfahren, um verschie-

dene Klassifizierer anhand einer Kreuzvalidierung innerhalb der Trainingsmenge zu bewerten sowie paarweise zu vergleichen, indem Teilmengen der Trainingsmenge, auf denen die Klassifizierer übereinstimmen oder nicht übereinstimmen, berechnet werden.

Zur Bewertung der Wahl der Merkmale ist es außerdem möglich, jeweils zwei Merkmale zur Erstellung eines Streudiagramms der Trainingsmenge gegenüberzustellen und so visuell auf gute Trennung zu überprüfen.

1.2 Methoden der maschinellen Klassifikation

Wie oben erwähnt lassen sich verschiedene Typen von Himmelsobjekten anhand charakteristischer Merkmale ihrer Spektren gut unterscheiden. Ebenso gibt es charakteristische Merkmale, die bei fast allen Objekten einer Klasse mehr oder weniger stark ausgeprägt sind. Am Beispiel von Quasaren wird hier exemplarisch aufgezeigt, wie eine geeignete Menge charakteristischer Merkmale zu einem trainierten Klassifizierer führt, der imstande ist, unbekannte Spektren anhand dieser Merkmale zu klassifizieren.

Aus Gründen der besseren Übersichtlichkeit werden dabei nur zwei besonders charakteristische Merkmale behandelt: die Intensitäten am linken und am rechten Ende des extrahierten Kontinuums der Spektren. Typisch für Quasare ist ein in Richtung zunehmender Wellenlänge abfallendes Kontinuum. Damit ist der stetige „Verlauf der Intensität [...] der Strahlung [...] geglättet und bereinigt von Absorptions- und Emissionslinien“ ([Wal13, Kapitel 3]) gemeint. Sie wird z. B. über Splines angenähert (rot in Abbildung 1.1).

Diese Merkmale lassen sich als reelle Funktionen F_1, F_2 definiert auf der Menge aller Spektren auffassen.⁵ Dass die Spektren s der in der Trainingsmenge als Quasare klassifizierten Objekte ein abfallendes Kontinuum aufweisen, lässt sich stark vereinfachend mit der Eigenschaft $F_1(s) > F_2(s)$ ausdrücken. Im Merkmalsraum

$$M_{F_1, F_2} := F_1(\text{Spektren}) \times F_2(\text{Spektren}) \subset \mathbb{R}^2$$

lassen sich die Spektren auf ihren Merkmalsvektor $(F_1(s), F_2(s))^T \in M_{F_1, F_2}$ abbilden und ihre charakteristische Eigenschaft führt dazu, dass (die meisten) Spektren von Quasaren im Merkmalsraum unter die Hauptdiagonale abgebildet werden, wo $F_1(s) > F_2(s)$ gilt. Im Gegensatz dazu ist bei den meisten der in der Trainingsmenge als Nichtquasare klassifizierten Objekte diese Eigenschaft nicht gegeben oder nicht so stark ausgeprägt.

In der hier vorgestellten Werkzeugkette werden zwei Verfahren zur Lösung des Entscheidungsproblems, welcher Klasse ein unbekanntes Objekt anhand seines Merkmalsvektors zugeordnet werden soll, verwendet: k -nächste-Nachbarn (kNN), das auf der Klassifikation einer lokalen Umgebung im Merkmalsraum beruht, sowie Support Vector Machines (SVM), die die Merkmalsvektoren zur Konstruktion einer Grenze im Merkmalsraum nutzen. Im Folgenden wird die Arbeitsweise beider Verfahren kurz skizziert.

⁵In Ausnahmefällen wird bei der Berechnung von Merkmalen noch zusätzliche Information verwendet, etwa eine Abweichung von einem Referenzspektrum, das von der Trainingsmenge abhängt. Diese Erweiterung ist jedoch an dieser Stelle nicht von Belang und würde die Definition nur unnötig verkomplizieren.

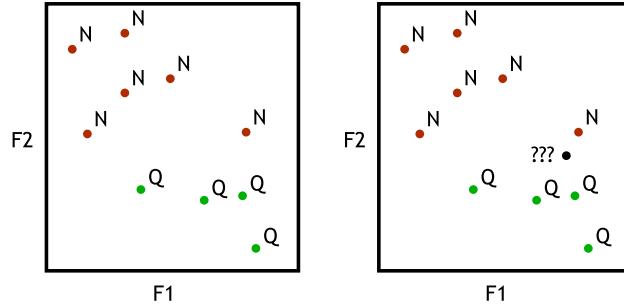


Abbildung 1.3: Merkmalsvektoren $(F_1(s), F_2(s))^T$ im Merkmalsraum M_{F_1, F_2} von Spektren s der Objekte der Trainingsmenge. Dabei sind Quasare mit Q und Nichtquasare mit N ausgezeichnet. Die Eigenschaft des abfallenden Kontinuums bei Quasaren wird darin sichtbar, dass sie mehr oder weniger weit unterhalb der Hauptdiagonalen abgebildet werden, wo $F_1(s) > F_2(s)$ gilt. Im rechten Bild ist ein bisher unklassifiziertes Objekt in den Merkmalsraum abgebildet. Mit Hilfe seiner Position im Merkmalsraum in Relation zur Trainingsmenge wird ein Klassifizierungsversuch unternommen. Bildquelle: [OCSW12a]

1.2.1 Lokale Demokratie: die k nächsten Nachbarn

Beim k -nächste-Nachbarn-Verfahren (kurz: kNN) wird für die Klassifikation anhand des Merkmalsvektor eines unbekannten Objekts die lokale Dichte der Merkmalsvektoren von Objekten der Trainingsmenge betrachtet. Der Parameter k gibt dabei indirekt die Größe der zu betrachtenden Umgebung an.

Zum Merkmalsvektor des unbekannten Objekts werden die k Merkmalsvektoren von bereits klassifizierten Objekten mit minimaler Distanz gesucht.⁶ Unter diesen Nachbarvektoren entscheidet ein Mehrheitsvotum über die Klassifizierung des unbekannten Objekts (siehe [HTF11, Abschnitt 13.3.]). Um eindeutige Ergebnisse zu erhalten, empfiehlt es sich, k ungerade zu wählen.

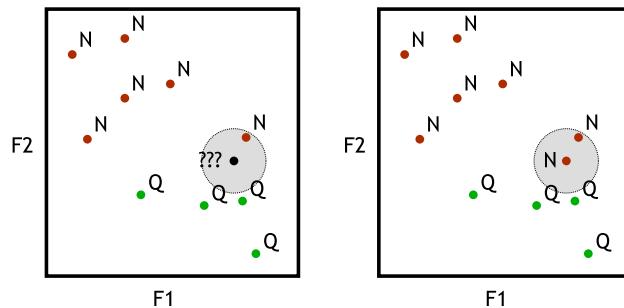


Abbildung 1.4: Klassifikation eines unbekannten Objekts mit dem 1NN-Verfahren: es wird die Klasse des im Merkmalsraum nächstgelegenen Objekts übernommen, gemessen mit der Euklidischen Distanz. In diesem Fall ist der nächste Nachbar der Merkmalsvektor eines als Nichtquasar klassifizierten Objekts. Daher wird das unbekannte Objekt ebenfalls als Nichtquasar klassifiziert. Bildquelle: [OCSW12a]

⁶In dieser Werkzeugkette wurde mit der Euklidischen Metrik gearbeitet. Andere Metriken sind aber ebenso denkbar und haben ihre Anwendungen.

Abbildung 1.4 zeigt, dass das Verfahren für ein klein gewähltes k anfällig für die Fehlklassifikation aufgrund von Ausreißern ist. Der nächste Nachbar befindet sich eher in einem Bereich des Merkmalsraumes, der von den Merkmalsvektoren von Quasaren dominiert wird. Wegen seiner Nähe wird er bei der Klassifizierung trotzdem den Quasaren vorgezogen.

Wählt man den Parameter k größer, wird der Einfluss dieser Ausreißer innerhalb der Trainingsmenge verringert. In Abbildung 1.5 sieht man, dass die höhere Dichte von Merkmalsvektoren von Quasaren in der Umgebung die Klassifikation ändert. In der Umgebung befinden sich zwei Quasare und nur ein Nichtquasar, wodurch das unbekannte Objekt als Quasar angesehen wird.

Sobald k sich aber in der Größenordnung der Mächtigkeit der Trainingsmenge bewegt, wird jedoch klar, dass k auch nicht beliebig erhöht werden sollte. Dann nämlich hängt die Klassifikation nur noch von der Verteilung innerhalb der Trainingsmenge, nicht aber von der lokalen Dichte im Merkmalsraum ab. Eine gute Wahl für diesen Parameter zu finden ist eine Aufgabe des Vergleichs von Klassifizierern anhand einer Gütebewertung.

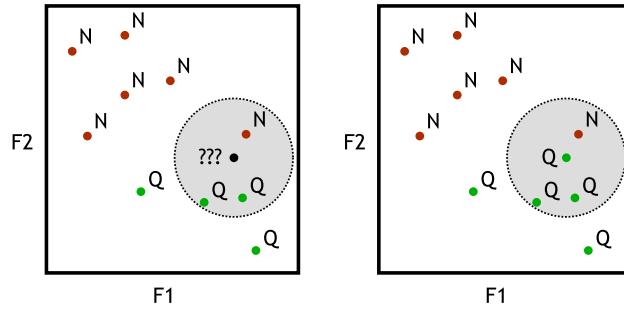


Abbildung 1.5: Bei einer 3NN-Klassifikation wird die Umgebung um den Merkmalsvektor des unbekannten Objekts so weit ausgedehnt, dass drei Merkmalsvektoren mit bekannter Klassifizierung in Reichweite gelangen. Gehören zwei oder mehr von ihnen einer bestimmten Klasse von Objekten an, so wird diese übernommen. Bildquelle: [OCSW12a]

1.2.2 Eine Grenze ziehen: Support Vector Machines

Der Trainingsprozess eines k NN-Klassifizierers ist mit dem Befüllen des Merkmalsraumes ausgehend von der Trainingsmenge abgeschlossen und es werden erst bei der Klassifikation Distanzen zu neuen Merkmalsvektoren berechnet. Man kann sich aber auch einen einmalig aufwendigeren Trainingsprozess wünschen, der in der Definition einer Grenze zwischen gesuchten und uninteressanten Objekten im Merkmalsraum resultiert, so dass die Klassifizierung eines unbekannten Objekts schnell wird, da nur noch geprüft wird, auf welcher Seite der Grenze der zugehörige Merkmalsvektor liegt.

Dieser Ansatz wird mit Hilfe von Support Vector Machines realisiert, deren Grundlagen hier vorgestellt werden (siehe auch [HTF11, Kapitel 12] und [BHW]). Mit der Trainingsmenge sei hier die Menge T bestehend aus N Paaren mit einem p -dimensionalen Merkmalsvektor und einer Klassifikation ± 1 gemeint:

$$T := \{(\mathbf{x}_i, y_i) \in \mathbb{R}^p \times \{-1, 1\} \mid i = 1, \dots, N\}$$

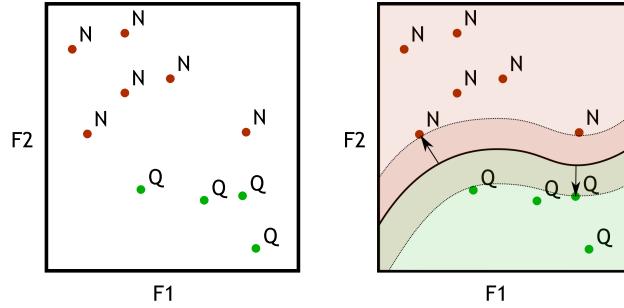


Abbildung 1.6: Mit Hilfe von Support Vector Machines kann im Merkmalsraum (links: gefüllt mit Merkmalsvektoren aus der Trainingsmenge) eine Grenze definiert werden, die zu den einzelnen Merkmalsvektoren der verschiedenen Klassen einen möglichst großen Abstand hat (rechts). Bildquelle: [OCSW12a]

Im besonders einfachen Fall, dass die Punkte mit verschiedener Klassifizierung im Merkmalsraum linear separierbar sind, ist das Klassifikationsproblem schnell zu entscheiden. Dann gibt es nämlich eine Hyperebene (gegeben durch einen Normalenvektor \mathbf{w} und einen Bias b)

$$\{\mathbf{x} \in \mathbb{R}^p \mid \mathbf{w}^T \mathbf{x} + b = \langle \mathbf{w}, \mathbf{x} \rangle + b = 0\},$$

die alle Punkte der Trainingsmenge entsprechend ihrer Klassifikation trennt. Einem unbekannten Merkmalsvektor \mathbf{x} lässt sich durch $\text{sign}(\mathbf{w}^T \mathbf{x} + b)$ eine Zahl ± 1 oder 0 (falls \mathbf{x} genau auf der Hyperebene liegt) zuordnen, die codiert, auf welcher Seite der Hyperebene der Merkmalsvektor liegt und damit zu welcher Klasse das zugehörige unbekannte Objekt gehören soll.

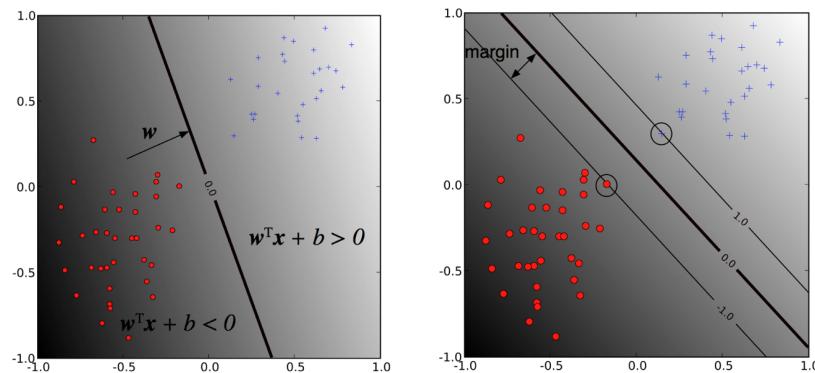


Abbildung 1.7: Ein mit Trainingsdaten linear separabel gefüllter Merkmalsraum. Die trennende Hyperebene im Bild wird über ihren Normalenvektor \mathbf{w} sowie eine durch den Bias b gegebene Verschiebung relativ zum Ursprung gegeben. Eine solche Trennung ist im Allgemeinen nicht eindeutig. Naheliegend ist, die trennende Hyperebene so zu wählen, dass der Abstand von Merkmalsvektoren der Trainingsmenge maximal ist (rechts). Bildquelle: [BHW]

Um eine möglichst gute Trennung zu erreichen, soll ein Rand M um diese Hyperebene, in dem keine Merkmalsvektoren der Trainingsmenge liegen, maximal sein.

Das Training lässt sich dann als Optimierungsproblem bezüglich dieser Hyperebene beschreiben: gesucht wird der normierte Normalenvektor \mathbf{w} , bezüglich dessen der Abstand M von allen Merkmalsvektoren zur Hyperebene maximal ist:

$$\max_{\mathbf{w}, b, \|\mathbf{w}\|=1} M \text{ mit } \forall i = 1, \dots, N : y_i(\mathbf{w}^T \mathbf{x}_i + b) \geq M$$

Da man den Bias b entsprechend anpassen kann, lässt sich durch eine Verknüpfung von M mit der Länge des Normalenvektors $M := 1/\|\mathbf{w}\|$ das Problem übersichtlicher ausdrücken:

$$\min_{\mathbf{w}, b} \|\mathbf{w}\| \text{ mit } \forall i = 1, \dots, N : y_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1$$

Im Allgemeinen ist ein linear separabler Merkmalsraum jedoch ein seltenes Glück. Merkmalsvektoren können als einzelne Ausreißer umgeben von solchen anderer Klasse liegen – in dem Fall existiert überhaupt keine trennende Hyperebene. Aber auch im linear separablen Fall können einzelne Merkmalsvektoren verschiedener Klassen so eng beieinander liegen, dass die resultierende Hyperebene mit maximalem Abstand von nur kleinen Abweichungen zweier verschieden klassifizierter Merkmalsvektoren stark abhängt.

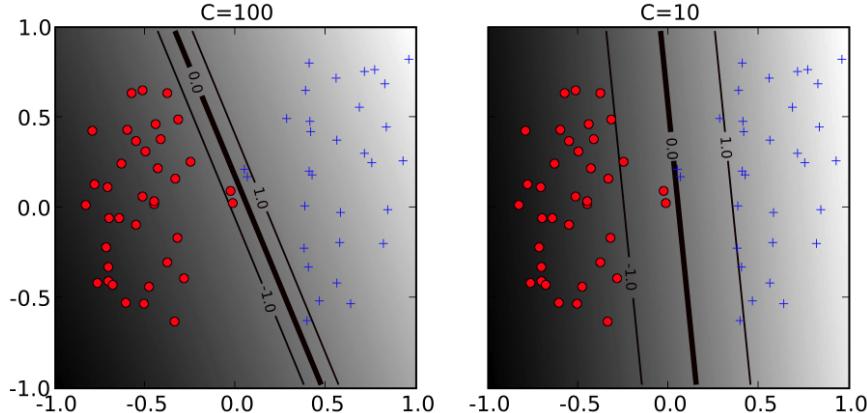


Abbildung 1.8: Dieser Merkmalsraum ist linear separabel, jedoch liegen einige Merkmalsvektoren verschiedener Klassen so nah beieinander, dass die resultierende Trennungshyperebene bei über den C -Parameter geregelter geringerer Toleranz von Abweichungen von einer visuell-intuitiven Trennung abweicht (links). Lässt man durch eine höhere Toleranz einen größeren Anteil von im Randbereich liegenden Merkmalsvektoren zu, wird die Trennung besser (rechts). Auch einzelne Ausreißer auf der falschen Seite der Trennhyperebene können so toleriert werden. Bildquelle: [BHW]

In beiden Fällen ist es erwünscht, dass bei der Optimierung vereinzelte Fehlklassifizierungen angenommen werden, die aber das Ergebnis im Ganzen nicht zu stark beeinflussen. Das wird mit Schlupfvariablen ξ_i für Merkmalsvektoren \mathbf{x}_i realisiert. Dabei bedeutet $0 < \xi_i \leq 1$, dass der zugehörige Merkmalsvektor innerhalb des Randes liegt, $1 < \xi_i$, dass er auf der falschen Seite der trennenden Hyperebene liegt. Über den Parameter C wird geregelt, wie stark solche ungünstig liegenden Merkmalsvektoren beim Optimieren der Hyperebene ins Gewicht fallen:

$$\min_{\mathbf{w}, b} \|\mathbf{w}\| + C \sum_{i=1}^n \xi_i \quad \text{mit } \forall i = 1, \dots, N : y_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1 - \xi_i \text{ und } \xi_i \geq 0$$

Ist C also groß, so fallen ungünstig liegende Merkmalsvektoren stärker ins Gewicht und die resultierende Hyperebene nähert sich der an, die man ohne Toleranz von ungünstig liegenden Merkmalsvektoren erhält. Je kleiner C wird, desto weiter dürfen Merkmalsvektoren innerhalb des Randes oder sogar auf der falschen Seite der trennenden Hyperebene liegen und der Rand M kann größer gewählt werden (siehe Abbildung 1.8).

Ein weiterer Aspekt, der mit realen Daten auftritt, ist, dass eine gute Trennung im Merkmalsraum im Allgemeinen mit Hyperebenen nicht gut genug an die Daten angepasst werden kann. Häufig würde eine gekrümmte Hyperfläche die Klassen besser trennen (siehe Abbildung 1.9).

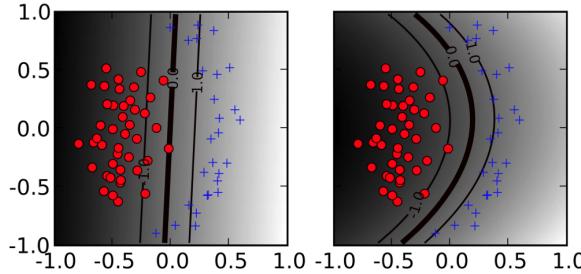


Abbildung 1.9: Die Lage der Merkmalsvektoren in diesem Fall lässt eine Trennung mit einer Hyperebene nur mit großer Toleranz von Abweichungen zu. Vielmehr erscheint eine gekrümmte Hyperfläche als passender für diese Trainingsmenge. Mit Hilfe von Kernelfunktionen kann das mit nur geringen Abweichungen von der bisherigen Definition des Optimierungsproblems realisiert werden. Bildquelle: [BHW]

Um das zu realisieren, werden die Merkmalsvektoren mit Hilfe einer Funktion $\phi : \mathbb{R}^p \rightarrow V$ in einen höherdimensionalen Raum mit Skalarprodukt $\langle \cdot, \cdot \rangle_V$ überführt, in dem die Daten eine bessere lineare Separierbarkeit aufweisen. Die für das Optimierungsproblem zu erfüllende Bedingung ändert sich dann zu

$$\forall i = 1, \dots, N : y_i(\langle \mathbf{w}, \phi(\mathbf{x}_i) \rangle_V + b) \geq 1 - \xi_i \text{ und } \xi_i \geq 0$$

Obwohl ein solches ϕ schwer zu berechnen sein kann, hilft der sogenannte „Kernel-Trick“, das Problem besser zu handhaben. Dazu sei zunächst festgehalten, dass man den Normalenvektor \mathbf{w} einer Hyperebene im untransformierten Merkmalsraum auch als Linearkombination von Vektoren aus der Trainingsmenge schreiben kann: $\mathbf{w} = \sum_{j=1}^N \alpha_j \mathbf{x}_j$. Dann erhält man als neue Bedingung:

$$\forall i = 1, \dots, N : y_i \left(\sum_{j=1}^N \alpha_j \langle \phi(\mathbf{x}_j), \phi(\mathbf{x}_i) \rangle_V + b \right) \geq 1 - \xi_i \text{ und } \xi_i \geq 0$$

Mit der Definition der zugehörigen *Kernelfunktion*

$$k : \mathbb{R}^p \times \mathbb{R}^p \rightarrow \mathbb{R}, \quad (\mathbf{x}, \mathbf{y}) \mapsto \langle \phi(\mathbf{x}), \phi(\mathbf{y}) \rangle_V$$

lässt sich obige Bedingung für das Optimierungsproblem schreiben als

$$\forall i = 1, \dots, N : y_i \left(\sum_{j=1}^N \alpha_j k(\mathbf{x}_j, \mathbf{x}_i) + b \right) \geq 1 - \xi_i \text{ und } \xi_i \geq 0$$

Es stellt sich heraus, dass in der Praxis die Kernelfunktion (und damit auch obige Ungleichung) wesentlich einfacher zu berechnen ist als ϕ , wodurch das Optimierungsproblem ohne große Änderungen für nichtlineare Trennungen angepasst werden kann.

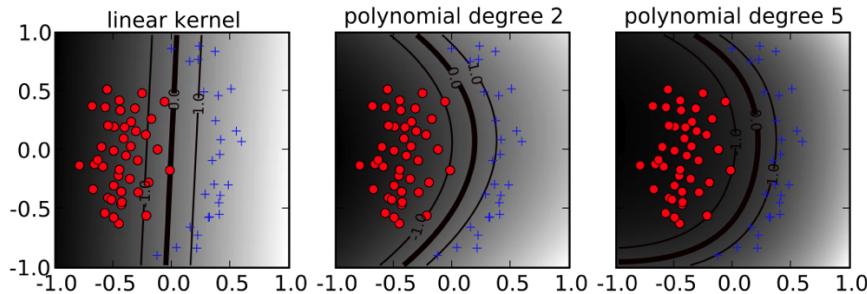


Abbildung 1.10: Mögliche Trennungen unter Verwendung des polynomiellen Kernels von verschiedenem Grad d . Bildquelle: [BHW]

In der Werkzeugkette verwendete Kernelfunktionen dazu sind etwa der *polynomiale Kernel vom Grad d* (siehe Abbildung 1.10)

$$k(\mathbf{x}, \mathbf{y}) := \langle \mathbf{x}, \mathbf{y} \rangle^d \text{ bzw. } (\langle \mathbf{x}, \mathbf{y} \rangle + 1)^d$$

oder der *Gauss- bzw. RBF-Kernel* (siehe Abbildung 1.11)

$$k(\mathbf{x}, \mathbf{y}) := \exp(-\gamma \|\mathbf{x} - \mathbf{y}\|^2),$$

manchmal mit der Bezeichnung $\gamma = 1/2\sigma^2$.

1.3 Gütebeurteilung für die Klassifizierung

Ein wesentlicher Bestandteil der durch die hier vorgestellte Werkzeugkette beschriebenen Arbeitsweise ist die Beurteilung der Güte von Klassifizierern, die durch eine Kombination von Merkmalsmenge und bestimmter Parametersätze für Klassifizierungsmethoden oder zur Simplifizierung von Spektren definiert sind. Ein wichtiges Werkzeug zur Beurteilung der Trennfähigkeit von Merkmalskombinationen sind die Streudiagramme. Die Güte eines Klassifizierers kann durch eine Kreuzvalidierung beurteilt werden.

Streudiagramme

Ein wichtiges Ziel bei der Konfiguration von Klassifizierern ist es, solche Merkmale $F_n : \text{Spektren} \rightarrow \mathbb{R}$ zu finden, die die Spektren einer Trainingsmenge sauber trennen. Auch wenn einzelne Merkmale dies im Allgemeinen nicht leisten, kann doch

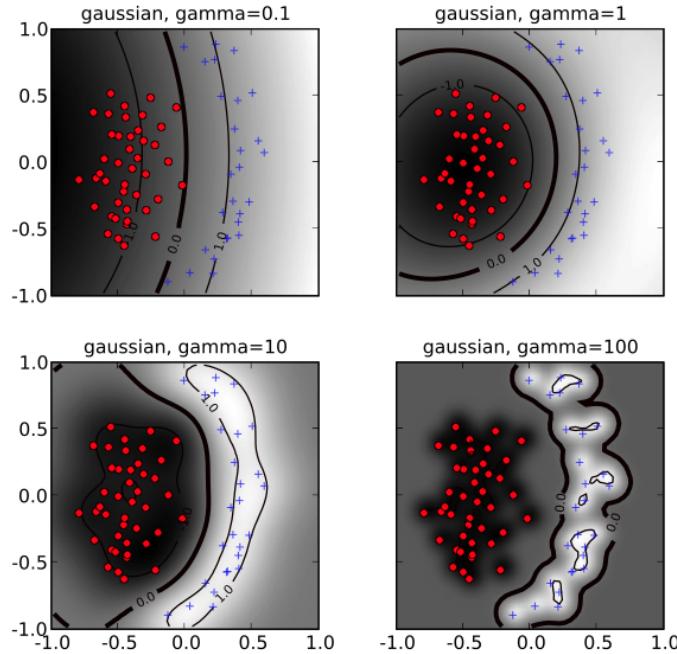


Abbildung 1.11: Mögliche Trennungen unter Verwendung des RBF-Kernels mit verschiedenen Parametern γ . Auffällig ist, dass sich die Trennung für große Werte von γ extrem stark an die Trainingsmenge anpasst. Bildquelle: [BHW]

die Kombination von zwei Merkmalen schon zu besseren Ergebnissen führen. Die Beurteilung der Kombination von zwei Merkmalen kann mit Hilfe von Streudiagrammen visuell durchgeführt werden. Diese sind einfach eine grafische Darstellung des (zweidimensionalen) Merkmalsraumes und ermöglichen anhand der intuitiven Trennbarkeit des Merkmalsraumes eine Beurteilung von Abhängigkeiten zwischen den Merkmalen (siehe Abbildung 1.12).

Kreuzvalidierung

Um die Merkmalsmenge und Parametrisierung von Klassifizierern zu optimieren, genügt nicht allein die intuitive visuelle Verifizierung einer guten Trennung für je zwei Merkmale. Vielmehr ist ein berechenbares Gütemaß für die Klassifikation wünschenswert, anhand dessen Merkmale und Parameter automatisiert optimiert werden können. Der *Matthews correlation coefficient* (MCC) auf Grundlage einer Kreuzvalidierung (siehe [HTF11, Kapitel 7.1]) kann ein solches Gütemaß realisieren.

Dazu wird die Trainingsmenge T in k möglichst gleichgroße Teilmengen T_1, \dots, T_k unterteilt. Für jedes $i = 1, \dots, k$ kann der zu bewertende Klassifizierer mit der Vereinigung der $k - 1$ Trainingsmengen T_j mit $j \neq i$ trainiert werden. Die bereits klassifizierten Objekte aus T_i übernehmen dann die Rolle unbekannter Objekte und werden vom Klassifizierer als Treffer oder Nichttreffer klassifiziert. Dieses Ergebnis wird mit der aus der Trainingsmenge bekannten Klassifikation verglichen, wodurch man nach Durchlauf aller Iterationen $i = 1, \dots, k$ jedes Objekt einstufen kann als *true positive* (korrekt als Treffer klassifiziert), *true negative* (korrekt als Nichttreffer klassifiziert), *false positive* (fälschlicherweise als Treffer klassifiziert) oder *false negative* (fälschlicherweise als Nichttreffer klassifiziert).

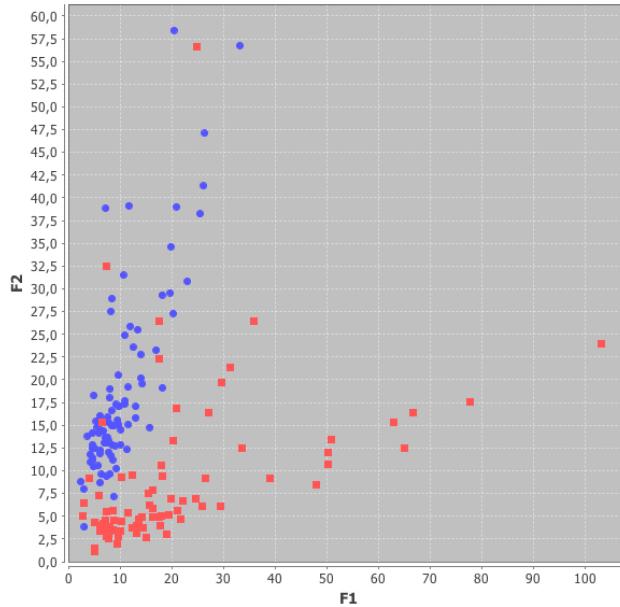


Abbildung 1.12: Ein Streudiagramm mit Quasaren (rot) und sternbildenden Galaxien (blau) für die Merkmale F_1 und F_2 . Wie weiter oben beschrieben, führt das bei Quasaren im Allgemeinen abfallende Kontinuum dazu, dass deren Merkmalsvektoren im Streudiagramm eher im unteren rechten Bereich dargestellt werden. Die hier blau dargestellten sternbildenden Galaxien teilen diese Eigenschaft nicht, was durch Betrachtung von Merkmal F_1 allein jedoch nicht deutlich wird, da es den Großteil der Objekte beider Klassen in den Bereich [0, 30] abbildet. Erst durch die Kombination mit F_2 wird eine bessere Trennung möglich. Einzelne rote Ausreißer im „blauen Lager“ zeigen jedoch, dass die Trennung noch weiter verbessert werden sollte, etwa durch Hinzunahme weiterer Merkmale, die Quasare ohne abfallendes Kontinuum anders erkennen.

Zur Berechnung des MCC werden dann lediglich die Anzahlen TP, TN, FP, FN der klassifizierten Objekte aus der Trainingsmenge benötigt, die jeweils als *true positive*, *true negative*, *false positive*, *false negative* eingestuft wurden:

$$\text{MCC} := \frac{\text{TP} \cdot \text{TN} - \text{FP} \cdot \text{FN}}{\sqrt{(\text{TP} + \text{FP}) \cdot (\text{TP} + \text{FN}) \cdot (\text{TN} + \text{FP}) \cdot (\text{TN} + \text{FN})}}$$

Der MCC selbst hat einen Wertebereich von $-1 \leq \text{MCC} \leq 1$ und wird so interpretiert: im Fall von $\text{MCC} = 1$ hat der untersuchte Klassifizierer perfekt gearbeitet, alle Objekte wurden in Übereinstimmung mit der Trainingsmenge klassifiziert. Im Fall von $\text{MCC} = -1$ hat der Klassifizierer jeweils genau falsch entschieden. $\text{MCC} = 0$ schließlich ist die Güte eines Klassifizierers, der zufällig entscheidet, welcher Klasse ein Objekt zugeordnet werden soll.

Zusätzlich zu einem möglichst hohen MCC sind jeweils die Mengen von Objekten aus der Trainingsmenge von Interesse, die in der Kreuzvalidierung als FP oder FN eingestuft wurden, denn diese unterscheiden sich bei der gewählten Konfiguration an Merkmalen und Parametern anscheinend so stark von den restlichen Objekten aus der Trainingsmenge, dass sie der falschen Klasse zugeordnet werden. Die Untersu-

Training				
Test	Training	Training	Training	Training
Training	Test	Training	Training	Training
Training	Training	Test	Training	Training
Training	Training	Training	Test	Training
Training	Training	Training	Training	Test

Abbildung 1.13: Kreuzvalidierung: die Trainingsmenge wird in k möglichst gleichgroße Teile unterteilt, die nacheinander die Rolle der unbekannten Objekte übernehmen, die von einem Klassifizierer, der mit den restlichen $k - 1$ Teilen trainiert wurde, klassifiziert werden. Das Ergebnis kann dann mit der aus der Trainingsmenge bekannten Klassifikation verglichen werden und geht in die Berechnung des Gütemaßes ein.

chung dieser Objekte legt häufig eine Anpassung von Merkmalen oder Parametern nahe, kann aber auch als Grundlage für eine Neubewertung der Trainingsmenge dienen.

Kapitel 2

Die Fréchet-Distanz als Abstandsbegriff polygonaler Kurven

Einen Abstandsbegriff für (polygonale) Kurven zu beschreiben, ist keine kanonisch lösbare Aufgabe. Die Wahl einer Distanzfunktion hängt davon ab, welche Aspekte der Kurven jeweils von Interesse sind und welche besonderen Merkmale von der Distanz berücksichtigt oder wohlwollend ignoriert werden sollen.

Zentraler Gegenstand der Diskussion in diesem Kapitel ist die nach *Maurice Fréchet* benannte Fréchet-Distanz, deren Berechnung in [AG92] präzise definiert wurde. Interessante weitere Diskussionen von Varianten dieser Distanz finden sich etwa in [BBW] (Ähnlichkeit von Teilkurven) und [DHP] (Zulassung von Abkürzungen). Als weitere Variante wird in diesem Kapitel die partielle Fréchet-Distanz nach [dCGM⁺13] vorgestellt, bevor im nächsten Kapitel eine nach Wellenlängen lokalisierbar sensible Variante der Fréchet-Distanz diskutiert wird, die die Standard-Fréchet-Distanz als Spezialfall enthält.

Zunächst stehen aber die Grundlagen sowie die exakte Beschreibung der Gestalt des im Mittelpunkt jeder Berechnung von Fréchet-Distanzen stehenden Freespace-Diagramms im Vordergrund.

2.1 Einführung und Abgrenzung

Betrachtet man zunächst von zwei Kurven¹ jeweils nur ihre Bilder als (nicht-leere und kompakte, [For06]) Punktmengen $A, B \subset \mathbb{R}^2$, so lässt sich die *Hausdorff-Distanz* δ_H unter Verwendung der Euklidischen Metrik $d : \mathbb{R}^2 \times \mathbb{R}^2 \rightarrow \mathbb{R}^{\geq 0}$ wie folgt definieren:

$$\delta_H(A, B) := \max \left(\sup_{a \in A} \inf_{b \in B} d(a, b), \sup_{b \in B} \inf_{a \in A} d(a, b) \right)$$

Man kann zeigen, dass diese auf der Menge der nichtleeren kompakten Teilmengen von \mathbb{R}^2 sogar eine Metrik ist [Hen, Seite 71 f]. Damit hat man einen formell

¹ Eine Kurve φ sei hier eine stetige Abbildung $\varphi : [a, b] \rightarrow \mathbb{R}^2$ eines Intervalls mit $a < b$ in die Euklidische Ebene. Durch Umparametrisierung kann man jede Kurve auch auf dem Einheitsintervall definieren.

eindeutig definierten und universellen verwendbaren Abstandsbegriff für Kurven in der Ebene. Jedoch trifft die Hausdorff-Distanz nur eine allgemeine Aussage über die Punktmengen dieser Kurven und berücksichtigt nicht ihren Verlauf, wodurch Kurven, deren Bilder sich als Menge ähneln, generell als ähnlich angesehen werden, auch wenn ihr Verlauf (also das Verhalten ihrer Bilder, wenn Parameter monoton durch ihre Definitionssintervalle laufen) sich stark unterscheidet (Abbildung 2.1).

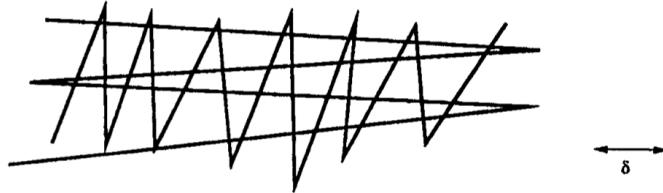


Abbildung 2.1: Zwei Kurven in der Ebene mit geringer Hausdorff-Distanz δ , die jedoch einen stark unterschiedlichen Verlauf haben. Bildquelle: [AG92, Seite 76].

Um einen Abstandsbegriff zu definieren, der den Verlauf zweier Kurven berücksichtigt, muss man also betrachten, in welcher Richtung und wie schnell eine Kurve von einem zum anderen Ende durchlaufen wird. Eine Möglichkeit dazu lässt sich sinnbildlich als den Weg eines Menschen mit Hund beschreiben, die jeweils auf ihrer eigenen Kurve laufen. Die kürzeste Leine, die die beiden verbunden sicher von ihren jeweiligen Start- zu den Zielpunkten bringt, kann man als Maß für den Abstand der beiden Kurven betrachten. Dabei soll variieren können, wie schnell Mensch und Hund auf ihren Wegen laufen unter der Bedingung, dass keiner sich jemals rückwärts bewegt.

Die Formalisierung dieser Beschreibung ergibt genau die **Fréchet-Distanz** zweier Kurven $f, g : I \rightarrow \mathbb{R}^2$ als den größten „gleichzeitigen“ Abstand von Punkten auf f und g bei Betrachtung aller nur möglichen monotonen Umparametrisierungen:

$$\delta_F(f, g) := \inf_{\substack{\alpha: I \rightarrow [0, 1] \\ \beta: I \rightarrow [0, 1]}} \max_{t \in I} d(f(\alpha(t)), g(\beta(t)))$$

mit α, β stetig und monoton wachsend.

2.2 Berechnung von Fréchet-Distanzen polygonaler Kurven

In den nachfolgenden Ausführungen werden ausschließlich polygonale Kurven betrachtet, was einerseits die Berechnung stark vereinfacht, andererseits aber auch die Anwendung in der astronomischen Datenverarbeitung widerspiegelt. Eine Kurve $\varphi : [0, n] \rightarrow \mathbb{R}^2$ heißt *polygonal*, falls alle Abschnitte $\varphi|_{[i, i+1]}$ mit $i \in \{0, \dots, n-1\}$ stückweise affin sind, also gilt:

$$\varphi(i + \lambda) = (1 - \lambda)\varphi(i) + \lambda\varphi(i + 1) \text{ für alle } \lambda \in I.$$

Dabei wird n die *Länge* von φ genannt [AG92, Seite 76]. Seien nun also zwei polygonale Kurven in der Ebene $T_1 : [0, n] \rightarrow \mathbb{R}$ und $T_2 : [0, m] \rightarrow \mathbb{R}^2$ von $a := T_1(0)$ nach $b := T_1(n)$ bzw. von $c := T_2(0)$ nach $d := T_2(m)$ gegeben (siehe Abbildung 2.2), für deren Fréchet-Distanz wir uns interessieren.

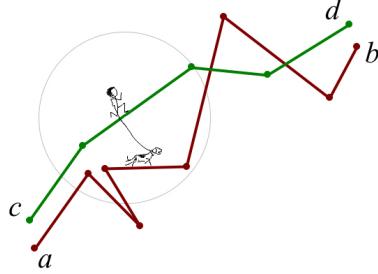


Abbildung 2.2: Zwei polygonale Kurven T_1, T_2 mit den Akteuren Mensch und Hund.

Diese Situation lässt verschiedene Problemstellungen zu, die in diesem und im nächsten Kapitel unter verschiedenen Voraussetzungen bearbeitet werden:

- Beim **Fréchet-Distanz-Entscheidungsproblem** wird danach gefragt, ob zu einer gegebenen maximalen Distanz δ Parametrierungen $\alpha : I \rightarrow [0, n]$ von T_1 und $\beta : I \rightarrow [0, m]$ von T_2 existieren, so dass für alle $t \in I$ gilt: $d(T_1(\alpha(t)), T_2(\beta(t))) \leq \delta$.
- Die **Fréchet-Distanz** selbst gibt den geringsten Abstand δ an, für den das vorherige Entscheidungsproblem zu einem positiven Ergebnis führt.
- Weiterhin kann man nach der **partiellen Fréchet-Distanz** fragen. Dabei wird zu einer gegebenen maximalen Distanz δ versucht, den Teil von T_1 bzw. T_2 zu minimieren, der linear ersetzt werden muss, damit das Entscheidungsproblem positiv wird.

Die Ansätze, diese Probleme zu lösen, verwenden alle als Hilfsmittel das *Freespace-Diagramm* zweier polygonaler Kurven zu gegebenem maximalen Abstand δ .

2.2.1 Das Freespace-Diagramm

Seien zunächst $\varphi, \psi : I \rightarrow \mathbb{R}^2$ zwei Liniensegmente, also polygonale Kurven der Länge 1. Dann kann man zu je einem Parameter für die beiden Liniensegmente den Abstand der resultierenden Punkte auf den Linien messen. Das liefert eine Funktion auf dem (zweidimensionalen) Parameterraum des Linienpaars, die direkt von der verwendeten Metrik des \mathbb{R}^2 abhängt²:

$$d_{\varphi, \psi} : I^2 \rightarrow \mathbb{R}^{\geq 0}, \quad (s, t) \mapsto d(\varphi(s), \psi(t))$$

Dann interessiert man sich für die Teilmengen ihres Definitionsbereiches I^2 , für die ihr Funktionswert kleiner oder gleich einem vorgegebenen δ ist, denn diese Mengen bestehen aus Parameterpaaren, für die die zugehörigen Punkte auf den Liniensegmenten einen Abstand $\leq \delta$ haben. Sie bilden den sogenannten *Freespace* oder *Whitespace*. Das Komplement, also die Parameterpaare, die Punkte mit größerem Abstand induzieren, heißt *forbidden space* oder *Blackspace*.

Um den Freespace besser zu verstehen, werden im Folgenden die Ränder dieser Mengen untersucht, die sich auch als Höhenlinien der Funktion $d_{\varphi, \psi}$ auffassen lassen.

²In dieser Arbeit wird hierfür stets die Euklidische Metrik verwendet, Resultate sind aber ebenso z. B. mit der L_1 -Metrik möglich (vgl. [AG92] und [dCGM⁺13])

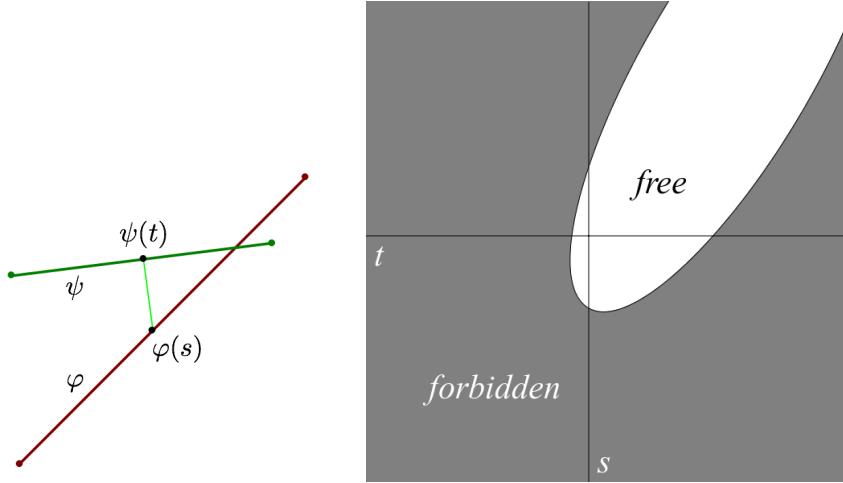


Abbildung 2.3: Zwei Liniensegmente φ und ψ zusammen mit ihrem Parameterraum I^2 , in dem ein Bereich als *Freespace* ausgezeichnet ist. Der markierte Punkt (s, t) ist z. B. frei, weil die zugehörigen Punkte $\varphi(s)$ und $\psi(t)$ einen Abstand kleiner als ein vorgegebenes δ haben.

Im allgemeineren Fall polygonaler Kurven $T_1 : [0, n] \rightarrow \mathbb{R}^2$, $T_2 : [0, m] \rightarrow \mathbb{R}^2$ ist der Parameterraum des Kurvenpaars das Rechteck $[0, n] \times [0, m]$.

Definition 2.1 Die (rechteckigen) Mengen $C^{i,j} := [i-1, i] \times [j-1, j]$ im Parameterraum $[0, n] \times [0, m]$ von T_1, T_2 heißen Zellen. Da ihnen jeweils das Paar von Liniensegmenten $T_1|_{[i-1,i]}, T_2|_{[j-1,j]}$ zugeordnet werden kann, kann man in ihnen analog zur obigen Situation den Freespace

$$C_W^{i,j} = \{(s, t) \in C^{i,j} \mid d(T_1(s), T_2(t)) \leq \delta\}$$

auszeichnen. Als Vereinigung über die Zellen erhält man den gesamten Freespace $W := \bigcup_{i,j} C_W^{i,j}$. Alle Zellen von T_1, T_2 zusammengenommen mit dem ausgezeichneten Freespace heißen Freespace-Diagramm zum Abstand δ .

Alle Paare von (monotonen) Parametrisierungen dieser zwei polygonalen Kurven lassen sich als (in beiden Richtungen monotone) Wege im Parameterraum von der linken unteren Ecke $(0, 0)$ in die rechte obere Ecke (n, m) beschreiben. Damit lassen sich die vorher beschriebenen drei Probleme im Kontext des Parameterraums beschreiben:

- Das Fréchet-Distanz-Entscheidungsproblem ist die Frage, ob ein in beide Richtungen monotoner Weg von der linken unteren in die rechte obere Ecke existiert, der vollständig im Freespace verläuft.
- Die Fréchet-Distanz zweier polygonaler Kurven ist das kleinste δ , so dass noch ein in beide Richtungen monotoner Weg von der linken unteren in die rechte obere Ecke des zugehörigen Freespace-Diagramms existiert, der vollständig im Freespace verläuft.
- Bei der partiellen Fréchet-Distanz zweier polygonaler Kurven schließlich wird zu gegebenem maximalen Abstand δ ein in beide Richtungen monotoner Weg

von der linken unteren in die rechte obere Ecke gesucht, dessen Abschnitt durch Freespace maximal lang ist.³

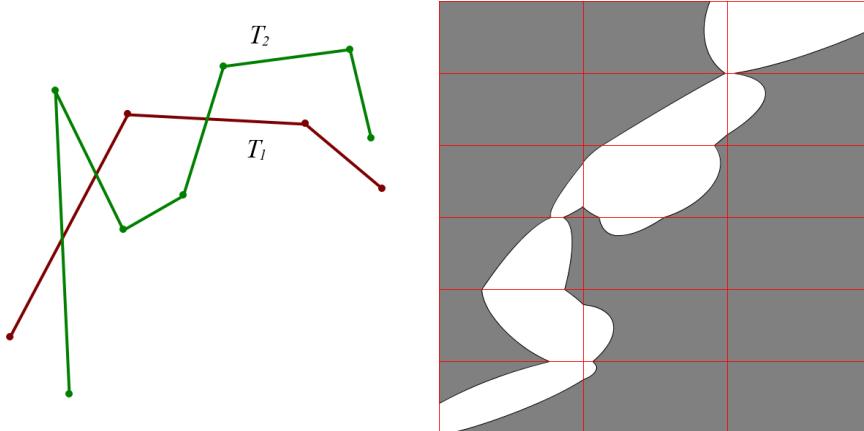


Abbildung 2.4: Zwei polygonale Kurven T_1 (Länge 3) und T_2 (Länge 6) zusammen mit ihrem Freespace-Diagramm bestehend aus dem Parameterraum $[0, 3] \times [0, 6]$ unterteilt in $3 \cdot 6$ Zellen und ausgezeichnetem Freespace (weiß).

Vor der Diskussion dieser Probleme wird aber zunächst beschrieben, wie der Free-space in Zellen des Freespace-Diagramms überhaupt genau aussieht. Es seien ein maximaler Abstand $\delta > 0$ sowie zwei Liniensegmente zwischen den Punkten $a \neq b$ sowie $c \neq d$ im Euklidisch-metrischen Raum (\mathbb{R}^2, d) gegeben, deren zugehörige Freespace-Zelle jetzt untersucht werden soll. Aus $n \cdot m$ solchen Zellen setzt sich dann das zugehörige Freespace-Diagramm zweier polygonaler Kurven der Längen n bzw. m zusammen.⁴

$$\begin{aligned}\varphi : I \rightarrow \mathbb{R}^2, \quad s &\mapsto a + s(b - a) \\ \psi : I \rightarrow \mathbb{R}^2, \quad t &\mapsto c + t(d - c)\end{aligned}$$

Seien $\bar{\varphi}, \bar{\psi} : \mathbb{R} \rightarrow \mathbb{R}^2$ die Fortsetzungen dieser Segmente zu ganzen Linien. Falls diese beiden Linien parallel sind, so hat der Freespace dieser Zelle die Gestalt eines (möglicherweise zur ganzen Zelle oder zu einer Linie der Breite 0 oder zur leeren Menge entarteten) Schlauchs. Ansonsten hat man es mit einer Ellipse zu tun.

2.2.2 Der Freespace als Schlauch

Es wird ein konstruktiver Weg vorgestellt, im Fall paralleler induzierter Linien einen Schlauch in der Freespace-Zelle zu konstruieren, der genau dem Freespace entspricht. Seien also $\bar{\varphi}$ und $\bar{\psi}$ parallel.

³Die genaue Berechnung der Länge der Abschnitte eines solchen Wegs ist ein bisschen komplizierter und wird später näher erläutert. Für diesen Moment soll die intuitive Vorstellung genügen.

⁴OBdA werden hier über I parametrisierte Segmente untersucht. Durch Umparametrisierung kann jedes Liniensegment über I parametrisiert werden: Die nötige Parametertransformation ist Addition der Position des Segments innerhalb einer polygonalen Kurve $T : [0, n] \rightarrow \mathbb{R}^2$. Das m -te Segment (von 0 an gezählt) lässt sich durch Komposition mit $(s \mapsto m + s)$ über I parametrisieren: $\varphi : I \rightarrow \mathbb{R}^2, s \mapsto T|_{[m, m+1]}(m + s)$.

Es bezeichne $\pi_{\bar{\psi}} : \mathbb{R}^2 \rightarrow \bar{\psi}(\mathbb{R})$ die (affine) orthogonale Projektion der Euklidischen Ebene auf die Linie $\bar{\psi}$. Betrachte nun die Projektionen $\pi_{\bar{\psi}}(\varphi(0))$ und $\pi_{\bar{\psi}}(\varphi(1))$ des Start- und Endpunktes von φ auf $\bar{\psi}$. Sie haben von allen Punkten auf $\bar{\psi}$ jeweils zu ihren Urbildern $\varphi(0)$ und $\varphi(1)$ minimalen Abstand, weil die Linien parallel sind. Seien $t_1, t_2 \in \mathbb{R}$ die zugehörigen Parameter mit $\bar{\psi}(t_1) = \pi_{\bar{\psi}}(\varphi(0))$ und $\bar{\psi}(t_2) = \pi_{\bar{\psi}}(\varphi(1))$. Betrachte nun das Liniensegment

$$\pi_{\bar{\psi}}(\varphi) : I \rightarrow \mathbb{R}^2, \quad s \mapsto \bar{\psi}(t_1) + s \cdot (\bar{\psi}(t_2) - \bar{\psi}(t_1)).$$

Lemma 2.2 *Mit obiger Definition von $\pi_{\bar{\psi}}(\varphi)$ gilt:*

- (a) *Das Liniensegment $\pi_{\bar{\psi}}(\varphi)$ liegt vollständig auf der Projektionslinie $\bar{\psi}$ und es gilt für $s \in I$:*

$$\pi_{\bar{\psi}}(\varphi)(s) = \bar{\psi}(t_1 + s \cdot (t_2 - t_1)).$$

- (b) *Projektion und Applikation von Linien(funktionen) sind vertauschbar, das folgende Diagramm kommutiert also:*

$$\begin{array}{ccc} I & \xrightarrow{\pi_{\bar{\psi}}(\varphi)} & \bar{\psi}(\mathbb{R}) \\ \varphi \searrow & & \swarrow \pi_{\bar{\psi}} \\ & \varphi(I) & \end{array}$$

Die zweite Aussage wird in Abbildung 2.5 geometrisch interpretiert.

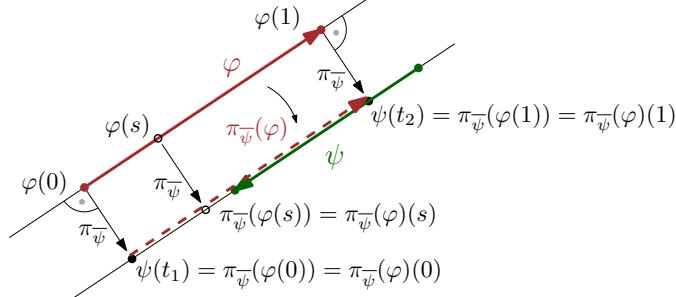


Abbildung 2.5: Geometrische Interpretation des kommutativen Diagramms aus Lemma 2.2 (b): man erhält denselben Punkt auf $\bar{\psi}$, gleichgültig ob man einen Punkt $\varphi(s)$ auf $\bar{\psi}$ projiziert oder ob man den gleichen Parameter s auf die projezierte Kopie $\pi_{\bar{\psi}}(\varphi)$ von φ anwendet.

Beweis.

- (a) Einsetzen der Definition von ψ bzw. $\bar{\psi}$ liefert für $s \in I$:

$$\begin{aligned} \pi_{\bar{\psi}}(\varphi)(s) &= \bar{\psi}(t_1) + s \cdot (\bar{\psi}(t_2) - \bar{\psi}(t_1)) \\ &= c + t_1(d - c) + s \cdot (c + t_2(d - c) - (c + t_1(d - c))) \\ &= c + t_1(d - c) + s \cdot (t_2(d - c) - t_1(d - c)) \\ &= c + t_1(d - c) + s \cdot (t_2 - t_1) \cdot (d - c) \\ &= c + (t_1 + s \cdot (t_2 - t_1)) \cdot (d - c) \\ &= \bar{\psi}(t_1 + s \cdot (t_2 - t_1)) \subset \bar{\psi}(\mathbb{R}). \end{aligned}$$

Damit ist $\pi_{\bar{\psi}}(\varphi)$ eine umparametrisierte Version von $\bar{\psi}$, deren Start- und Endpunkte genau die Projektionen von Start- und Endpunkt von φ sind:

$$\pi_{\bar{\psi}}(\varphi)(0) = \bar{\psi}(t_1) = \varphi(0), \quad \pi_{\bar{\psi}}(\varphi)(1) = \bar{\psi}(t_2) = \varphi(1).$$

(b) Es gilt für $s \in I$:

$$\begin{aligned} \pi_{\bar{\psi}}(\varphi)(s) &= \bar{\psi}(t_1) + s \cdot (\bar{\psi}(t_2) - \bar{\psi}(t_1)) \\ &= \pi_{\bar{\psi}}(\varphi(0)) + s \cdot (\pi_{\bar{\psi}}(\varphi(1)) - \pi_{\bar{\psi}}(\varphi(0))) \\ &= \pi_{\bar{\psi}}(a) + s \cdot (\pi_{\bar{\psi}}(b) - \pi_{\bar{\psi}}(a)) \\ &= \pi_{\bar{\psi}}(a) + s \cdot (b - a) \tag{*} \\ &= \pi_{\bar{\psi}}(a + s \cdot (b - a)) \tag{**} \\ &= \pi_{\bar{\psi}}(\varphi(s)) \end{aligned}$$

Bei (*) wird verwendet, dass a und b auf einer zu $\bar{\psi}$ parallelen Linie liegen, ihre Differenz wird von der Projektion also nicht beeinflusst. Auch bei (**) kommt dieses Argument zum Tragen. Der Vektor ist auch dann noch unter Projektion invariant, wenn er um s skaliert wird. Daher ist es irrelevant, ob er vor oder nach Projektion addiert wird.

□

Weil $\pi_{\bar{\psi}}$ eine Orthogonalprojektion ist, haben die Punktpaare

$$(\varphi(s), \pi_{\bar{\psi}}(s)) = (\varphi(s), \bar{\psi}(t_1 + s \cdot (t_2 - t_1))) \quad \text{für } s \in I$$

zueinander genau den Abstand, den die beiden Linien $\bar{\varphi}$ und $\bar{\psi}$ voneinander haben. Dies ist auch der kleinste Abstand, den Punkte auf den beiden Linien überhaupt voneinander haben können. Die zugehörigen Parameterpaare für φ und $\bar{\psi}$ sind von der Form $(s, t_1 + s \cdot (t_2 - t_1))$, was zumindest für die Start- und Endpunkte von φ stimmt:

$$\begin{aligned} s = 0 : \quad d(\varphi(0), \bar{\psi}(t_1 + 0 \cdot (t_2 - t_1))) &= d(a, \bar{\psi}(t_1)) \\ &= d(a, \pi_{\bar{\psi}}(\varphi(0))) \\ &= d(a, \pi_{\bar{\psi}}(a)) \quad \text{minimal} \\ s = 1 : \quad d(\varphi(1), \bar{\psi}(t_1 + 1 \cdot (t_2 - t_1))) &= d(b, \bar{\psi}(t_2)) \\ &= d(b, \pi_{\bar{\psi}}(\varphi(1))) \\ &= d(b, \pi_{\bar{\psi}}(b)) \quad \text{minimal} \end{aligned}$$

Diese Beobachtung legt eine Konstruktion im Parameterraum nahe:

Korollar 2.3 *Das Liniensegment der Paare von Parametern für φ bzw. $\bar{\psi}$*

$$\text{center} := \{(s, t) \in I \times \mathbb{R} \mid t = t_1 + s \cdot (t_2 - t_1)\}$$

induziert genau die Punktpaare mit minimalem Abstand auf dem Liniensegment φ bzw. der Linie $\bar{\psi}$. Siehe Abbildung 2.6.

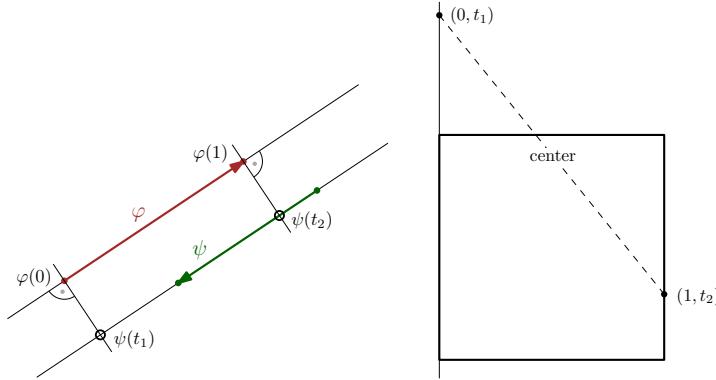


Abbildung 2.6: Die Projektion der Endpunkte von φ liefert zwei Parameterpaare $(0, t_1)$ und $(1, t_2)$. Im Parameterraum (rechts) ist zwischen ihnen die Verbindungslinee center gestrichelt eingezeichnet.

Beweis. Sei zunächst $(s, t) \in \text{center}$. Zu zeigen ist, dass das zugehörige Punktpaar auf φ bzw. $\bar{\psi}$ minimalen Abstand hat. Es gilt: $t = t_1 + s(t_2 - t_1)$ und damit ist das resultierende Punktpaar:

$$\begin{aligned} \varphi(s) \quad \text{und} \quad \bar{\psi}(t) &= \bar{\psi}(t_1 + s(t_2 - t_1)) \\ &= \pi_{\bar{\psi}}(\varphi)(s) && \text{Lemma 2.2 (a)} \\ &= \pi_{\bar{\psi}}(\varphi(s)) && \text{Lemma 2.2 (b)} \end{aligned}$$

Also ist $\bar{\psi}(t)$ die Projektion von $\varphi(s)$ auf $\bar{\psi}$ und damit ist der Abstand dieser beiden Punkte minimal.

Sei nun umgekehrt $\varphi(s)$ ein Punkt auf dem Liniensegment φ und $\bar{\psi}(t)$ ein Punkt auf der Linie $\bar{\psi}$, so dass der Abstand zwischen diesen Punkten minimal ist. Dann ist die Verbindungslinee zwischen den beiden Punkten orthogonal zu beiden Linien und damit $\bar{\psi}(t)$ die $\pi_{\bar{\psi}}$ -Projektion von $\varphi(s)$. Daher gilt:

$$\begin{aligned} \bar{\psi}(t) &= \pi_{\bar{\psi}}(\varphi(s)) \\ &= \pi_{\bar{\psi}}(\varphi)(s) && \text{Lemma 2.2 (b)} \\ &= \bar{\psi}(t_1 + s(t_2 - t_1)) && \text{Lemma 2.2 (a)} \end{aligned}$$

Also hat das resultierende Parameterpaar immer die Eigenschaft $t = t_1 + s(t_2 - t_1)$ und damit $(s, t) \in \text{center}$. \square

Außerdem lässt sich eine kleine Aussage über die Lage dieses Liniensegments machen, die es bei der Implementierung ermöglicht, es über eine der beiden Koordinaten parametrisiert aufzufassen. Da bei der Konstruktion die erste Koordinate bereits als von 0 bis 1 laufend angelegt wurde, bietet sich diese besonders an.

Lemma 2.4 *Das Liniensegment center im Parameterraum verläuft niemals senkrecht und niemals waagerecht.*

Beweis. Dass center nicht senkrecht verlaufen kann, ergibt sich direkt aus der Konstruktion mit den Punkten $(0, t_1)$ und $(1, t_2)$, denn ihre ersten Koordinaten sind verschieden.

Aber angenommen, center verlief waagerecht, dann wäre für alle Punkte dieser Linie die t -Koordinate gleich. Insbesondere wäre dies für die zwei aus der Konstruktion bekannten Punkte $(0, t_1)$ und $(1, t_2)$ der Fall, also auch für die $\bar{\psi}$ -Parameter $t_1 = t_2$ und die zugehörigen Punkte $\psi(t_1) = \bar{\psi}(t_2)$ auf $\bar{\psi}$. Da $\bar{\varphi}$ und $\bar{\psi}$ aber parallel sind, gilt Gleichheit dann auch für die Projektionsurbilder auf $\bar{\varphi}$

$$\{\varphi(0)\} = \pi_{\bar{\psi}}^{-1}(\bar{\psi}(t_1)) \cap \bar{\varphi}(\mathbb{R}) = \pi_{\bar{\psi}}^{-1}(\bar{\psi}(t_2)) \cap \bar{\varphi}(\mathbb{R}) = \{\varphi(1)\}$$

und damit

$$a = \varphi(0) = \varphi(1) = b.$$

Im Widerspruch dazu waren aber anfangs a, b als verschieden angenommen worden. \square

Es gibt also eine Linie im Parameterraum der beiden Segmente, die Auskunft über die Punktpaare mit kleinstem Abstand gibt und es ist auch bekannt, wie diese liegt. Wie genau der Freespace der Zelle aber damit in Verbindung steht, hängt vom vorgegebenen kleinsten Abstand δ ab: Ist δ kleiner als der Abstand der beiden induzierten Linien, so ist der Freespace der Zelle die leere Menge, denn keine zwei Punkte auf den beiden Liniensegmenten können sich nahe genug kommen. Falls Gleichheit gilt, ist center geschnitten mit I^2 als dem Parameterraum der beiden Liniensegmente genau der Freespace der Zelle⁵.

Falls δ größer ist, ist auch der Freespace größer, wie die folgende Konstruktion zeigt. Der δ -Kreis um $\varphi(0)$ hat mit der Linie $\bar{\psi}$ dann genau zwei Schnittpunkte. Diese Schnittpunkte lassen sich als Parameter t_3, t_4 von $\bar{\psi}$ realisieren, wobei ohne Beschränkung der Allgemeinheit $t_3 < t_4$ angenommen sei.

Lemma 2.5 *Die oben definierten Parameter t_3 und t_4 von $\bar{\psi}$ beschreiben die untere und obere Grenze des Freespaces geschnitten mit der linken Kante $s = 0$. Eine Verschiebung entlang center ergibt ein Parallelogramm als Freespace für alle Punktpaare auf φ und $\bar{\psi}$ (siehe Abbildung 2.7).*

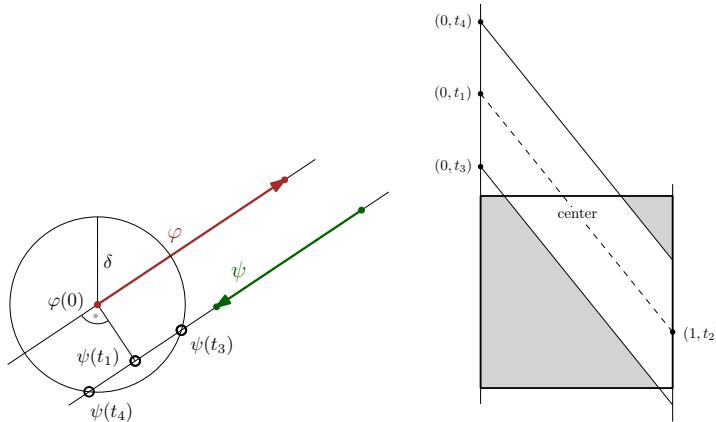


Abbildung 2.7: Wenn man einen δ -Kreis um den Startpunkt des Segments φ mit der Linie $\bar{\psi}$ schneidet, erhält man als Distanz der Parameter der Schnittpunkte die Dicke des Schlauches in ψ -Richtung.

⁵Falls die Liniensegmente auf ihren Linien weit genug auseinander liegen, ist der Schnitt natürlich ebenfalls leer.

Beweis. Alle Punkte $\overline{\psi}(\mathbb{R}) \cap \overline{B_\delta(\varphi(0))}$ von $\overline{\psi}$ geschnitten mit dem (abgeschlossenen) δ -Kreis um $\varphi(0)$ haben zu $\varphi(0)$ einen Abstand $\leq \delta$, die zugehörigen Parameterpaare

$$\{(s, t) \in \{0\} \times \mathbb{R} \mid d(\varphi(s), \overline{\psi}(t)) \leq \delta\}$$

sind also im Freespace. Wird t als Parameter von $\overline{\psi}$ variiert, so dass $t \notin [t_3, t_4]$ gilt, so ist $\overline{\psi}(t)$ nicht mehr im δ -Kreis um $\varphi(0)$ und $(0, t)$ folglich nicht mehr im Freespace. Die Menge $\{0\} \times [t_3, t_4]$ beschreibt also genau den Freespace geschnitten mit der linken Kante $s = 0$.

Variiere nun der Parameter $s \in I$ von φ . Aufgrund der Parallelität der Geraden erhält man jedoch als Abstand der Parameter $t_{s,u} < t_{s,o}$ der zwei Schnittpunkte eines δ -Kreises um $\varphi(s)$ mit $\overline{\psi}$ eine Konstante $t_{s,o} - t_{s,u} = t_4 - t_3$. Genau so liegt der Parameter der Projektion $\pi_{\overline{\psi}}(\varphi(t))$ stets genau zwischen $t_{s,u}$ und $t_{s,o}$. Mit einer Überlegung wie zuvor erhält man, dass für jedes $s \in I$ die Menge $\{s\} \times [t_{s,u}, t_{s,o}]$ die zugehörigen freien Parameterpaare enthält.

Damit ist der Freespace der Zelle das Parallelogramm, das sich aus der folgenden Vereinigung ergibt:

$$\{(s, t) \in I \times \mathbb{R} \mid d(\varphi(s), \overline{\psi}(t)) \leq \delta\} = \bigcup_{s \in I} \{s\} \times [t_{s,u}, t_{s,o}] \supset \text{center}$$

Die Inklusion von center sieht man an der Überlegung, dass für jedes $s \in I$ stets die Projektion $\pi_{\overline{\psi}}(\varphi(s))$ zwischen den Punkten $\overline{\psi}(t_{s,u})$ und $\overline{\psi}(t_{s,o})$ liegt. Der zugehörige Parameter t mit $\overline{\psi}(t) = \pi_{\overline{\psi}}(\varphi(s))$ aber bildet gerade zusammen mit s die Parameterpaare in center. \square

Konstruktionsvorschrift

Aus den in diesem Abschnitt gemachten Aussagen lässt sich eine Vorschrift zur Konstruktion des Freespace-Schlauchs im Fall paralleler Liniensegmente φ und ψ zusammenfassen:

1. Projeziere die Punkte $\varphi(0)$ und $\varphi(1)$ auf die Linie $\overline{\psi}$ und ermittle die zugehörigen Parameter t_1, t_2 . Falls der Abstand der Linien mit $d(\varphi(0), \overline{\psi}(t_1)) > \delta$ zu groß ist, ist der Freespace die leere Menge.
2. Die Linie im Parameterraum zwischen den Punkten $(0, t_1)$ und $(1, t_2)$ bildet die Mittellinie des Freespace-Schlauchs. Falls der Abstand zwischen den Linien mit $d(\varphi(0), \overline{\psi}(t_1)) = \delta$ genau dem minimalen Abstand entspricht, ist die Mittellinie geschnitten mit I^2 der gesamte Freespace.
3. Schneide die Linie $\overline{\psi}$ mit einem δ -Kreis um $\varphi(0)$ und ermittle die Parameter $t_3 < t_4$ der beiden Schnittpunkte. Ihr Abstand $t_4 - t_3$ ist die Ausdehnung des Freespace-Schlauchs in t -Richtung.⁶ Das resultierende Parallelogramm geschnitten mit I^2 ist der gesamte Freespace der Zelle.

2.2.3 Der Freespace als Ellipse

Im Allgemeinen befinden sich die für eine Freespace-Diagrammzelle zu betrachtenden Liniensegmente nicht auf parallelen Linien. In dem Fall hat der Freespace die

⁶Falls benötigt könnte man die Breite des Freespace-Schlauchs als die Entfernung des Punktes $(0, t_3)$ von der Linie durch $(0, t_1)$ und $(1, t_2)$ ermitteln.

Form einer Ellipse. In diesem Abschnitt wird diskutiert, wie diese Ellipse genau aussieht, also wo Mittelpunkt und Halbachsen liegen und wie lang letztere sind. Es wird eine algebraische Lösung des Problems vorgestellt, indem eine Matrix angegeben wird, die eine Norm induziert, bezüglich derer die gesuchte Ellipse (in den Koordinatenursprung verschoben) ein Kreis vom Radius δ^2 ist. Die Halbachsen werden dann durch die Eigenräume dieser Matrix angegeben. Die Verschiebung wird durch Umparametrisierung der Linien realisiert. Zur Motivation zunächst eine Beobachtung⁷ mit Bezug auf die in 2.2.1 vorgestellte Funktion

$$d_{\varphi,\psi} : I^2 \rightarrow \mathbb{R}^{\geq 0}, \quad (s,t) \mapsto d(\varphi(s),\psi(t)).$$

Ihr Graph ist ein sich nach oben öffnender Kegel (mit nicht notwendig kreisrunden Schnitten) und ihre δ -Höhenlinien sind die gesuchten Ellipsen zum minimalen Abstand δ . Die Spitze des Kegels ist gerade die 0-Höhenlinie und somit das Parameterpaar (s_0, t_0) , für das der Abstand der zugehörigen Punkte auf den Linien $\bar{\varphi}(s_0), \bar{\psi}(t_0)$ den Abstand 0 hat, s_0 und t_0 sind also die Parameter des Schnittpunkts der Linien. Damit haben unabhängig von der Wahl von δ die gesuchten Ellipsen das Parameterpaar des Schnittpunkts der beiden Geraden als Mittelpunkt.

Da Ellipsen, deren Mittelpunkt nicht der Koordinatenursprung sind, nicht als Kreise einer Matrixnorm realisiert werden können, werden die Linien so umparametrisiert, dass das Parameterpaar $(0,0)$ den Schnittpunkt trifft. Die Umparametrisierung ist die Addition einer Konstanten, kann also später einfach rückgängig gemacht werden. Die Liniensegmente

$$\begin{aligned} \varphi_0 : I &\rightarrow \mathbb{R}^2, & s &\mapsto a + (s_0 + s)(b - a) && \text{und} \\ \psi_0 : I &\rightarrow \mathbb{R}^2, & t &\mapsto c + (t_0 + t)(d - c), \end{aligned}$$

besitzen den Schnittpunkt $\varphi_0(0) = \bar{\varphi}(s_0) = \bar{\psi}(t_0) = \psi_0(0)$ und lassen ohne konkrete Angabe vom maximalen Abstand δ die Definition der gewünschten Matrix zu:

Lemma 2.6 *Die von der symmetrischen Matrix*

$$M = \begin{pmatrix} d(a,b)^2 & -\langle(b-a), (d-c)\rangle \\ -\langle(b-a), (d-c)\rangle & d(c,d)^2 \end{pmatrix}$$

induzierte Norm $\|\cdot\|_M : \mathbb{R}^2 \rightarrow \mathbb{R}^{\geq 0}$ für Vektoren $p = \begin{pmatrix} s \\ t \end{pmatrix} \in \mathbb{R}^2$ mit

$$\|p\|_M := p^T \cdot M \cdot p = (s, t) \cdot \begin{pmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{pmatrix} \cdot \begin{pmatrix} s \\ t \end{pmatrix}$$

realisiert die Freespace-Ellipsen der Liniensegmente φ_0, ψ_0 zum maximalen Abstand δ als δ^2 -Kreise.

Beweis. Mit den Endpunkten dieser umparametrisierten Liniensegmente

$$\begin{aligned} (\alpha_1, \alpha_2) &= \alpha := \varphi_0(0) = \psi_0(0), \\ (\beta_1, \beta_2) &= \beta := \varphi_0(1), \\ (\gamma_1, \gamma_2) &= \gamma := \psi(1) \end{aligned}$$

⁷Aus Zeit- und Platzgründen ohne Beweise, die Erklärungen samt Illustration sollen jedoch ein intuitives Verständnis der Zusammenhänge ermöglichen.

ergibt sich für $p = \begin{pmatrix} s \\ t \end{pmatrix}$ vom dem Rand der (verschobenen) Ellipse:

$$\begin{aligned}
\delta^2 &= d(\varphi_0(s), \psi(t))^2 \\
&= + \frac{(\alpha_1 + s(\beta_1 - \alpha_1) - (\alpha_1 + t(\gamma_1 - \alpha_1)))^2}{(\alpha_2 + s(\beta_2 - \alpha_2) - (\alpha_2 + t(\gamma_2 - \alpha_2)))^2} \\
&= + \frac{(s(\beta_1 - \alpha_1) - t(\gamma_1 - \alpha_1))^2}{(s(\beta_2 - \alpha_2) - t(\gamma_2 - \alpha_2))^2} \\
&= + \frac{((\beta_1 - \alpha_1)^2 + (\beta_2 - \alpha_2)^2) \mathbf{s}^2}{-2 \cdot ((\beta_1 - \alpha_1)(\gamma_1 - \alpha_1) + (\beta_2 - \alpha_2)(\gamma_2 - \alpha_2)) \mathbf{s} \mathbf{t}} \\
&+ \frac{((\gamma_1 - \alpha_1)^2 + (\gamma_2 - \alpha_2)^2) \mathbf{t}^2}{d(\alpha, \beta)^2 \mathbf{s}^2 + (-2) \cdot \langle (\beta - \alpha), (\gamma - \alpha) \rangle \mathbf{s} \mathbf{t} + d(\alpha, \gamma)^2 \mathbf{t}^2} \\
&= d(a, b)^2 \mathbf{s}^2 + (-2) \cdot \langle (b - a), (d - c) \rangle \mathbf{s} \mathbf{t} + d(d, c)^2 \mathbf{t}^2 \quad (*) \\
&= d(a, b)^2 \mathbf{s}^2 + (-2) \cdot \langle (b - a), (d - c) \rangle \mathbf{s} \mathbf{t} + d(d, c)^2 \mathbf{t}^2
\end{aligned}$$

Der letzte Schritt ist möglich, obwohl das Zentrum der Ellipse durch Umparametrisierung in den Nullpunkt des Parameterraums verschoben wurde, da die auftretenden Terme lediglich von der Länge der Segmente oder dem Skalarprodukt ihrer Richtungsvektoren abhängen – beides wird aber von der Umparametrisierung nicht verändert.

Mit den Bezeichnungen

$$\begin{aligned}
m_{11} &:= d(a, b)^2 \\
m_{12} &:= m_{21} = -\langle (b - a), (d - c) \rangle \\
m_{22} &:= d(d, c)^2
\end{aligned}$$

lässt sich (*) schreiben als Gleichung mit der gewünschten Matrixnorm:

$$\begin{aligned}
\delta^2 &= m_{11} \mathbf{s}^2 + 2m_{12} \mathbf{s} \mathbf{t} + m_{22} \mathbf{t}^2 \\
&= (m_{11}s + m_{12}t)s + (m_{21}s + m_{22}t)t \\
&= (s, t) \cdot \begin{pmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{pmatrix} \cdot \begin{pmatrix} s \\ t \end{pmatrix} \\
&= p^T \cdot M \cdot p \\
&= \|p\|_M
\end{aligned}$$

□

Man kann an der Matrix schon ablesen, dass im Falle von senkrecht stehenden Linien die resultierenden Ellipsen von besonders einfacher Gestalt sind:

Lemma 2.7 *Stehen die Linien $\bar{\varphi}$ und $\bar{\psi}$ senkrecht, so ist die zugehörige Freespace-Ellipse zum maximalen Abstand δ nicht verdreht und sie hat*

$$\text{die Breite } \frac{2 \cdot \delta}{d(a, b)} \text{ und die Höhe } \frac{2 \cdot \delta}{d(c, d)}.$$

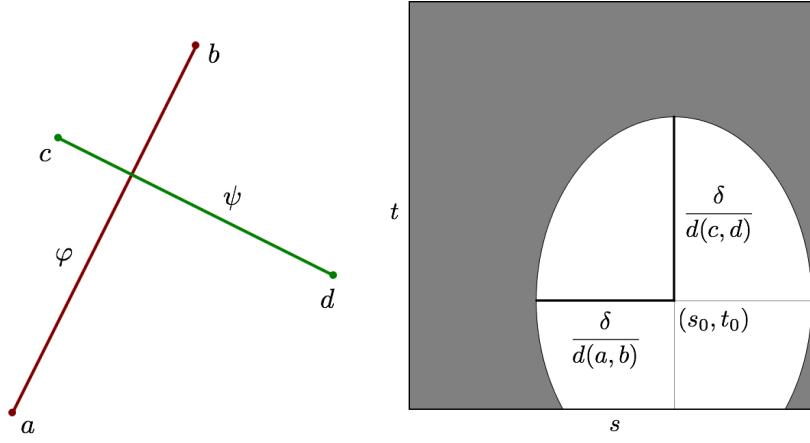


Abbildung 2.8: Liniensegmente φ und ψ mit ihrer Freespace-Zelle: Stehen die zugehörigen Linien senkrecht, so kann man Lage und Gestalt der dann nicht verdrehten Ellipse direkt ermitteln. Der Mittelpunkt ergibt sich durch die Parameter des Schnittpunktes der Linien (s_0, t_0) , die Längen der Halbachsen sind antiproportional zur Länge der zugehörigen Segmente und proportional zum maximalen Abstand δ . Da ψ hier kürzer ist, ist die Halbachse in Richtung des Parameters t von ψ länger.

Die Aussage dieses Lemmas wird in Abbildung 2.8 illustriert.

Beweis. Stehen die Linien senkrecht aufeinander, so sind ihre Richtungsvektoren orthogonal und deren Skalarprodukt verschwindet. Die Einträge der Matrix M neben der Hauptdiagonalen $m_{12} = m_{21} = -\langle(b-a), (d-c)\rangle$ verschwinden also ebenfalls und M wird eine Diagonalmatrix. M war so definiert, dass für $p = \begin{pmatrix} s \\ t \end{pmatrix}$ auf der Ellipse $\|p\|_M = \delta^2$ gilt. Sind $m_{12} = m_{21} = 0$, so folgt

$$\begin{aligned} \delta^2 &= \|p\|_M = p^T M p \\ &= (s, t) \cdot \begin{pmatrix} d(a, b)^2 & 0 \\ 0 & d(c, d)^2 \end{pmatrix} \cdot \begin{pmatrix} s \\ t \end{pmatrix} \\ &= (s, t) \cdot \begin{pmatrix} d(a, b)^2 s \\ d(c, d)^2 t \end{pmatrix} \\ &= d(a, b)^2 s^2 + d(c, d)^2 t^2 \\ &= \left\langle \begin{pmatrix} d(a, b) s \\ d(c, d) t \end{pmatrix}, \begin{pmatrix} d(a, b) s \\ d(c, d) t \end{pmatrix} \right\rangle \\ &= \left\| \begin{pmatrix} d(a, b) s \\ d(c, d) t \end{pmatrix} \right\|_2^2 \end{aligned}$$

und damit

$$\left\| \begin{pmatrix} d(a, b) s \\ d(c, d) t \end{pmatrix} \right\|_2 = \delta. \quad (\star\star)$$

Daraus folgt, dass die Vektoren eines δ -Kreises bezüglich der Euklidischen Norm in horizontaler Richtung um $1/d(a, b)$ und in vertikaler Richtung um $1/d(c, d)$ skaliert werden, um auf dem δ^2 -Kreis bezüglich der Matrixnorm zu landen. Insbesondere ist die Ellipse also nicht verdreht.

Sei nun $v = \begin{pmatrix} v_1 \\ 0 \end{pmatrix}$ auf der Ellipse geschnitten mit der x -Achse. Dann gilt mit (**)

$$\delta = \left\| \begin{pmatrix} d(a, b) v_1 \\ 0 \end{pmatrix} \right\|_2 = d(a, b) \cdot |v_1| \cdot \left\| \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right\|_2 = d(a, b) \cdot |v_1|$$

Und damit weiß man die Länge von v :

$$\|v\| = |v_1| = \frac{\delta}{d(a, b)}$$

Die Länge der horizontalen Halbachse beträgt das Doppelte der Länge eines Vektors auf der Ellipse geschnitten mit der x -Achse: $2 \cdot \delta/d(a, b)$. Mit der gleichen Rechnung für einen Vektor $w = \begin{pmatrix} 0 \\ w_2 \end{pmatrix}$ auf der Ellipse geschnitten mit der y -Achse erhält man die Länge der vertikalen Halbachse $2 \cdot \delta/d(c, d)$. \square

Als direkte Folgerung erhält man eine Aussage für einen besonders übersichtlichlichen Spezialfall:

Korollar 2.8 *Stehen die Linien $\bar{\varphi}$ und $\bar{\psi}$ senkrecht und sind die sie induzierenden Liniensegmente φ von a nach b und ψ von c nach d gleichlang, so ist die Freespace-Ellipse ein Kreis vom Radius $\delta/d(a, b)$.*

Beweis. Sind die Liniensegmente gleichlang, so gilt $d(a, b) = d(c, d)$. Mit Lemma 2.7 erhält man dann als Länge für die Halbachsen die identischen Werte

$$\frac{2 \cdot \delta}{d(a, b)} = \frac{2 \cdot \delta}{d(c, d)}.$$

\square

Im Allgemeinen Fall ist die Berechnung jedoch etwas komplizierter. Es wird mit einer Beobachtung begonnen.⁸ Die Anwendung einer Matrix, deren Norm die gesuchte Ellipse als „Kreis“ hat, hat auf die Punkte in einem richtigen Kreis den Einfluss, dass sie diese auf eine eben solche Ellipse abbildet. Diejenigen von ihnen, die dabei nur skaliert werden und ihre Richtung beibehalten, sind die Punkte, die auf den Halbachsen der Ellipse landen. Alle anderen werden zusätzlich in Richtung der längeren Halbachse verschoben (siehe Abbildung 2.9).

Somit liegen die Halbachsen der Ellipsen zu dieser Matrix in den Eigenräumen der Matrix. Die Eigenwerte zu diesen Eigenräumen geben an, wie stark Punkte auf der Ellipse geschnitten mit einer ihrer beiden Halbachsen von einem Kreis abweichen. Damit lassen sich aus ihnen die Längen der Halbachsen ermitteln:

Lemma 2.9 *Die Länge und Breite der Freespace-Ellipse zu den Liniensegmenten φ, ψ mit nichtorthogonalen Richtungsvektoren ist gegeben durch*

$$\frac{2 \cdot \delta}{\sqrt{\frac{m_{11} + m_{22}}{2} + \sqrt{\left(\frac{-m_{11} - m_{22}}{2}\right)^2 - m_{11} m_{22} + m_{12}^2}}}$$

bzw.

$$\frac{2 \cdot \delta}{\sqrt{\frac{m_{11} + m_{22}}{2} - \sqrt{\left(\frac{-m_{11} - m_{22}}{2}\right)^2 - m_{11} m_{22} + m_{12}^2}}}.$$

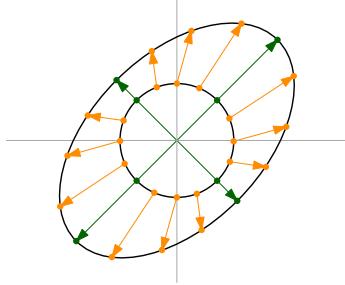


Abbildung 2.9: Bei der Anwendung unserer Matrix auf einen Kreis werden nur die Punkte auf den Halbachsen skaliert – alle anderen werden zusätzlich in Richtung der längeren Halbachse verschoben.

Beweis. Zunächst ermittelt man die Eigenwerte der zugehörigen Matrix M als Nullstellen des charakteristischen Polynoms:

$$\begin{aligned} 0 &= \chi_M(\lambda) := \det(\lambda E_2 - M) \\ &= \det \begin{pmatrix} \lambda - m_{11} & -m_{12} \\ -m_{21} & \lambda - m_{22} \end{pmatrix} \\ &= (\lambda - m_{11})(\lambda - m_{22}) - m_{12} m_{21} \\ &= \lambda^2 + (-m_{11} - m_{22})\lambda + m_{11} m_{22} - m_{12}^2 \end{aligned}$$

Lösungen für λ :

$$\lambda_1, \lambda_2 = \frac{m_{11} + m_{22}}{2} \pm \sqrt{\left(\frac{-m_{11} - m_{22}}{2}\right)^2 - m_{11} m_{22} + m_{12}^2}$$

Analog zum Beweis für den orthogonalen Fall gilt für $p = \begin{pmatrix} s \\ t \end{pmatrix}$ vom Rand der Ellipse geschnitten mit dem Eigenraum zum Eigenwert λ :

$$\delta^2 = \|p\|_M = p^T M p = p^T \lambda p = \lambda \langle p, p \rangle = \lambda \|p\|_2^2$$

$$\Rightarrow \|p\|_2 = \frac{\delta}{\sqrt{\lambda}}$$

Die Länge der Halbachse ergibt sich dann als die doppelte Länge $2 \cdot \delta / \sqrt{\lambda}$. \square

Um die Form der Ellipse endgültig zu verstehen, braucht man noch einen Drehwinkel. Aus praktischen Gründen genügt es für meinen speziellen Anwendungsfall, den Drehwinkel der längeren Halbachse zu ermitteln. Dabei hilft das folgende Lemma.

Lemma 2.10 *Die Verdrehung der Halbachse der durch M generierten Ellipse, die im Eigenraum zum Eigenwert λ liegt, beträgt*

$$\arccos \frac{1}{\sqrt{1 + \left(\frac{\lambda - m_{11}}{m_{12}}\right)^2}}$$

⁸Diese Beobachtung wird hier aus Zeit- und Platzgründen nicht detailliert bewiesen, soll aber ein intuitives Verständnis für die Grundlage der folgenden Rechnungen liefern.

Beweis. Die Verdrehung lässt sich als Drehwinkel eines Eigenvektors zum Eigenwert λ ermitteln. Den Eigenraum erhält man als Kern der Abbildung $\lambda E_2 - M$. Gesucht sind also Lösungen des linearen Gleichungssystems

$$\begin{pmatrix} \lambda - m_{11} & -m_{12} \\ -m_{21} & \lambda - m_{22} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix},$$

Da die Linien $\bar{\varphi}, \bar{\psi}$ als nichtorthogonal angenommen wurden, sind die Nebeneinträge $-m_{21} = \langle (b-a), (d-c) \rangle \neq 0$ und folglich die Eigenräume rotiert. Daher kann man in obigem Gleichungssystem gefahrlos $x := 1$ setzen um einen Repräsentanten $\neq 0$ des Eigenraumes, also einen Eigenvektor zu erhalten:

$$\lambda - m_{11} - m_{12} y = 0 \implies y = \frac{\lambda - m_{11}}{m_{12}}$$

Zur Bestimmung der Drehung der Ellipse wird der Winkel des ermittelten Eigenvektors $v := (1, (\lambda - m_{11})/m_{12})^T$ zur x -Achse im mathematisch-positiven Sinn gemessen:

$$\cos \angle(v) = \frac{\langle v, \begin{pmatrix} 1 \\ 0 \end{pmatrix} \rangle}{\|v\| \cdot \left\| \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right\|} = \frac{1}{\|v\|} = \frac{1}{\sqrt{1 + \left(\frac{\lambda - m_{11}}{m_{12}} \right)^2}}$$

Durch Anwendung von \arccos erhält man die gewünschte Aussage für den Winkel $\angle(v)$ des Eigenvektors v zum Eigenwert λ . \square

Konstruktionsvorschrift

Aus den in diesem Abschnitt gemachten Aussagen lässt sich eine Vorschrift zur Konstruktion der Freespace-Ellipse im Fall nichtparalleler Liniensegmente φ von a nach b und ψ von c nach d zusammenfassen:

1. Ermittle die Parameter s_0, t_0 des Schnittpunkts $\bar{\varphi}(s_0) = \bar{\psi}(t_0)$ der induzierten Linien.
2. Überprüfe, ob das Skalarprodukt der Richtungsvektoren $\langle (b-a), (d-c) \rangle$ verschwindet. In dem Fall ist die Ellipse entlang der Koordinatenachsen ausgerichtet und hat die Breite $2\delta/d(a,b)$ sowie die Länge $2\delta/d(c,d)$. Ihr Zentrum befindet sich im Punkt (s_0, t_0) .
3. Berechne die symmetrische Matrix

$$M = \begin{pmatrix} d(a,b)^2 & -\langle (b-a), (d-c) \rangle \\ -\langle (b-a), (d-c) \rangle & d(c,d)^2 \end{pmatrix}.$$

4. Ermittle die Eigenwerte von M :

$$\lambda_1, \lambda_2 = \frac{m_{11} + m_{22}}{2} \pm \sqrt{\left(\frac{-m_{11} - m_{22}}{2} \right)^2 - m_{11} m_{22} + m_{12}^2}$$

(Wähle den kleineren Eigenwert als λ_1 .⁹) Die Ellipse hat dann die Breite $2\delta/\sqrt{\lambda_1}$ und die Höhe $2\delta/\sqrt{\lambda_2}$.

⁹Daraus folgt in den weiteren Schritten, dass die generierten Ellipsen immer breiter als hoch sind. Im nächsten Schritt wird die Rotation bezüglich eines Eigenvektors zu diesem Eigenwert bestimmt. Man hätte also auch andersherum wählen können, aber es hätte auf die resultierende Ellipsen keinen Einfluss. Lediglich aus gesellschaftskritischen Gründen werden hier breite Ellipsen schlanken vorgezogen.

5. Berechne die Rotation der Ellipse mit Hilfe des für die Breite zuständigen Eigenwerts λ_1 :

$$\angle = \arccos \frac{1}{\sqrt{1 + \left(\frac{\lambda_1 - m_{11}}{m_{12}}\right)^2}}$$

6. Verschiebe die generierte Ellipse mit dem Mittelpunkt in (s_0, t_0) .

2.2.4 Das Fréchet-Distanz-Entscheidungsproblem

Von den vorher¹⁰ aufgezählten Problemstellungen, die sich aus der Definition der Fréchet-Distanz ergeben, ist das grundlegendste das Entscheidungsproblem für polygonale Kurven $T_1 : [0, n] \rightarrow \mathbb{R}^2$ und $T_2 : [0, m] \rightarrow \mathbb{R}^2$:

Beim **Fréchet-Distanz-Entscheidungsproblem** wird danach gefragt, ob zu einer gegebenen maximalen punktweisen Distanz δ Parametrisierungen $\alpha : I \rightarrow [0, n]$ von T_1 und $\beta : I \rightarrow [0, m]$ von T_2 existieren, so dass für alle $t \in I$ gilt: $d(T_1(\alpha(t)), T_2(\beta(t))) \leq \delta$.

Man kann die Parametrisierungen α, β zu einer in beide Richtungen monotonen Kurve $\xi : I^2 \rightarrow [0, n] \times [0, m]$ mit $\xi(0) = (0, 0)$ und $\xi(1) = (n, m)$ kombinieren.¹¹ Das passt zur Formulierung des Problems im Kontext des Freespace-Diagramms¹²:

Das Fréchet-Distanz-Entscheidungsproblem ist die Frage, ob ein in beide Richtungen monotoner Weg von der linken unteren in die rechte obere Ecke [des Freespace-Diagramms] existiert, der vollständig im Freespace verläuft.

Im Folgenden wird ein Algorithmus zur Lösung des Entscheidungsproblems vorgestellt [AG92, Seite 80]. Dazu zunächst eine wichtige Folgerung:

Korollar 2.11 *Der Freespace $C_W^{i,j}$ einer Zelle $C^{i,j}$ aus dem Freespace-Diagramm zweier polygonaler Kurven ist konvex.*

Beweis. In den Lemmata 2.5 und 2.6 wurde gezeigt, dass der nichtleere Freespace einer Zelle entweder aus einem Parallelogramm entlang einer Mittellinie oder einer Ellipse, jeweils geschnitten mit dem Quadrat I^2 , besteht. Der Schnitt kann natürlich auch leer sein. Da Schnitte konvexer Mengen wieder konvex sind, folgt damit die Behauptung. \square

Um im Kontext dieser konvexen Mengen eine nützliche Folgerung zu formulieren, sind zunächst einige Bezeichnungen sinnvoll:

Definition 2.12 *Der linke bzw. untere Rand einer Zelle $C^{i,j}$ werde bezeichnet durch die Segmente*

$$L^{i,j} := \{i-1\} \times [j-1, j] \quad \text{bzw.} \quad B^{i,j} := [i-1, i] \times \{j-1\}$$

Als freie Zellenrandstücke werden die Schnitte mit dem Freespace $C_W^{i,j}$ definiert:

$$L_W^{i,j} := L^{i,j} \cap C_W^{i,j} \quad \text{bzw.} \quad B_W^{i,j} := B^{i,j} \cap C_W^{i,j}.$$

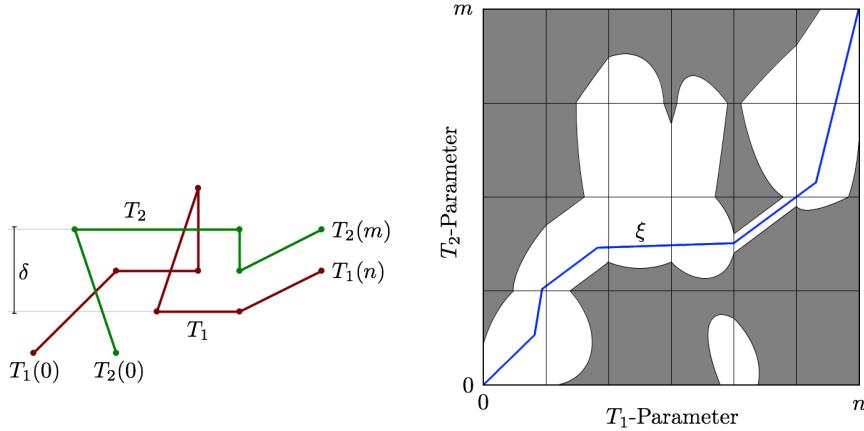


Abbildung 2.10: Zu den zwei polygonalen Kurven T_1, T_2 links wurde das δ ein kleines bisschen größer als der Abstand der beiden Hilfslinien gewählt. Die Kurve $\xi : I \rightarrow [0, n] \times [0, m]$ parametrisiert T_1 und T_2 so, dass zu jeder Zeit $t \in I$ das Parameterpaar $\xi(t)$ im Freespace liegt, also das Punktpaar $T_1(\pi_1(\xi(t))), T_2(\pi_2(\xi(t)))$ einen Abstand $\leq \delta$ hat. Sie ist eine von mehreren möglichen Kurven mit dieser Eigenschaft. Ihre Existenz löst das Fréchet-Distanz-Entscheidungsproblem.

Die Gestalt der beiden Kurven stammt aus [AG92, Seite 79], die Visualisierung wurde mit Hilfe der für diese Arbeit entwickelten Software realisiert.

Das folgende Lemma zeigt, dass es sinnvoll und problemlos möglich ist, die rechten bzw. oberen freien Zellenrandstücke durch die linken bzw. unteren freien Zellenrandstücke der rechten bzw. oberen Nachbarzellen auszudrücken. Der Sonderfall der Zellen mit $i = n$ oder $j = m$ (wenn die entsprechenden Nachbarzellen nicht existieren) lässt sich in der Implementierung umgehen.

Lemma 2.13 Für die Zellen $C^{i,j}$ mit $i < n$ und $j < m$ stimmen die benachbarten Ränder

$$R^{i,j} := \{i\} \times [j-1, j] = L^{i+1,j} \quad \text{bzw.} \quad T^{i,j} := [i-1, i] \times \{j\} = B^{i,j+1}$$

sowie die darauf liegenden freien Zellenrandstücke

$$R_W^{i,j} := R^{i,j} \cap C_W^{i,j} = L_W^{i+1,j} \quad \text{bzw.} \quad T_W^{i,j} := T^{i,j} \cap C_W^{i,j} = B_W^{i,j+1}$$

überein (siehe Abbildung 2.12).

Beweis. Die erste Aussage, dass die gemeinsamen Ränder benachbarter Zellen übereinstimmen, ergibt sich direkt aus der Definition der Zellenränder. Die zweite Aussage, dass auch die freien Randstücke benachbarter Zellen auf dem gemeinsamen Rand übereinstimmen, ist eine Stetigkeitsaussage des Freespace an Zellenrändern und wird hier durch ein Widerspruchsargument bewiesen:

Unter der Annahme, dass zwei zusammenliegende freie Zellenrandstücke $T_W^{i,j}$ und $B_W^{i,j+1}$ verschieden sind, folgt, dass mindestens eine der beiden Differenzmengen $T_W^{i,j} \setminus B_W^{i,j+1}$ oder $B_W^{i,j+1} \setminus T_W^{i,j}$ nicht leer ist.

¹⁰In Abschnitt 2.2 auf Seite 20

¹¹Die ursprünglichen Parametrisierungen α und β erhält man dann als die Projektionen auf die Koordinatenachsen $\pi_1 \circ \xi \circ (s \mapsto s/n) = \alpha$ und $\pi_2 \circ \xi \circ (t \mapsto t/m) = \beta$.

¹²2.2.1 auf Seite 21

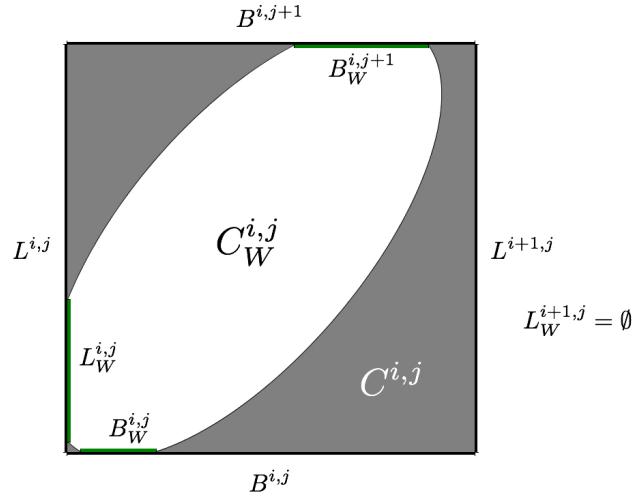


Abbildung 2.11: Die freien Zellenrandstücke ergeben sich als der Schnitt der Zellenränder mit dem Freespace der Zelle und können leer sein, wie hier am rechten Rand.

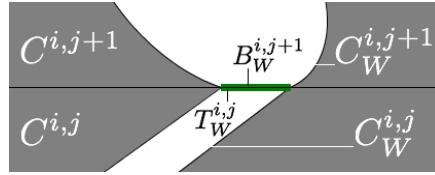


Abbildung 2.12: Die Aussage von Lemma 2.13: freie Zellenrandstücke auf den gemeinsamen Rändern benachbarter Zellen stimmen überein (hier: $T_W^{i,j} = B_W^{i,j+1}$).

Falls $B_W^{i,j+1} \setminus T_W^{i,j} \neq \emptyset$ ist, dann existiert ein Parameterpaar $(s, t) \in B_W^{i,j+1} \subset C_W^{i,j+1}$ aber $(s, t) \notin T_W^{i,j} \subset C_W^{i,j}$. Für alle diese in Frage kommenden Parameterpaare gilt $i-1 \leq s \leq i$ und $t = j$ nach Definition der Zellenränder. Die zu den Zellen gehörigen Liniensegmente sind durch die Einschränkungen $T_1|_{[i-1,i]}$ und $T_2|_{[j-1,j]}$ für $C^{i,j}$ bzw. $T_1|_{[i-1,i]}$ und $T_2|_{[j,j+1]}$ für $C^{i,j+1}$ gegeben und mit den möglichen Werten des Parameterpaars (s, j) kompatibel. Die obige Aussage über das Enthaltensein bzw. Nichtenthaltensein von (s, j) in den freien Zellenrandstücken bedeutet geometrisch

$$\delta < d(T_1|_{[i-1,i]}(s), T_2|_{[j-1,j]}(j)) = d(T_1(s), T_2(j))$$

aber

$$\delta \geq d(T_1|_{[i-1,i]}(s), T_2|_{[j,j+1]}(j)) = d(T_1(s), T_2(j)),$$

woraus zusammen mit der Stetigkeit von T_2 die Ungleichung $\delta < \delta$ folgt. Das ist unmöglich, ein solches Parameterpaar (s, j) kann also nicht existieren, also muss die Differenz $B_W^{i,j+1} \setminus T_W^{i,j} = \emptyset$ schon leer gewesen sein.

Mit dem gleichen Argument für vertauschte Mengen erhält man $T_W^{i,j} \setminus B_W^{i,j+1} = \emptyset$ und damit Gleichheit für die beiden freien Zellenrandstücke $T_W^{i,j} = B_W^{i,j+1}$.

Der Beweis für $R_W^{i,j} = L_W^{i+1,j}$ funktioniert analog. \square

Es lässt sich als nächstes eine Teilmenge des Freespace definieren, der eng mit der Lösung des Fréchet-Distanz-Entscheidungsproblems zusammenhängt: der *reachable space* oder *erreichbare Raum* im Freespace-Diagramm (vgl. [AG92, Seite 79f.]). Gemeinsam mit der Konvexitätsaussage aus Korollar 2.11 lässt sich damit die algorithmische Lösung des Fréchet-Distanz-Entscheidungsproblems einfach realisieren.

Definition 2.14 Zu gegebenem Freespace-Diagramm mit gesamtem Freespace W ist der *reachable space* oder *erreichbare Raum* einer Zelle $C^{i,j}$ gegeben als die Menge aller freien Punkte darin, die von der linken unteren Ecke $(0,0)$ des Diagramms aus frei erreichbar ist:

$$C_R^{i,j} := \left\{ (s,t) \in C_W^{i,j} \mid \begin{array}{l} \exists \xi : I \rightarrow W \text{ stetig und in beide Richtungen} \\ \text{monoton mit } \xi(0) = (0,0) \text{ und } \xi(1) = (s,t) \end{array} \right\}.$$

Die erreichbaren Zellenrandstücke definiert man als *Schnitt des erreichbaren Raums einer Zelle mit seinen Randstücken*

$$\begin{aligned} L_R^{i,j} &:= L^{i,j} \cap C_R^{i,j} \\ B_R^{i,j} &:= B^{i,j} \cap C_R^{i,j} \\ R_R^{i,j} &:= R^{i,j} \cap C_R^{i,j} \\ T_R^{i,j} &:= T^{i,j} \cap C_R^{i,j}. \end{aligned}$$

Genau wie bei freien Zellenrandstücken stimmen diese Teilmengen von benachbarten Randstücken überein:

Lemma 2.15 Es gilt

$$\begin{aligned} R_R^{i,j} &= L_R^{i+1,j} \quad \text{für } i = 1, \dots, n-1, j = 1, \dots, m \\ T_R^{i,j} &= B_R^{i,j+1} \quad \text{für } i = 1, \dots, n, \quad j = 1, \dots, m-1. \end{aligned}$$

Beweis. Sei $(s,t) \in R_R^{i,j}$, dann existiert eine Kurve $\xi : I \rightarrow W$ mit $\xi(0) = (0,0)$ und $\xi(1) = (s,t)$. Das ist aber auch die Bedingung für $(s,t) \in L_R^{i+1,j}$, also $R_R^{i,j} \subset L_R^{i+1,j}$, die andere Inklusion und damit Gleichheit folgt symmetrisch. Das gleiche Argument zeigt auch die Gleichheit $T_R^{i,j} = B_R^{i,j+1}$. \square

Daran, dass nur der Abstand der durch das Parameterpaar $(0,0)$ gegebenen Punkte $T_1(0), T_2(0)$ groß genug sein muss, damit es keinen solchen Weg ξ mehr gibt, wird deutlich, dass der erreichbare Raum im Allgemeinen nicht mit dem Freespace übereinstimmt (siehe Abbildung 2.13). Es gibt weitere Situationen, in denen der erreichbare Raum eine nichttriviale Teilmenge des Freespace ist. Mehr dazu in Abschnitt 2.2.5 (Berechnung der Fréchet-Distanz).

Mit dem erreichbaren Raum lässt sich das Fréchet-Distanz-Entscheidungsproblem mathematisch formulieren als die Frage, ob zu einem gegebenen δ die rechte obere Ecke des resultierenden Freespace-Diagramms erreichbar ist: $(n,m) \in C_R^{n,m}$.

Zur Beantwortung dieser Frage ist die explizite Berechnung des gesamten erreichbaren Raumes allerdings überhaupt nicht notwendig, da der Endpunkt auf einem Randstück liegt und sich der erreichbare Raum eingeschränkt auf die Randstücke induktiv berechnen lässt ([AG92, Seite 79f.]). Diese in [AG92] nicht bewiesene Aussage führt direkt zu einem Algorithmus zur Berechnung der erreichbaren Zellenrandstücke und lässt sich in der hier verwendeten Notation so ausdrücken:

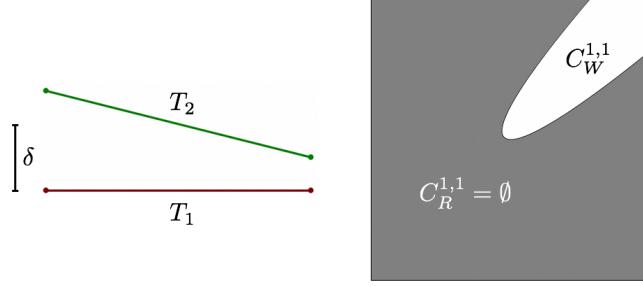


Abbildung 2.13: Ein einfaches Beispiel für Freespace, der nicht erreichbar ist. Die zwei Linien T_1 und T_2 , jeweils bestehend aus nur einem Segment, sind zu Beginn zu weit voneinander entfernt als dass die Parameterecke $(0, 0)$ im Freespace liegen könnte: $d(T_1(0), T_2(0)) > \delta > d(T_1(1), T_2(1))$. Dadurch kann es keinen Weg $\xi : I \rightarrow W$ mit $\xi(0) = (0, 0)$ und damit auch keine erreichbaren Punkte geben.

Lemma 2.16 (Induktive Berechnung der erreichbaren Zellenrandstücke)

Zu zwei polygonalen Kurven T_1 der Länge n und T_2 der Länge m seien die zugehörigen Freespace-Zellen $(C^{i,j})_{i,j}$ mit Freespace $(C_W^{i,j})_{i,j}$ gegeben. Durch Anwendung der folgenden Regeln können die erreichbaren Zellenrandstücke induktiv errechnet werden:

- (a) Falls $(0, 0) \notin C_W^{1,1}$ gilt, ist kein einziger Punkt des Freespace-Diagramms erreichbar, damit sind alle erreichbaren Zellenrandstücke leer. Ist $(0, 0) \in C_W^{1,1}$ frei, so auch erreichbar.
- (b) Für die Zellen am linken/unteren Rand des Diagramms $C^{1,j}$ für $j = 1, \dots, m$ bzw. $C^{i,1}$ für $i = 1, \dots, n$ ermittelt man die linken/unteren erreichbaren Zellenrandstücke so:

$$L_R^{1,j} = \begin{cases} L_W^{1,j} & \text{falls } (0, j-1) \text{ erreichbar} \\ \emptyset & \text{sonst} \end{cases}$$

$$B_R^{i,1} = \begin{cases} B_W^{i,1} & \text{falls } (i-1, 0) \text{ erreichbar} \\ \emptyset & \text{sonst} \end{cases}$$

- (c) Induktive Berechnung der erreichbaren rechten und oberen erreichbaren Zellenrandstücke von $C^{i,j}$: Sind die folgenden Zellenrandstücke bekannt

$$\begin{aligned} L_R^{i,j} &= \{i-1\} \times [l_b, l_t] \\ B_R^{i,j} &= [b_l, b_r] \times \{j-1\} \\ R_W^{i,j} &= \{i\} \times [r_b, r_t] \\ T_W^{i,j} &= [t_l, t_r] \times \{j\}, \end{aligned}$$

so gilt:

$$R_R^{i,j} = \begin{cases} R_W^{i,j} & \text{falls } B_R^{i,j} \neq \emptyset \\ \emptyset & \text{falls } B_R^{i,j} = \emptyset \text{ und } l_b > r_t \\ \{i\} \times [\max(l_b, r_b), r_t] & \text{falls } B_R^{i,j} = \emptyset \text{ und } l_b \leq r_t \end{cases}$$

$$T_R^{i,j} = \begin{cases} T_W^{i,j} & \text{falls } L_R^{i,j} \neq \emptyset \\ \emptyset & \text{falls } L_R^{i,j} = \emptyset \text{ und } b_l > t_r \\ [\max(b_l, t_l), t_r] \times \{j\} & \text{falls } L_R^{i,j} = \emptyset \text{ und } b_l \leq t_r . \end{cases}$$

Beweis.

- (a) Falls der linke untere Punkt des Freespace-Diagramms nicht frei ist, so kann auch kein freier Weg dort beginnen, der in irgendwelchen erreichbaren Punkten enden könnte (vergleiche die Argumentation direkt nach Definition 2.14 und Abbildung 2.13). Ist $(0, 0)$ frei, so trivialerweise auch erreichbar (durch den konstanten Weg $\xi : I \rightarrow \{(0, 0)\}$).
- (b) Ist $(0, j-1)$ frei, so ist $(0, j-1)$ im freien Randstück $L_W^{1,j}$ enthalten, da die freien Zellenrandstücke konvex und damit zusammenhängend sind. Es gilt also $L_W^{1,j} = \{0\} \times [j-1, l_t]$ für ein $l_t \geq j-1$. Aus der zusätzlichen Erreichbarkeit von $(0, j-1)$ folgt die Existenz eines (monotonen) Weges

$$\xi : I \rightarrow \bigcup_{k=1, \dots, j-1} L_W^{1,k}, \quad \text{mit } \xi(0) = (0, 0) \text{ und } \xi(1) = (0, j-1)$$

Da $(0, j-1) \in L_W^{1,j}$ frei ist, gibt es für jeden Punkt $(0, l) \in L_W^{1,j}$ den (monotonen) Weg

$$\zeta : I \rightarrow L_W^{1,j}, \quad t \mapsto (0, j-1 + t(l - (j-1)))$$

so dass $\zeta(0) = (0, j-1)$ und $\zeta(1) = (0, l)$ gilt. Dann ist aber die Verknüpfung der beiden Wege ξ und ζ :

$$I \rightarrow \bigcup_{k=1, \dots, j} L_W^{1,k}, \quad t \mapsto \begin{cases} \xi(2t) & \text{falls } t \in [0, \frac{1}{2}] \\ \zeta(2t-1) & \text{falls } t \in [\frac{1}{2}, 1] \end{cases}$$

wohldefiniert und ein monotoner Weg von $(0, 0)$ nach $(0, l)$, womit $(0, l)$ erreichbar ist. Da $(0, l) \in L_W^{1,j}$ beliebig gewählt wurde, folgt $L_W^{1,j} = L_R^{1,j}$.

Sei nun umgekehrt $(0, j-1)$ nicht erreichbar. Angenommen, es gäbe einen erreichbaren Punkt $(0, l) \in L_W^{1,j}$, dann gäbe es auch einen (monotonen) Weg

$$\xi : I \rightarrow \bigcup_{k=1, \dots, j} L_W^{1,k} \quad \text{mit } \xi(0) = (0, 0) \text{ und } \xi(1) = (0, l).$$

Dann ist die Projektion dieses Weges auf die zweite Koordinate $\pi_2 \circ \xi$ eine stetige Funktion mit $(\pi_2 \circ \xi)(0) = 0$ und $(\pi_2 \circ \xi)(1) = l$. Mit dem Zwischenwertsatz folgt die Existenz eines $t \in I$ mit $(\pi_2 \circ \xi)(t) = j-1$. Nach Konstruktion und wegen der Monotonie von ξ gilt also $\xi(t) = (0, j-1)$. Dann aber widerlegt der (monotone) Weg

$$I \rightarrow \bigcup_{k=1, \dots, j-1} L_W^{1,k}, \quad s \mapsto \xi(st)$$

von $(0, 0)$ nach $(0, j-1)$ die Nichterreichbarkeit von $(0, j-1)$. Einen erreichbaren Punkt $(0, l) \in L_W^{1,j}$ kann es also nicht geben, woraus $L_R^{1,j} = \emptyset$ folgt.

Die Aussagen für die Erreichbarkeit der Randstücke am unteren Rand zeigt man analog.

- (c) Sei $p_1 \in B_R^{i,j} \neq \emptyset$, dann existiert ein freier in beide Richtungen monotoner Weg

$$\xi : I \rightarrow W \text{ mit } \xi(0) = (0, 0) \text{ und } \xi(1) = p_1$$

Sei außerdem $p_2 \in R_W^{i,j}$ ein Punkt auf dem freien rechten Zellenrandstück. Aufgrund der Konvexität von $C_W^{i,j}$ (Korollar 2.11) verläuft der (lineare und in beide Richtungen monotone) Weg

$$\zeta : I \rightarrow C_W^{i,j}, \quad t \mapsto p_1 + t(p_2 - p_1)$$

(siehe Abbildung 2.14) komplett im Freespace $C_W^{i,j}$ der Zelle. Die Verknüpfung

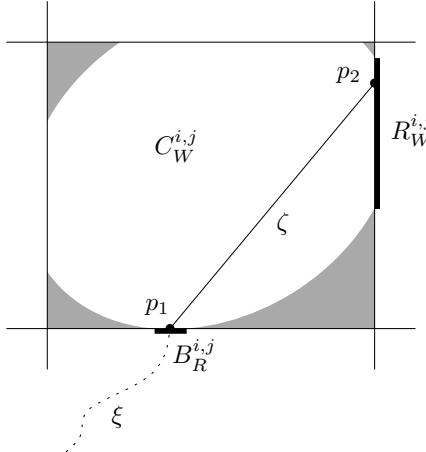


Abbildung 2.14: Ist ein Punkt $p_1 \in B_R^{i,j}$ mit dem Weg ξ erreichbar, so ist auch jeder Punkt $p_2 \in R_W^{i,j}$ über die Verknüpfung von ξ mit der linearen Verbindung von p_1 und p_2 erreichbar.

der beiden Wege ξ und ζ

$$I \rightarrow W, \quad t \mapsto \begin{cases} \xi(2t) & \text{falls } t \in [0, \frac{1}{2}] \\ \zeta(2t - 1) & \text{falls } t \in [\frac{1}{2}, 1] \end{cases}$$

ist dann ebenfalls wohldefiniert, frei und in beide Richtungen monoton und zeigt damit die Erreichbarkeit von p_2 . Da $p_2 \in R_W^{i,j}$ frei gewählt werden kann, folgt $R_R^{i,j} = R_W^{i,j}$.

Sei nun $B_R^{i,j} = \emptyset$ und $l_b > r_t$. Angenommen, es gäbe einen erreichbaren Punkt $p \in R_R^{i,j} \neq \emptyset$, dann folgt die Existenz eines in beide Richtungen monotonen Weges $\xi : I \rightarrow W$ mit $\xi(0) = (0, 0)$ und $\xi(1) = p$. Da es auf dem unteren Zellenrand keine erreichbaren Punkte gibt, muss mit einem Zwischenwertsatzargument ein Zeitpunkt t existieren, zu dem $\xi(t) \in L_R^{i,j}$, es gilt also insbesondere $\pi_2(\xi(t)) \geq l_b$. Gleichzeitig gilt $\pi_2(\xi(1)) \leq r_t$, zusammen also

$$\pi_2(\xi(t)) \geq l_b > r_t \geq \pi_2(\xi(1)) \quad \text{obwohl } t < 1$$

Also ist $\pi_2 \circ \xi$ nicht monoton, also auch ξ nicht in vertikaler Richtung im Widerspruch zu der Annahme, p sei via ξ erreichbar. Da $p \in R_R^{i,j}$ beliebig gewählt war, gilt $R_R^{i,j} = \emptyset$ (siehe Abbildung 2.15).

Sei zuletzt $B_R^{i,j} = \emptyset$, $l_b \leq r_t$ und $p_2 \in R_W^{i,j}$ ein freier Punkt auf dem rechten Zellenrand. Sei $p_1 = (i-1, l_b) \in L_R^{i,j}$ der unterste (durch einen Weg ξ) erreichbare Punkt auf dem linken Zellenrand. Dann existiert ein in beide Richtungen

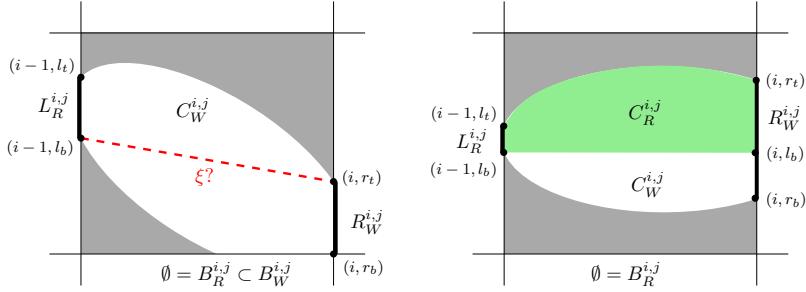


Abbildung 2.15: Ist $B_R^{i,j} = \emptyset$ und beginnt das linke erreichbare Zellenrandstück $L_R^{i,j}$ in vertikaler Richtung weiter oben, als das rechte freie Zellenrandstück $R_W^{i,j}$ endet, so ist keine in vertikaler Richtung monotope Verbindung zweier Punkte dieser beiden Stücke möglich. Daher ist kein Punkt von $R_W^{i,j}$ erreichbar (links). Überlappen sich hingegen die Höhen des linken erreichbaren Zellenrandstücks $L_R^{i,j}$ und des rechten freien Zellenrandstücks $R_W^{i,j}$, so sind auf letzterem nur die Punkte nicht erreichbar, die tiefer liegen als der unterste Punkt von $L_R^{i,j}$ (rechts).

monotoner Weg $\zeta : I \rightarrow C_W^{i,j}$ von $p_1 = \zeta(0)$ nach $p_2 = \zeta(1)$ genau dann, wenn $\pi_2(p_2) \geq \pi_2(p_1) = l_b$: ist dies der Fall, so ist $t \mapsto p_1 + t(p_2 - p_1)$ ein geeigneter Weg ζ . Ist das nicht der Fall, so ist p_2 nicht in vertikaler Richtung monoton von p_1 aus zu erreichen (Abbildung 2.15).

Existiert so ein Weg ζ , so ist p_2 durch die Verknüpfung von ξ und ζ erreichbar. Aufgrund der Wahl von p_1 war $\pi_2(p_1) = l_b$ die kleinste Wahl für die Höhe erreichbarer Punkte in $C_W^{i,j}$ überhaupt und es gibt keine erreichbaren Punkte p auf dem rechten Zellenrandstück mit $\pi_2(p) < l_b$. Da $R_R^{i,j} \subset R_W^{i,j} = \{i\} \times [r_b, r_t]$ gilt insgesamt

$$R_R^{i,j} = \{p \in R_W^{i,j} \mid \pi_2(p) \geq l_b\} = \{i\} \times [\max(l_b, r_b), r_t]$$

Der Beweis für das obere erreichbare Zellenrandstück wird analog geführt.

□

Wie bereits angekündigt, lässt sich aus den Aussagen dieses Lemma eine Konstruktionsvorschrift ableiten (vgl. [AG92, Algorithm 1, Seite 80]):

Algorithmus zur Lösung des Fréchet-Distanz-Entscheidungsproblems

Zu zwei polygonalen Kurven $T_1 : [0, n] \rightarrow \mathbb{R}^2$ und $T_2 : [0, m] \rightarrow \mathbb{R}^2$ existieren genau dann Parametrisierungen $\alpha : I \rightarrow [0, n]$ von T_1 und $\beta : I \rightarrow [0, m]$ von T_2 mit $d(T_1(\alpha(t)), T_2(\beta(t))) \leq \delta$ für alle $t \in I$, wenn $(n, m) \in R_R^{n,m} \subset C_R^{n,m}$ liegt. Diese erreichbaren Zellenrandstücke errechnet man so:

- Ist $d(T_1(0), T_2(0)) > \delta$, so ist $(0, 0)$ im Freespace-Diagramm nicht frei und damit kein einziger Punkt des Diagramms erreichbar – die oben genannten Parametrisierungen können nicht existieren.
- Für die Zellen am linken Rand $C^{1,j}$ für $j = 1, \dots, m$ bzw. am unteren Rand des Diagramms $C^{i,1}$ für $i = 1, \dots, n$ erhält man die linken bzw. unteren erreichbaren Zellenrandstücke als die entsprechenden freien Zellenrandstücke, falls der linke untere Punkt $(0, j-1)$ bzw. $(i-1, 0)$ erreichbar ist.

- Ausgehend von der linken unteren Ecke kann man für Indizes $i = 1, \dots, n$ und $j = 1, \dots, m$ die erreichbaren Zellenrandstücke von $C^{i,j}$ schrittweise errechnen: Sind die linken und unteren erreichbaren Zellenrandstücke $L_R^{i,j}, B_R^{i,j}$ sowie die rechten und oberen freien Zellenrandstücke $R_W^{i,j}, T_W^{i,j}$ bekannt, ermittelt man die rechten und oberen erreichbaren Zellenrandstücke wie folgt:

$$R_R^{i,j} = \begin{cases} R_W^{i,j} & \text{falls } B_R^{i,j} \neq \emptyset \\ \emptyset & \text{falls } B_R^{i,j} = \emptyset \text{ und } l_b > r_t \\ \{i\} \times [\max(l_b, r_b), r_t] & \text{falls } B_R^{i,j} = \emptyset \text{ und } l_b \leq r_t \end{cases}$$

$$T_R^{i,j} = \begin{cases} T_W^{i,j} & \text{falls } L_R^{i,j} \neq \emptyset \\ \emptyset & \text{falls } L_R^{i,j} = \emptyset \text{ und } b_l > t_r \\ [\max(b_l, t_l), t_r] \times \{j\} & \text{falls } L_R^{i,j} = \emptyset \text{ und } b_l \leq t_r \end{cases}$$

2.2.5 Exakte Berechnung der Fréchet-Distanz

Das oben diskutierte Entscheidungsproblem beantwortet die Frage, ob die Fréchet-Distanz $\delta_F(T_1, T_2)$ zweier polygonaler Kurven kleiner oder gleich einem gegebenen maximalen punktweisen Abstand δ ist. Um die Fréchet-Distanz selbst zu ermitteln, muss der kleinste solche Wert δ gefunden, für das das Entscheidungsproblem noch positiv beantwortet wird.

Nach [AG92, Seite 80] gibt es nur eine begrenzte Anzahl von Distanzen δ , bei denen im zugehörigen Freespace-Diagramm für die zwei Kurven überhaupt ein Wechsel der Erreichbarkeit der rechten oberen Ecke stattfinden kann. Es sind dies die folgenden drei Kategorien sogenannter *kritischer Werte* für δ :

- (a) δ ist minimal mit $(0, 0) \in C_W^{1,1}$ und $(n, m) \in C_W^{n,m}$ (Freiheit von Start- und Endpunkt, Abbildung 2.16).
- (b) δ ist minimal, so dass das freie Zellenrandstück $L_W^{i,j}$ oder $B_W^{i,j}$ nicht leer ist für eine Zelle $C^{i,j}$ (Übergang zwischen benachbarten Zellen, Abbildung 2.16).
- (c) δ ist minimal so dass
 - für zwei Zellen $C^{i_1,j}, C^{i_2,j}$ mit $i_1 < i_2$ und $L_W^{i_1,j} = \{i_1 - 1\} \times [l_b, l_t]$ sowie $L_W^{i_2,j} = \{i_2 - 1\} \times [r_b, r_t]$ gilt: $l_b = r_t$ (Horizontale zellenübergreifende Passage, Abbildung 2.17) oder
 - für zwei Zellen C^{i,j_1}, C^{i,j_2} mit $j_1 < j_2$ und $B_W^{i,j_1} = [b_l, b_r] \times \{j_1 - 1\}$ sowie $B_W^{i,j_2} = [t_l, t_r] \times \{j_2 - 1\}$ gilt: $b_l = t_r$ (Vertikale zellenübergreifende Passage).

Da die daraus resultierenden Werte für δ nach [AG92, Seite 80] die einzigen kritischen Werte sind, die als Realisierung der Fréchet-Distanz in Frage kommen, genügt es, für diese Kandidaten den kleinsten Wert zu finden, für den das Entscheidungsproblem positiv gelöst wird. Zunächst wird aber diskutiert, wie diese kritischen Werte konkret ermittelt werden können.

Ermittlung kritischer Werte des Freespace-Diagramms

Dazu müssen die kritischen Werte im Freespace-Diagramm zunächst geometrisch im Kontext der zugrundeliegenden polygonalen Kurven interpretiert werden:

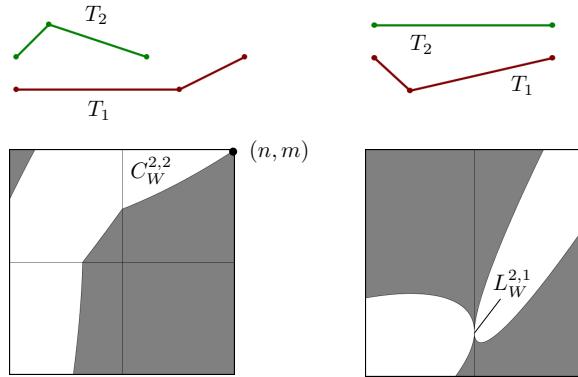


Abbildung 2.16: Kritische Werte von δ , die das Fréchet-Distanz-Entscheidungsproblem beeinflussen. Links: Typ (a), δ ist minimal, so dass der Endpunkt (n, m) für freie Wege durch das Freespace-Diagramm frei ist. Rechts: Typ (b), δ ist minimal, so dass ein freier Übergang zweier benachbarter Zellen existiert.

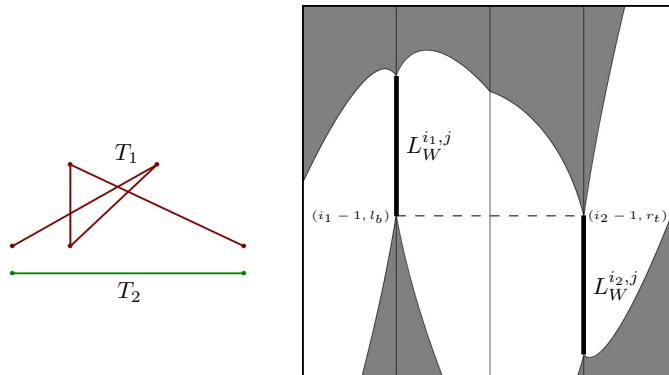


Abbildung 2.17: Kritischer Wert von δ , der das Fréchet-Distanz-Entscheidungsproblem beeinflusst, Typ (c): δ ist minimal, so dass gerade eben noch eine horizontale Passage zwischen den freien Zellenrandstücken $L_W^{i_1,j}, L_W^{i_2,j}$ existiert.

- (a) Das minimale δ , das gerade eben noch Start- und Endpunkt freier Wege im Freespace-Diagramm frei lässt, ist gerade das Maximum der Distanzen zwischen den Start- und Endpunkten der zugrundeliegenden Kurven T_1 und T_2 .
- (b) Die minimale Distanz δ , die gerade eben einen freien Übergang zwischen horizontalen Nachbarzellen $C^{i,j}$ und $C^{i+1,j}$ zulässt, wird realisiert durch den Abstand des Punktes $T_1(i)$ vom Segment $T_2|_{[j-1,j]}$, falls das Lot des Punktes auf die durch das Segment induzierte Gerade auch auf dem Segment selbst liegt. Das wird deutlich, wenn man sich klarmacht, dass das Zellenrandstück $L^{i+1,j}$, das gerade zwischen den beiden Zellen liegt, genau die Punktpaarmenge $\{(T_1(x), T_2(y)) \mid x = i, y \in [j-1, j]\}$ im Datenraum definiert. Den Fall für vertikaler Nachbarzellen überlegt man sich analog.
- (c) Die geometrische Interpretation der minimalen Distanz δ für eine freie (monotone) zellenübergreifende horizontale Passage zwischen Zellenrandstücken $L^{i_1,j}$ und $L^{i_2,j}$ ist nicht besonders intuitiv. Zunächst fixieren die beiden Zel-

lenrandstücke zwei Punkte $a := T_1(i_1 - 1)$ und $b := T_1(i_2 - 1)$. Das korrespondierende Segment auf T_2 ist durch die Zeile im Diagramm gegeben: $T_2|_{[j-1,j]}$. Die zu findende minimal monotone Passage ist dann horizontal und definiert einen Punkt auf dem vorgenannten Segment. Da die Endpunkte der Passage zwischen den beiden Zellenrandstücken genau auf dem Rand des Free-space liegen, realisieren auch die durch die Schnittpunkte der Passage mit den Zellenrandstücken jeweils gegebenen Punktpaare auf T_1 bzw. T_2 genau die gesuchte Distanz δ . Gesucht ist also ein Punkt c auf $T_2|_{[j-1,j]}$, der zu a und b dieselbe (minimale) Distanz aufweist (Abbildung 2.18).

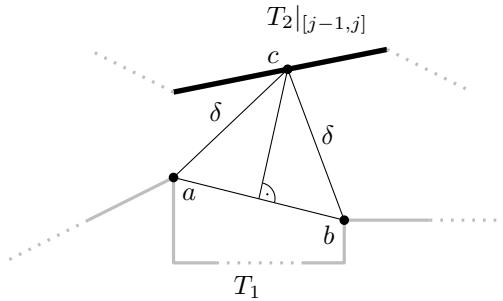


Abbildung 2.18: Der als Typ (c) klassifizierte kritische Wert für horizontale zellenübergreifende Passagen δ wird im Datenraum der polygonalen Kurven T_1, T_2 realisiert als die minimal gleiche Distanz der durch die Spalten gegebenen Punkte a, b zu einem Punkt c auf dem durch die Zeile gegebenen Segment $T_2|_{[j-1,j]}$.

Das legt als Konstruktionsvorschrift nahe, zunächst durch den Mittelpunkt von a und b eine zu ihrer Verbindungslinie orthogonale Linie zu konstruieren. Die Punkte dieser Linie enthalten alle die Punkte der Ebene, die zu den beiden Punkten a und b den selben Abstand haben, c muss also auf dieser Linie liegen. Der Schnittpunkt dieser Linie mit dem durch $T_2|_{[j-1,j]}$ gegebenen Segment ist dann der gesuchte Punkt c . Falls kein Schnittpunkt existiert, gibt es auch keinen kritischen Wert δ , für den es eine Passage geben könnte. Falls die Linie mit dem Segment zusammenfällt, so ist die kleinste Distanz δ natürlich die kleinste gemeinsame Distanz zu dem Segment, also $d(a, b)/2$.

Durch die Beschränkung der in Frage kommenden Werte δ , die die Fréchet-Distanz zweier polygonaler Kurven T_1, T_2 realisieren können, erhält man nach [AG92, Algorithm 2, Seite 81] eine einfache Vorschrift, die exakte Fréchet-Distanz zu ermitteln:

Algorithmus zur exakten Berechnung der Fréchet-Distanz von T_1 und T_2

1. Berechne alle in Frage kommenden kritischen Werte δ der Typen (a), (b) und (c).
2. Organisiere diese Kandidatenwerte in einer sortierten Liste.
3. Eine binäre Suche nach dem kleinsten Kandidaten δ , für den das Fréchet-Distanz-Entscheidungsproblem positiv ist, liefert die Fréchet-Distanz $\delta_F(T_1, T_2)$ der Kurven.

Nach ausführlicher Diskussion der genauen Gestalt des Freespace-Diagramms und der Algorithmen zur Lösung des Entscheidungsproblems sowie zur Ermittlung der

Fréchet-Distanz selbst, wird im Folgenden noch eine flexible Variante, die partielle Fréchet-Distanz vorgestellt, bevor in Kapitel 3 eine lokalisierte Version diskutiert wird, die den Besonderheiten in der Analyse astronomischer Spektren besonders gut anpassbar erscheint.

2.3 Die partielle Fréchet-Distanz

Gerade im Kontext astronomischer Spektren als Datenquelle für polygonale Kurven ist es sinnvoll, die Distanzberechnung unabhängig von Ausreißern zu machen, die z. B. durch automatische aber nicht ganz präzise Korrektur von sogenannten Himmelslinien entstehen können:

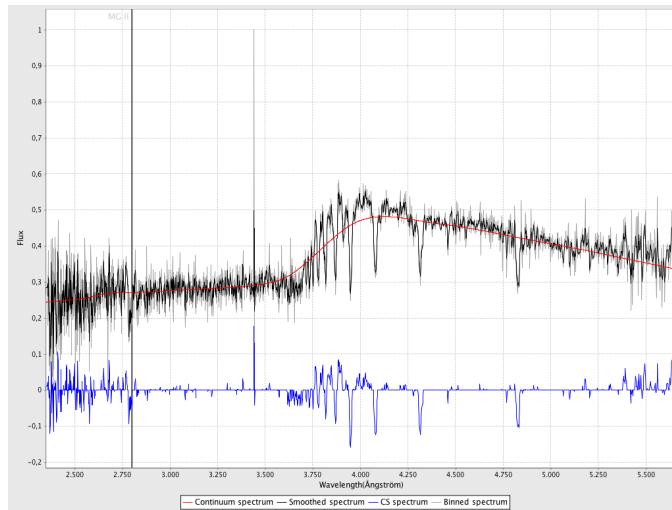


Abbildung 2.19: Eine sternbildende Galaxie mit typischem Verlauf der Strahlungsintensität über die beobachteten Wellenlängen, aber deutlich sichtbaren großen Ausreißern in den Daten bei ca. 2800 und 3450Å.

Eine einfache Möglichkeit dazu ist, die Kurve entsprechend zu glätten. In der hier verwendeten Arbeitsumgebung stehen dazu verschiedene Verfahren zur Verfügung, die in Kapitel 4 kurz vorgestellt werden. Wünschenswert wäre aber auch eine Variante der Fréchet-Distanz, die sich gegenüber vereinzelten Ausreißern robuster verhält als die Standard-Fréchet-Distanz. Ausreißer wie in Abbildung 2.19, die in einer der beiden polygonalen Kurven auftreten, verursachen Distanzen etwa in Höhe der Intensität des Ausreißers. Der weitere Verlauf der Kurven hat dabei keinen Einfluss mehr auf die Berechnung.

Ein Ansatz dazu, die *partielle Fréchet-Distanz*, wird in [dCGM⁺13] diskutiert und hier vorgestellt. Wie bereits in 2.2 angedeutet, geht es dabei darum, zu gegebener maximaler punktweiser Distanz δ die Weglängen auf den beiden Kurven zu minimieren, auf denen diese maximale Distanz überschritten wird. Dazu zunächst eine kleine Erinnerung:

Beim Entscheidungsproblem zur Standard-Fréchet-Distanz wird über die Parameterpaare $\alpha : I \rightarrow [0, n]$, $\beta : I \rightarrow [0, m]$ der Kurven T_1 der Länge n und T_2 der Länge m variiert, bis zu jedem Zeitpunkt $t \in I$ der Abstand $d(T_1(\alpha(t)), T_2(\beta(t)))$ kleiner oder gleich einem gegebenen maximalen Abstand δ ist. Ist das nicht möglich, so wird das Entscheidungsproblem negativ beantwortet, mit anderen Worten: in der gegebenen Situation ist die Fréchet-Distanz $\delta_F(T_1, T_2) > \delta$.

Im Freespace-Diagramm wird durch das Parametrisierungs paar eine (in beide Richtungen monotone) Kurve

$$\xi : I^2 \rightarrow [0, n] \times [0, m], \quad (s, t) \mapsto (\alpha(s), \beta(t))$$

von der linken unteren Ecke $(0, 0)$ in die rechte obere Ecke (n, m) induziert, die komplett im Freespace verlaufen soll. Gesucht ist also ein freier in beide Richtungen monotoner Weg zwischen diesen beiden Punkten. Ist ein solcher Weg zu gegebenem δ nicht möglich, so ist die partielle Fréchet-Distanz trotzdem noch eine Lösung, die auf intuitiv nachvollziehbare Weise als optimal bezeichnet werden kann.

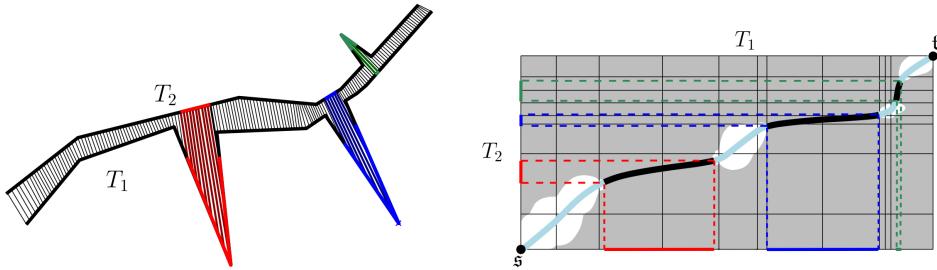


Abbildung 2.20: Zwei polygonale Kurven T_1 und T_2 , die Ausreißer aufweisen. Durch die feinen Linien sind durch das Parametrisierungs paar zusammengehörige Punkte auf den Kurven verbunden. Wo sie farbig sind, ist die punktweise Distanz δ zu groß (links). Im Freespace-Diagramm (rechts) verläuft die Kurve von $s := (0, 0)$ nach $t := (n, m)$ entsprechend nicht vollständig im Freespace. Wie die Länge der im verbotenen Teil verlaufenden Abschnitte in die Berechnung eingeht, wird im Text diskutiert. Bildquelle: [dCGM⁺13, Fig. 1.]

2.3.1 Qualitätsmaße für die partielle Fréchet-Distanz

Um eine präzise Vorstellung des gewünschten Ziels der folgenden Diskussion zu erlangen, werden in [dCGM⁺13] zunächst zwei (äquivalente) Probleme definiert:

- Das *MinEx-Problem* (Min-Exclusion) bezeichnet die Suche nach einem Parametrisierungspaar, für das die Weglänge auf den Kurven, wo eine längere als die mit δ vorgegebene punktweise Distanz benötigt wird, minimal ist.
- Das *MaxIn-Problem* (Max-Inclusion) umgekehrt bezeichnet die Suche nach einem Parametrisierungspaar, für das die Weglänge auf den Kurven, wo die vorgegebene punktweise Distanz δ ausreicht, maximal ist.

In formaler Schreibweise kann man zunächst zu gegebenen Parametrisierungen α von T_1 und β von T_2 die Mengen $\mathcal{B}_{\alpha, \beta}, \mathcal{W}_{\alpha, \beta} \subset I$ definieren als den Abschluss der Zeitpunkte $t \in I$, zu denen die durch α, β induzierte Kurve ξ im Freespace-Diagramm im verbotenen bzw. freien Raum verläuft:

$$\begin{aligned}\mathcal{B}_{\alpha, \beta} &:= \overline{\{t \in I \mid d(T_1(\alpha(t)), T_2(\beta(t))) > \delta\}} \\ \mathcal{W}_{\alpha, \beta} &:= \overline{\{t \in I \mid d(T_1(\alpha(t)), T_2(\beta(t))) \leq \delta\}}\end{aligned}$$

Dadurch kann man die durch die vorgenannten Probleme definierten Weglängen auf den Kurven T_1, T_2 als Kurvenintegrale über diesen Mengen definieren und erhält die

Qualitätsmaße für Parametrisierungspaare

$$Q_{\alpha,\beta}^B := \int_{t \in \mathcal{B}_{\alpha,\beta}} \|(T_1 \circ \alpha)'(t)\| dt + \int_{t \in \mathcal{B}_{\alpha,\beta}} \|(T_2 \circ \beta)'(t)\| dt,$$

$$Q_{\alpha,\beta}^W := \int_{t \in \mathcal{W}_{\alpha,\beta}} \|(T_1 \circ \alpha)'(t)\| dt + \int_{t \in \mathcal{W}_{\alpha,\beta}} \|(T_2 \circ \beta)'(t)\| dt,$$

wobei mit $\|\cdot\|$ hier die Euklidische Norm gemeint ist. Im Sinne der oben genannten Probleme wird ein Parametrisierungspaar (α, β) in [dCGM⁺13, 1.2] *optimal* genannt, falls es $Q_{\alpha,\beta}^B$ minimiert (und $Q_{\alpha,\beta}^W$ maximiert), was zur Definition der Qualität von T_1 und T_2 bezüglich δ führt:

$$Q^B(T_1, T_2) := \inf_{\alpha, \beta: I \rightarrow I} Q_{\alpha,\beta}^B$$

$$Q^W(T_1, T_2) := \sup_{\alpha, \beta: I \rightarrow I} Q_{\alpha,\beta}^W$$

In [dCGM⁺13, 2] wird gezeigt, dass diese Probleme nicht exakt berechenbar sind. Um jedoch gute Näherungen berechnen zu können, wird das Problem zunächst in ein Problem im Freespace-Diagramm überführt.

2.3.2 Das deformierte Freespace-Diagramm

Im Gegensatz zum Freespace-Diagramm für die Standard-Fréchet-Distanz wird hier eine deformierte Version davon eingesetzt, bei der die Längen und Breiten der Zellen direkt mit den Längen der korrespondierenden Segmente der zugrundeliegenden polygonalen Kurven übereinstimmen. Es sei zunächst eine Notation für die Länge eines Kurvenabschnitts vom Beginn an gemessen definiert:

$$l_i(T) := \sum_{k=1}^i d(T(k), T(k-1))$$

Damit lassen sich die deformierten Freespace-Zellen für T_1, T_2 so angeben:

$$C^{i,j} = [l_{i-1}(T_1), l_i(T_1)] \times [l_{j-1}(T_2), l_j(T_2)]$$

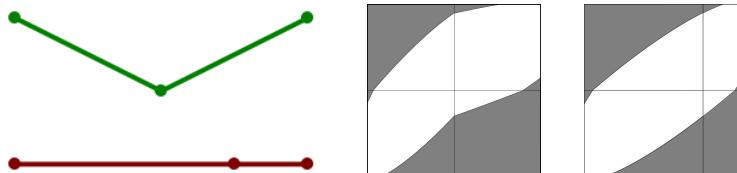


Abbildung 2.21: Die Deformierung des Freespace. Im linken Teil sind zwei Kurven sichtbar. Die untere hat verschiedene lange Segmente. Während das Standard-Fréchet-Distanz-Freespace-Diagramm für diese Situation in gleichgroße Parameterzellen aufgeteilt ist (mitte), weist das deformierte Diagramm eine den Längen der Segmente entsprechende Verzerrung auf. So ist hier die erste Spalte (entsprechend dem ersten Segment der unteren Kurve) dreimal so breit wie die zweite Spalte.

Die auf diese Weise deformierten Freespace-Diagramme haben dann die nützliche Eigenschaft, dass die Länge der Abschnitte im verbotenen (bzw. freien) Teil mit der Summe der Längen der entsprechenden Abschnitte auf den polygonalen Kurven übereinstimmen ([dCGM⁺13, Observation 1], vergleiche auch Abbildung 2.20):

Beobachtung 2.17 Seien zu einem maximalen punktweisen Abstand δ zwei Kurven $T_1 : [0, n] \rightarrow \mathbb{R}^2$ und $T_2 : [0, m] \rightarrow \mathbb{R}^2$ zusammen mit ihren Parametrisierungen $\alpha : I \rightarrow [0, n]$ und $\beta : I \rightarrow [0, m]$ gegeben. Der dadurch induzierte Pfad $\pi_{\mathfrak{s}\mathfrak{t}}$ im Freespace-Diagramm von $\mathfrak{s} := (0, 0)$ nach $\mathfrak{t} := (l_n(T_1), l_m(T_2))$ hat dann folgende Eigenschaft:

$$\int_{t \in \mathcal{B}_{\alpha, \beta}} \|\pi'_{\mathfrak{s}\mathfrak{t}}(t)\|_1 dt = Q_{\alpha, \beta}^B \quad \text{und} \quad \int_{t \in \mathcal{W}_{\alpha, \beta}} \|\pi'_{\mathfrak{s}\mathfrak{t}}(t)\|_1 dt = Q_{\alpha, \beta}^W.$$

Dazu sei speziell darauf hingewiesen, dass die Länge im Freespace-Diagramm mit der L_1 -Metrik, Längen im Datenraum jedoch mit der L_2 -Metrik gemessen werden. Auf der Grundlage dieser Beobachtung lassen sich die oben definierten Probleme MinEx und MaxIn im Kontext des Freespace-Diagramms so formulieren ([dCGM⁺13, Seite 628]):

- *Gewichteter kürzester xy-monotoner Pfad (wShortMP) Problem:* Berechne einen xy -monotonen gewichteten kürzesten Pfad von \mathfrak{s} nach \mathfrak{t} im Freespace-Diagramm, wobei das Gewicht im verbotenen Raum eins und das Gewicht im freien Raum null sei. Die Länge des Pfades ist dann definiert als die Summe der Längen (unter L_1 -Metrik gemessen) der Teilpfade durch den verbotenen Raum.
- *Gewichteter längster xy-monotoner Pfad (wLongMP) Problem:* Berechne einen xy -monotonen gewichteten längsten Pfad von \mathfrak{s} nach \mathfrak{t} im Freespace-Diagramm, wobei das Gewicht im verbotenen Raum null und das Gewicht im freien Raum eins sei. Die Länge des Pfades ist dann definiert als die Summe der Längen (unter L_1 -Metrik gemessen) der Teilpfade durch den freien Raum.

2.3.3 Ein Approximationsalgorithmus

Wie bereits angedeutet, wird in [dCGM⁺13, Abschnitt 2] gezeigt, dass eine exakte Lösung der genannten Probleme im *Algebraic Computational Model over the Rational Numbers* (ACM \mathbb{Q}) und damit mit herkömmlicher Rechnerarithmetik im Allgemeinen nicht möglich ist. Jedoch ermöglicht der in [dCGM⁺13, Abschnitt 3] präsentierte Algorithmus eine gute Approximation von Lösungen für folgende gegebene Daten:

- Zwei polygonale Kurven $T_1 : [0, n] \rightarrow \mathbb{R}^2$, $T_2 : [0, m] \rightarrow \mathbb{R}^2$,
- eine beliebige aber fixierte punktweise maximale Distanz δ und
- ein Approximationsparameter d .

Der Algorithmus berechnet ein Parametrisierungspaar $(\tilde{\alpha}, \tilde{\beta})$ und seine Qualität $Q_{\tilde{\alpha}\tilde{\beta}}^B$ derart, dass $Q_{\tilde{\alpha}\tilde{\beta}}^B$ eine gute Approximation von $Q^B(T_1, T_2)$ ist. Außerdem ist die Konstruktion zweier modifizierter polygonaler Kurven T'_1, T'_2 wünschenswert, so dass für diese beiden das Standard-Fréchet-Distanz-Entscheidungsproblem bezüglich maximaler punktweiser Distanz δ positiv gelöst werden kann.

Dazu wird in einem iterativen Verfahren ein gerichteter azyklischer Graph G im Freespace-Diagramm konstruiert, von dem gezeigt werden kann, dass jeder xy -monotone Pfad $\pi_{\mathfrak{s}\mathfrak{t}}$ im Freespace-Diagramm nah an einem Pfad $\tilde{\pi}_{\mathfrak{s}\mathfrak{t}}$ in diesem Graphen verläuft. Der Algorithmus geht dabei in den folgenden Schritten vor:

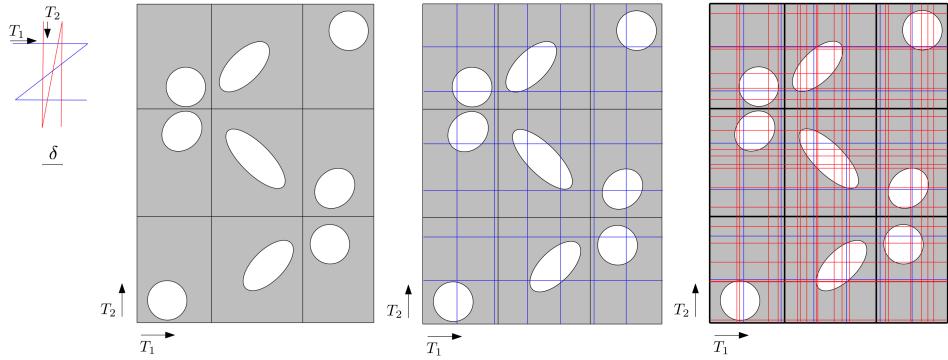


Abbildung 2.22: Konstruktion neuer Gitterlinien als Zwischenschritt bei der Approximation eines optimalen Pfades π_{st} durch das Freespace-Diagramm. Das Freespace-Diagramm wird entsprechend der Längen der jeweils zu den Zellen gehörigen Segmenten deformiert (Bild 1). Ge steuert durch den Parameter d werden dann n/d vertikale und m/d horizontale zusätzliche Gitterlinien eingefügt (Bild 2). An Schnittpunkten dieser Gitterlinien mit den Freespace-Ellipsen werden im nächsten Schritt weitere Gitterlinien eingefügt (Bild 3). Bildquelle (Notation angepasst): [dCGM⁺13, Fig. 3]

Schritt 1 Berechne mit den bekannten Methoden das Freespace-Diagramm für die Kurven T_1, T_2 und deforme es so, dass die Breiten der Spalten bzw. die Höhe der Zeilen den Längen der korrespondierenden Liniensegmente von T_1, T_2 entsprechen (siehe Abschnitt 2.3.2).

Schritt 2 Füge n/d vertikale und m/d horizontale äquidistante Gitterlinien zusätzlich in das Freespace-Diagramm ein und berechne die Schnittpunkte der neuen vertikalen Gitterlinien mit dem Rand des Freespaces innerhalb der Zellen. Füge für jeden dieser Schnittpunkte eine neue horizontale Schnittlinie durch diesen Punkt ein. Führe das gleiche Verfahren mit den horizontalen Gitterlinien durch. Dabei werden die durch die Zellengrenzen definierten Linien ebenfalls als Gitterlinien betrachtet.

Schritt 3 Berechne das *Arrangement* A gegeben durch alle Gitterlinien, die Schnittlinien und die Ränder des Freespace. Die Punkte des Graphen G sind die Punkte von A und es gibt eine gerichtete Kante von einem Punkt $p \in G$ zu einem Punkt $q \in G$ genau dann, wenn \overrightarrow{pq} eine xy -monotone Kante in A ist. Das Gewicht dieser Kante ist seine Länge gemessen in der L_1 -Metrik, falls sie im verbotenen Raum verläuft und null sonst.

Schritt 4 Berechne den gewichteten kürzesten Pfad $\tilde{\pi}_{st}$ von s nach t in G . Seine Darstellung im Freespace-Diagramm induziert die gesuchte Näherungslösung (Parametrisierungspaar $(\tilde{\alpha}, \tilde{\beta})$).

Zur übersichtlicheren Laufzeitbestimmung wird dabei angenommen, dass beide polygonale Kurven aus gleichvielen Punkten p bestehen. Sei $p := \max(n, m)$, dann kann durch Hinzufügen von Punkten ohne Richtungsänderung diese Tatsache bei der in Punkten kürzeren Kurve herbeigeführt werden. Als Laufzeit für diesen Algorithmus erhält man $O(p^4/d^2)$, wobei d der Approximationsparameter ist ([dCGM⁺13, Lemma 4]). Zur Laufzeitverbesserung wird in [dCGM⁺13, Abschnitt 4] noch eine weitere Modifikation vorgestellt.

Zu den im Rahmen der partiellen Fréchet-Distanz diskutierten Problem und Werkzeugen ist zu bemerken, dass es sich hierbei um eine Optimierung im Rahmen des Standard-Fréchet-Distanz-Entscheidungsproblems handelt, die dann zum Tragen kommt, wenn dieses negativ beantwortet wird. Dann nämlich wird versucht, durch Variation der Parametrisierungen möglichst kurze Wegstrecken mit einer längeren als der mit δ vorgegebenen Leine zurückzulegen, wobei die vorgegebenen Leinenlänge dabei festgehalten wird. Was in [dCGM⁺13] nicht diskutiert wird, ist das Problem, zu festgehaltenem maximalen Weglängenanteil im verbotenen Raum die minimale Leinenlänge zu finden. Da die beiden Wünsche (kurze Wegstrecke im verbotenen Raum, möglichst kurze Leine) sich gegenseitig entgegenstehen, sind hier bestenfalls Pareto-optimale Lösungen denkbar. Aufgrund der Abwesenheit einfach handhabbarer kritischer Werte wie bei der exakten Berechnung der Standard-Fréchet-Distanz in Abschnitt 2.2.5 macht dieses Problem einen nichttrivialen und interessanten Eindruck.

Kapitel 3

Eine lokalisierte Version der Fréchet-Distanz

Bei der Klassifikation von Spektren überprüft man die Nähe im Merkmalsraum bezüglich einer gewissen Auswahl von Merkmalen (siehe Kapitel 1). Um die Fréchet-Distanz von Spektren (aufgefasst als polygonale Kurven) untereinander in diesem Kontext einsetzen zu können, ist es sinnvoll, die Distanz zu einem repräsentativen Spektrum zu messen und diesen Wert als Merkmal zu verwenden. Es ist eine interessante Frage, wie man ein solches repräsentatives Spektrum auswählen kann. Zunächst sollen hier aber die Konsequenzen daraus diskutiert werden, dass die vorliegende Trainingsmenge bei einer bestimmten Klassifikations-Aufgabenstellung in der Astronomie eine unpraktische Eigenschaft aufweist: die als positiv vorklassifizierten Spektren ähneln sich in bestimmten Wellenlängenbereichen sehr. In anderen sind jedoch große Abweichungen möglich. Demzufolge werden Fréchet-Distanzen zwischen einzelnen Spektren der Trainingsmenge fast immer in einem Wellenlängen-Bereich realisiert, der überhaupt nicht von Interesse ist.

Eine Möglichkeit, mit diesem Problem umzugehen, ist die Einschränkung des Wellenlängenbereichs, in dem Distanzen überhaupt gemessen werden. Das erfordert aber gut begründete Entscheidungen eines Experten und ist naturgemäß von der konkreten Problemstellung in hohem Maße abhängig. Universeller wäre es, aus den Wellenlängenbereichen und den darüber vorhandenen Abweichungen innerhalb der Trainingsmenge generisch abzuleiten, an welchen Stellen besonders gute Übereinstimmungen gefordert sind.

In einem zweiten Schritt müsste auf der Grundlage dieses Toleranzverlaufs ein Distanzbegriff generiert werden, der unterschiedlich wichtige Wellenlängenbereiche entsprechend bewertet und z. B. in einem Bereich mit großen Abweichungen in der Trainingsmenge auch größere Abweichungen nicht zu stark bestraft.

In diesem Abschnitt werden Ansätze für diese beiden Konzepte in umgekehrter Reihenfolge vorgestellt. Zunächst wird diskutiert, wie überhaupt eine Variante der Fréchet-Distanz sinnvoll definiert und berechnet werden kann, die in Abhängigkeit von Wellenlängen unterschiedlich empfindlich ist. Um dieses Konzept schließlich zur Anwendung zu bringen, wird danach der konkrete Vorgang der *Lokalisierung der Leine* besprochen, wie also aus einer solchen Trainingsmenge ein Toleranzverlauf abgeleitet werden kann, der sich mit einer lokalisierten Fréchet-Distanz sinnvoll kombinieren lässt. Auch dieser Vorgang kann auf der Grundlage der Fréchet-Distanz selbst erarbeitet werden.

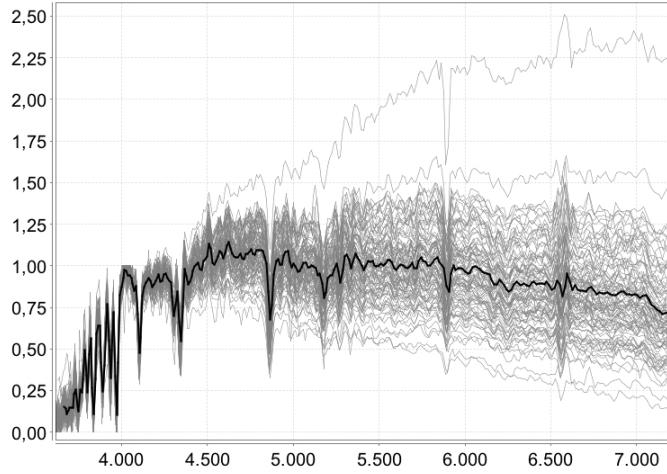


Abbildung 3.1: Eine Visualisierung der Spektren von als positiv vorklassifizierten sternbildenden Galaxien aus der dem Autor vorliegenden Trainingsmenge aus dem SDSS-Katalog. Die Spektren sind so normiert, dass ihre Intensitäten jeweils in der sogenannten Ballmer-Range (der Aufschwung im linken Bereich bis ca. 4200Å) Werte zwischen null und eins annehmen. Deutlich zu erkennen ist hier, dass die Intensitätswerte oberhalb von etwa 4500Å starke Abweichungen aufweisen. Die paarweisen Fréchet-Distanzen werden dadurch groß. Im Bild schwarz markiert ist ein fiktives Spektrum, das aus den durchschnittlichen Intensitäten zu jeder Wellenlänge errechnet wurde.

3.1 Formalisierung

Da die hier verwendeten Spektren (als Funktionen der Wellenlänge) zu in x -Richtung monotonen polygonalen Kurven führen, kann man eine Zuordnung von Toleranzen zu Wellenlängen auch durch eine Zuordnung von Toleranzen zu Segmentadressen einer Kurve ausdrücken. Bei der Wahl von Parametern $s \in [0, n]$ und $t \in [0, m]$ zweier zu vergleichender Kurven $T_1 : [0, n] \rightarrow \mathbb{R}^2$ und $T_2 : [0, m] \rightarrow \mathbb{R}^2$ wird nun eine der beiden Kurven (hier T_1) als Referenzkurve ausgezeichnet (die lokalisierte Fréchet-Distanz ist kein symmetrisches Konzept) und die erlaubte Distanz zweier Punkte $(T_1(s), T_2(t))$ in Abhängigkeit vom Parameter s für die Referenzkurve lokal ermittelt. Dazu nehme man eine Abbildung dl_{T_1} als gegeben an, die jedem Segment der Referenzkurve T_1 eine (nichtnegative) maximale Distanz zuordnet.¹ (Abbildung 3.2)

Nicht unmittelbar klar ist, welche maximale punktweise Distanz gelten soll, wenn der zugehörige Punkt auf der Referenzkurve auf einem Knoten zwischen zwei Segmenten a und b mit verschiedenen Distanzen $dl_{T_1}(a) \neq dl_{T_1}(b)$ liegt. Im Folgenden wird es sich als sinnvoll herauszustellen, hier das Minimum der beiden in Frage kommenden Distanzen zu wählen.

Überlegt man sich dann noch, dass die Parameterpaare (s, t) im Parameterraum $[0, n] \times [0, m]$ durch Parametrisierungen $\alpha : I \rightarrow [0, n]$ von T_1 und $\beta : I \rightarrow [0, m]$ von T_2 bei gleichzeitiger Auswertung zum Zeitpunkt $t \in I$ realisiert werden und

¹ Dabei seien die Segmente durch den Index des ihr Ende bildenden Punktes indiziert, das erste Segment hat also den Index 1, das letzte den Index n . Diese Wahl erweist sich im Kontext des resultierenden Freespace-Diagramms als nützlich, da die Zellen $C^{i,j}$ dabei jeweils zum Segment i von T_1 gehören, sich also in der i -ten Spalte befinden.

es diese Parametrisierungen sind, die im Kontext der Entscheidung des Fréchet-Distanz-Entscheidungsproblems bzw. ihrer konkreten Berechnung variiert werden, kann man obige Überlegungen in folgender Definition formalisieren:

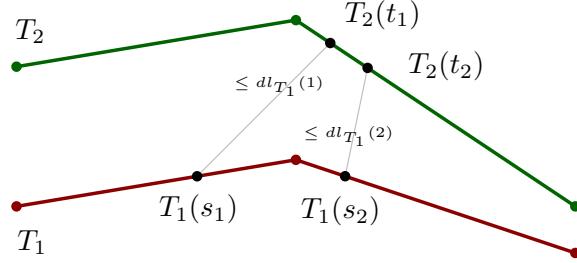


Abbildung 3.2: Bei der Frage, ob ein Punktpaar auf den polygonalen Kurven T_1 , T_2 von einem verbotenen oder erlaubten Parameterpaar (s, t) induziert wird, ist in der lokalisierter Variante entscheidend, welches Segment der Referenzkurve (hier: T_1) beteiligt ist. In diesem Beispiel liegen die Punkte auf T_1 für die beiden Parameter $s_1 < s_2$ jeweils auf verschiedenen Segmenten $1 < 2$. Entsprechend gelten möglicherweise verschiedene Leinenlängen $dl_{T_1}(1) \neq dl_{T_1}(2)$.

Definition 3.1 Zu einer polygonalen Kurve $T_1 : [0, n] \rightarrow \mathbb{R}^2$ bestehend aus n Segmenten gemeinsam mit einer **Distanzlokalisierung** $dl_{T_1} : \{1, \dots, n\} \rightarrow \mathbb{R}^{\geq 0}$ und einer weiteren polygonalen Kurve $T_2 : [0, m] \rightarrow \mathbb{R}^2$ bestehend aus m Segmenten löst ein Parametrisierungspaar $\alpha : I \rightarrow [0, n]$ für T_1 und $\beta : I \rightarrow [0, m]$ für T_2 das **lokalierte Fréchet-Distanz-Entscheidungsproblem**, falls für $t \in I$ stets gilt:

$$d(T_1(\alpha(t)), T_2(\beta(t))) \leq \begin{cases} dl_{T_1}(1) & \text{falls } \alpha(t) = 0 \\ \min(dl_{T_1}(\alpha(t)), dl_{T_1}(\alpha(t) + 1)) & \text{falls } \alpha(t) \in \{1, \dots, n - 1\} \\ dl_{T_1}(\lceil \alpha(t) \rceil) & \text{sonst} \end{cases}$$

Bemerkung 3.2 Wählt man in der obigen Definition 3.1 die Distanzlokalisierung $dl_{T_1} : \{1, \dots, n\} \rightarrow \{\delta\}$ konstant zu einem vorgegebenen maximalen punktweisen Abstand δ , so erhält man das Standard-Fréchet-Distanz-Entscheidungsproblem als Spezialfall.

Im Kontext des Freespace-Diagramms bedeutet die Einführung einer solchen Lokalisierung, dass die Spalten (entsprechend Segmenten der Kurve in x -Richtung) auf unterschiedlichen punktweisen maximalen Distanzen beruhen. An der Berechnung des Freespace in den einzelnen Zellen ändert das nichts, denn dort ist die Distanz eindeutig definiert. An den Kanten zwischen Spalten jedoch stimmen dann die freien Zellenrandstücke im Allgemeinen nicht mehr überein (im Gegensatz zu Lemma 2.13). Durch Definition 3.1 ist hier festgelegt, dass das Minimum der beiden benachbarten Distanzen gewählt wird. Was das geometrisch bedeutet, wird im Folgenden diskutiert:

Definition 3.3 Zu zwei polygonalen Kurven $T_1 : [0, n] \rightarrow \mathbb{R}^2$ und $T_2 : [0, m] \rightarrow \mathbb{R}^2$ gemeinsam mit einer Distanzlokalisierung dl_{T_1} für die Referenzkurve T_1 erhält man das **lokalierte Freespace-Diagramm** wie folgt: im Inneren einer Zelle $C^{i,j}$ sowie am Diagrammrand ist ein Punkt (s, t) genau dann frei, wenn er bezüglich des maximalen punktweisen Abstands für die zugehörige i -te Spalte frei ist:

$$(s, t) \in C_W^{i,j} \iff d(T_1(s), T_2(t)) \leq dl_{T_1}(i)$$

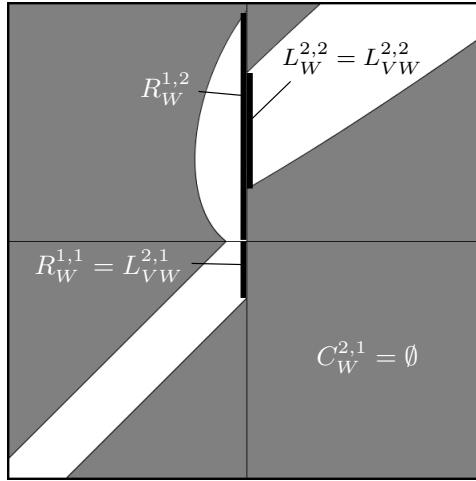


Abbildung 3.3: Das lokalisierte Freespace-Diagramm der Kurven aus Abbildung 3.2 mit einer Distanzlokalisierung $dl_{T_1}(1) > dl_{T_1}(2)$. Dadurch stimmen benachbarte freie Zellenrandstücke nicht mehr überein. Basierend auf Definition 3.1 erhält man in Definition 3.3 den Begriff des gemeinsamen freien Zellenrandstücks, das auf dem Minimum der dort angrenzenden maximalen punktweisen Distanzen beruht.

Als lokalisierten Whitespace auf inneren Spaltengrenzen ($i \in \{2, \dots, n\}$) definiert man das gemeinsame freie Zellenrandstück („very white“) als

$$L_{VW}^{i,j} := R_{VW}^{i-1,j} := L_W^{i,j} \cap R_W^{i-1,j}.$$

Ein Punkt auf inneren Spaltengrenzen ist genau dann frei, wenn er innerhalb eines gemeinsamen freien Zellenrandstücks liegt.

Aus Konsistenzgründen sei schließlich noch für das ganze Diagramm die Notation mit VW statt W eingeführt: Für alle Zellenrandstücke und Zellen X , für die X_{VW} oben noch nicht definiert wurde, sei X_{VW} definiert als die herkömmliche freie Teilmenge X_W .

Diese Definitionen sind so gemacht, dass die Begriffe des lokalisierten Fréchet-Distanz-Entscheidungsproblems mit denen des lokalisierten Freespace-Diagramms kompatibel sind, damit letzteres zur Lösung des ersteren überhaupt Verwendung finden kann:

Korollar 3.4 Verläuft die Kurve $\xi = (\alpha, \beta) : I \rightarrow [0, n] \times [0, m]$ im Freespace-Diagramm als Parametrisierungspaar für die polygonalen Kurven $T_1 : [0, n] \rightarrow \mathbb{R}^2$ und $T_2 : [0, m] \rightarrow \mathbb{R}^2$ samt einer Distanzlokalisierung $dl_{T_1} : \{1, \dots, n\} \rightarrow \mathbb{R}^{\geq 0}$ komplett im Freespace nach Definition 3.3, so löst sie das lokalisierte Fréchet-Distanz-Entscheidungsproblem nach Definition 3.1.

Beweis. Verlufe $\xi = (\alpha, \beta)$ also im Freespace. Dann muss zu jedem Zeitpunkt $t \in I$ die Ungleichung aus Definition 3.1 erfüllt sein. Betrachte die folgende Fallunterscheidung:

- Zum Startzeitpunkt $t = 0$ gilt aufgrund der Eigenschaft von α , eine Parametrisierung von T_1 zu sein, $\alpha(0) = 0$. Nach Voraussetzung gilt

$$\xi(t) = \xi(0) = (0, 0) \in C_W^{1,1},$$

also gilt nach Definition 3.3:

$$d(T_1(\alpha(t)), T_2(\beta(t))) = d(T_1(0), T_2(0)) \leq dl_{T_1}(1)$$

Damit ist der erste Fall der Fallunterscheidung in Definition 3.1 erfüllt.

- Sei nun t so gewählt, dass T_1 durch den Parameter $\alpha(t) \in \{1, \dots, n-1\}$ auf einen Knoten im inneren Bereich abbildet. Dann liegt $\xi(t) = (\alpha(t), \beta(t))$ auf einem inneren Zellenrandstück, nach Definition 3.3 gibt es also $j \in \{1, \dots, m\}$, so dass gilt:

$$\xi(t) = (\alpha(t), \beta(t)) \in L_{VW}^{\alpha(t)+1, j} = L_W^{\alpha(t)+1, j} \cap R_W^{\alpha(t), j}$$

Sei ohne Einschränkung $dl_{T_1}(\alpha(t)) < dl_{T_2}(\alpha(t) + 1)$ angenommen (umgekehrt funktioniert das Argument genau so). Aufgrund der Gestalt des Freespace gilt dann $R_W^{\alpha(t), j} \subset L_W^{\alpha(t)+1, j}$, woraus

$$L_{VW}^{\alpha(t)+1, j} = L_W^{\alpha(t)+1, j} \cap R_W^{\alpha(t), j} = R_W^{\alpha(t), j}$$

folgt. Und damit gilt

$$d(T_1(\alpha(t)), T_2(\beta(t))) \leq dl_{T_1}(\alpha(t)) = \min(dl_{T_1}(\alpha(t)), dl_{T_1}(\alpha(t) + 1)),$$

womit der zweite Fall der Fallunterscheidung in Definition 3.1 erfüllt ist.

- In allen anderen Fällen für $t \in I$ gilt entweder

$$\exists i \in \{1, \dots, n\} : \alpha(t) \in]i-1, i[\quad \text{oder} \quad \alpha(t) = n =: i.$$

In beiden Fällen gibt es dann ein $j \in \{1, \dots, m\}$, so dass

$$\xi(t) \in C_W^{\lceil \alpha(t) \rceil, j} = C_W^{i, j} = C_{VW}^{i, j}.$$

und damit

$$d(T_1(\alpha(t)), T_2(\beta(t))) \leq dl_{T_1}(\lceil \alpha(t) \rceil)$$

nach Definition 3.3.

Damit ist auch der letzte Fall der Fallunterscheidung aus Definition 3.1 abgedeckt und $\xi = (\alpha, \beta)$ eine Lösung des gegebenen lokalisierten Fréchet-Distanz-Entscheidungsproblems. \square

Wird also ein Weg gefunden, einen solchen Pfad ξ von der unteren linken Ecke $(0, 0)$ in die obere rechte Ecke (n, m) des lokalisierten Freespace-Diagramms zu bilden, so kann damit analog zum Standard-Fréchet-Distanz-Entscheidungsproblem auch die lokalisierte Variante gelöst werden.

3.2 Konkrete Berechnungen

Eine kleine Erinnerung: bei der Standard-Fréchet-Distanz war das formale Mittel zur Lösung des Entscheidungsproblems die Bestimmung des erreichbaren Raumes im Freespace-Diagramm nebst der Frage, ob der obere rechte Punkt (n, m) erreichbar sei. Damit gibt es dann einen Pfad ξ von $(0, 0)$ dorthin, dessen Komponenten (α, β) Parametrisierungen der beiden polygonalen Kurven T_1 und T_2 sind, die die Eigenschaft

$$d(T_1(\alpha(t)), T_2(\beta(t))) \leq \delta \quad \forall t \in I$$

erfüllen. Diese Eigenschaft wurde durch die Distanzlokalisierung in Definition 3.1 auf komplexe Weise variiert. Mit Korollar 3.4 wurde jedoch gezeigt, dass ein Pfad, der komplett im Freespace des lokalisierten Freespace-Diagramms verläuft, auch diese variierte Eigenschaft automatisch erfüllt. Kann man also auf Basis des lokalisierten Diagramms den erreichbaren Raum berechnen, so erhält man eine Lösung des lokalisierten Fréchet-Distanz-Entscheidungsproblems. Wie im Folgenden gezeigt wird, ist das aber fast genau so einfach möglich, wie im Standardfall:

Definition 3.5 Analog zum erreichbaren Raum im Standardfall definiert man den erreichbaren Raum im lokalisierten Freespace-Diagramm („very reachable“) auf Basis des lokalisierten Freespace:

$$C_{VR}^{i,j} := \left\{ (s, t) \in C_{VW}^{i,j} \mid \begin{array}{l} \exists \xi : I \rightarrow \bigcup_{i,j} C_{VW}^{i,j} \text{ stetig und in beide Richtungen} \\ \text{en monoton mit } \xi(0) = (0, 0) \text{ und } \xi(1) = (s, t) \end{array} \right\}.$$

Auch die erreichbaren Zellenrandstücke sind Schnitte mit dem lokalisierten Freespace:

$$\begin{aligned} L_{VR}^{i,j} &:= L^{i,j} \cap C_{VR}^{i,j} \\ B_{VR}^{i,j} &:= B^{i,j} \cap C_{VR}^{i,j} \\ R_{VR}^{i,j} &:= R^{i,j} \cap C_{VR}^{i,j} \\ T_{VR}^{i,j} &:= T^{i,j} \cap C_{VR}^{i,j}. \end{aligned}$$

Mit den hier definierten Begriffen ist es nun problemlos möglich, die induktive Berechnung der erreichbaren lokalisierten Zellenrandstücke analog zum Standardfall durchzuführen. Damit erhält man eine einfach zu berechnende Lösung für das lokalisierte Fréchet-Distanz-Entscheidungsproblem.

Korollar 3.6 Zu zwei polygonalen Kurven $T_1 : [0, n] \rightarrow \mathbb{R}$ und $T_2 : [0, m] \rightarrow \mathbb{R}$ seien die zugehörigen Freespace-Zellen $(C^{i,j})_{i,j}$ mit lokalisiertem Freespace $(C_{VW}^{i,j})_{i,j}$ gegeben. Durch Anwendung der folgenden Regeln können die erreichbaren Zellenrandstücke induktiv errechnet werden:

- (a) Falls $(0, 0) \notin C_{VW}^{1,1}$ gilt, ist kein einziger Punkt des lokalisierten Freespace-Diagramms erreichbar, damit sind alle erreichbaren Zellenrandstücke leer. Ist $(0, 0) \in C_{VW}^{1,1}$ frei, so auch erreichbar.
- (b) Für die Zellen am linken/unteren Rand des Diagramms $C^{1,j}$ für $j = 1, \dots, m$ bzw. $C^{i,1}$ für $i = 1, \dots, n$ ermittelt man die linken/unteren erreichbaren Zellenrandstücke so:

$$\begin{aligned} L_{VR}^{1,j} &= \begin{cases} L_{VW}^{1,j} & \text{falls } (0, j-1) \text{ erreichbar} \\ \emptyset & \text{sonst} \end{cases} \\ B_{VR}^{i,1} &= \begin{cases} B_{VW}^{i,1} & \text{falls } (i-1, 0) \text{ erreichbar} \\ \emptyset & \text{sonst} \end{cases} \end{aligned}$$

- (c) Induktive Berechnung der erreichbaren rechten und oberen erreichbaren Zellenrandstücke von $C^{i,j}$: Sind die folgenden Zellenrandstücke bekannt

$$\begin{aligned} L_{VR}^{i,j} &= \{i-1\} \times [l_b, l_t] \\ B_{VR}^{i,j} &= [b_l, b_r] \times \{j-1\} \\ R_{VR}^{i,j} &= \{i\} \times [r_b, r_t] \\ T_{VR}^{i,j} &= [t_l, t_r] \times \{j\}, \end{aligned}$$

so gilt:

$$R_{VR}^{i,j} = \begin{cases} R_{VW}^{i,j} & \text{falls } B_{VR}^{i,j} \neq \emptyset \\ \emptyset & \text{falls } B_{VR}^{i,j} = \emptyset \text{ und } l_b > r_t \\ \{i\} \times [\max(l_b, r_b), r_t] & \text{falls } B_{VR}^{i,j} = \emptyset \text{ und } l_b \leq r_t \end{cases}$$

$$T_{VR}^{i,j} = \begin{cases} T_{VW}^{i,j} & \text{falls } L_{VR}^{i,j} \neq \emptyset \\ \emptyset & \text{falls } L_{VR}^{i,j} = \emptyset \text{ und } b_l > t_r \\ [\max(b_l, t_l), t_r] \times \{j\} & \text{falls } L_{VR}^{i,j} = \emptyset \text{ und } b_l \leq t_r . \end{cases}$$

Beweis. Die im Beweis zu Lemma 2.16 verwendeten Argumente funktionieren auf der Basis des lokalisierten Freespace-Begriffs genau so wie im Standardfall. \square

Damit lässt sich auch direkt ein Algorithmus angeben:

3.2.1 Lösung des lokalisierten Entscheidungsproblems

Zu zwei polygonalen Kurven $T_1 : [0, n] \rightarrow \mathbb{R}^2$ und $T_2 : [0, m] \rightarrow \mathbb{R}^2$ gemeinsam mit einer Distanzlokalisierung $dl_{T_1} : \{1, \dots, n\} \rightarrow \mathbb{R}^{\geq 0}$ für T_1 existieren genau dann Parametrisierungen $\alpha : I \rightarrow [0, n]$ von T_1 und $\beta : I \rightarrow [0, m]$ von T_2 , die das lokalisierte Fréchet-Distanz-Entscheidungsproblem nach Definition 3.1 lösen, wenn $(n, m) \in R_{VR}^{n,m} \subset C_{VR}^{n,m}$ liegt. Diese erreichbaren Zellenrandstücke errechnet man analog zum Standardfall so:

- Ist $d(T_1(0), T_2(0)) > dl_{T_1}(1)$, so ist $(0, 0)$ im lokalisierten Freespace-Diagramm nicht frei und damit kein einziger Punkt des Diagramms erreichbar – die oben genannten Parametrisierungen können nicht existieren.
- Für die Zellen am linken bzw. unteren Rand $C^{1,j}$ für $j = 1, \dots, m$ bzw. $C^{i,1}$ für $i = 1, \dots, n$ erhält man

$$L_{VR}^{1,j} = L_{VW}^{1,j} \quad \text{bzw.} \quad B_{VR}^{i,1} = B_{VW}^{i,1}$$

falls der linke untere Punkt $(0, j - 1)$ bzw. $(i - 1, 0)$ erreichbar ist.

- Ausgehend von der linken unteren Ecke kann man für Indizes $i = 1, \dots, n$ und $j = 1, \dots, m$ die erreichbaren Zellenrandstücke von $C^{i,j}$ schrittweise berechnen: Sind die linken und unteren erreichbaren Zellenrandstücke $L_{VR}^{i,j}, B_{VR}^{i,j}$ sowie die rechten und oberen freien Zellenrandstücke $R_{VW}^{i,j}, T_{VW}^{i,j}$ bekannt, ermittelt man die rechten und oberen Zellenrandstücke wie folgt:

$$R_{VR}^{i,j} = \begin{cases} R_{VW}^{i,j} & \text{falls } B_{VR}^{i,j} \neq \emptyset \\ \emptyset & \text{falls } B_{VR}^{i,j} = \emptyset \text{ und } l_b > r_t \\ \{i\} \times [\max(l_b, r_b), r_t] & \text{falls } B_{VR}^{i,j} = \emptyset \text{ und } l_b \leq r_t \end{cases}$$

$$T_{VR}^{i,j} = \begin{cases} T_{VW}^{i,j} & \text{falls } L_{VR}^{i,j} \neq \emptyset \\ \emptyset & \text{falls } L_{VR}^{i,j} = \emptyset \text{ und } b_l > t_r \\ [\max(b_l, t_l), t_r] \times \{j\} & \text{falls } L_{VR}^{i,j} = \emptyset \text{ und } b_l \leq t_r . \end{cases}$$

Mit einer Berechnungsvorschrift für das lokalisierte Fréchet-Distanz-Entscheidungsproblem ist auch die Voraussetzung für die exakte Berechnung einer lokalisierten Fréchet-Distanz gesichert. Zur Erinnerung: im Standardfall ist die Fréchet-Distanz zweier Kurven δ der kleinste maximale punktweise Abstand, für den das Entscheidungsproblem noch positiv entschieden wird. Da im lokalisierten Fall dieser maximale punktweise Abstand von der Position auf der Referenzkurve abhängt, muss zunächst ein sinnvoller Begriff einer lokalisierten Fréchet-Distanz definiert werden:

Definition 3.7 Sind $T_1 : [0, n] \rightarrow \mathbb{R}^2$ und $T_2 : [0, m] \rightarrow \mathbb{R}^2$ polygonale Kurven und ist $dl_{T_1} : \{1, \dots, n\} \rightarrow \mathbb{R}^{\geq 0}$ eine Distanzlokalisierung für T_1 , so sei zunächst für $r \in \mathbb{R}^{\geq 0}$ die **skalierte Distanzlokalisierung** $r \cdot dl_{T_1}$ das punktweise r -fache von dl_{T_1} :

$$r \cdot dl_{T_1} : \{1, \dots, n\} \rightarrow \mathbb{R}^{\geq 0}, \quad i \mapsto r \cdot dl_{T_1}(i)$$

Die **lokalisierte Fréchet-Distanz** $\delta_F(T_1, T_2, dl_{T_1})$ ist dann definiert als der kleinste Faktor $r \in \mathbb{R}^{\geq 0}$, für den das lokalisierte Fréchet-Distanz-Entscheidungsproblem mit der Distanzlokalisierung $r \cdot dl_{T_1}$ noch positiv entschieden wird:

$$\delta_F(T_1, T_2, dl_{T_1}) := \min \left\{ r \in \mathbb{R}^{\geq 0} \mid \begin{array}{l} \text{lokalisiertes Fréchet-Distanz-Entscheidungs-} \\ \text{problem mit } T_1, T_2 \text{ und } r \cdot dl_{T_1} \text{ positiv} \end{array} \right\}$$

Dass diese Definition sinnvoll ist, kann etwa dadurch gezeigt werden, dass die Standard-Fréchet-Distanz als Spezialfall davon abgedeckt wird:

Korollar 3.8 Ist in der Situation von obiger Definition 3.7 die Distanzlokalisierung $dl_{T_1} : \{1, \dots, n\} \rightarrow \{1\}$ konstant, dann entspricht die zugehörige lokale Fréchet-Distanz der Standard-Fréchet-Distanz:

$$\delta_F(T_1, T_2, dl_{T_1}) = \delta_F(T_1, T_2)$$

Beweis. Sei zunächst die Standard-Fréchet-Distanz gegeben:

$$\delta := \delta_F(T_1, T_2) = \inf_{\substack{\alpha: I \rightarrow [0, n] \\ \beta: I \rightarrow [0, m]}} \max_{t \in I} d(T_1(\alpha(t)), T_2(\beta(t)))$$

Dann ist nach der Definition der lokalisierten Fréchet-Distanz zu zeigen, dass das lokalisierte Fréchet-Distanz-Entscheidungsproblem mit T_1 , T_2 und $\delta \cdot dl_{T_1}$ positiv gelöst wird und dass diese Wahl für die Skalierung minimal war:

- Seien $\alpha : I \rightarrow [0, n]$ und $\beta : I \rightarrow [0, m]$ Parametrisierungen von T_1 und T_2 , für die δ den maximalen „gleichzeitigen“ punktweisen Abstand auf T_1 und T_2 realisiert:

$$\delta = \max_{t \in I} d(T_1(\alpha(t)), T_2(\beta(t)))$$

Dann gilt die Ungleichung aus Definition 3.1 für alle $t \in I$ und sogar für alle T_1 -Segmente $i \in \{1, \dots, n\}$ unabhängig von $\alpha(t)$ da dl_{T_1} konstant ist:

$$d(T_1(\alpha(t)), T_2(\beta(t))) \leq \delta = \delta \cdot (dl_{T_1}(i)) = (\delta \cdot dl_{T_1})(i)$$

Damit folgt insbesondere, dass das Parametrisierungspaar α für T_1 und β für T_2 zusammen mit der konstanten Distanzlokalisierung $\delta \cdot (i \mapsto 1) = (i \mapsto \delta)$ das lokalisierte Fréchet-Distanz-Entscheidungsproblem löst.

- Sei nun $d < \delta$, dann gilt für alle Parametrisierungen $\alpha : I \rightarrow [0, n]$ von T_1 und $\beta : I \rightarrow [0, m]$ von T_2 und alle T_1 -Segmente $i \in \{1, \dots, n\}$:

$$\exists t \in I : (d \cdot dl_{T_1})(i) = d \cdot (dl_{T_1}(i)) = d < d(T_1(\alpha(t)), T_2(\beta(t)))$$

Entsprechend können keine solchen Parametrisierungen das lokalisierte Fréchet-Distanz-Entscheidungsproblem für $T_1, T_2, d \cdot dl_{T_1}$ lösen.

Aus der durch δ gegebenen Lösbarkeit des lokalisierten Problems sowie dem Minimalitätsargument folgt, dass in diesem speziellen Fall mit konstanter Distanzlokalisierung die beiden Fréchet-Distanz-Begriffe äquivalent sind. \square

Zur konkreten Berechnung der lokalisierten Variante ist wiederum eine Modifikation des Vorgehens im Standardfall nötig, die diesen aber als Spezialfall mit einschließt. Zur Erinnerung: die exakte Berechnung der Standard-Fréchet-Distanz basierte auf einer Liste von kritischen Werten für die maximale punktweise Distanz, bei denen im Freespace-Diagramm überhaupt eine Änderung mit Einfluss auf die Erreichbarkeit der rechten oberen Ecke stattfinden konnte. Von diesen kritischen Werten ist der kleinste mit positiver Lösung des Entscheidungsproblems die gesuchte Fréchet-Distanz.

3.2.2 Ermittlung kritischer Werte

Da die maximale punktweise Distanz von der Position auf der Referenzkurve abhängt (Definition 3.1), ist es sinnlos, lokale Werte der Distanzlokalisierung als kritisch einzustufen. Entsprechend dem Vorgehen in Definition 3.7 wird hingegen die Distanzlokalisierung als Ganzes mit einem Faktor versehen, welcher im lokalisierten Fall die Rolle der zu variierenden maximalen punktweisen Distanz übernimmt. Demzufolge ist es sinnvoll, kritische Werte für diesen Distanzlokalisierungsfaktor $r \in \mathbb{R}^{\geq 0}$ zu ermitteln:

Lemma 3.9 *Kritische Werte für den Distanzlokalisierungsfaktor $r \in \mathbb{R}^{\geq 0}$ im lokalisierten Freespace-Diagramm lassen sich in Verallgemeinerung zum Standardfall (Abschnitt 2.2.5 und [AG92, Seite 80]) durch folgende drei Typen charakterisieren:*

- (a) *r ist minimal mit $(0, 0) \in C_{VW}^{1,1}$ und $(n, m) \in C_{VW}^{n,m}$ (Freiheit von Start- und Endpunkt). Der kritische Wert ist*

$$\max \left(\frac{d(T_1(0), T_2(0))}{dl_{T_1}(1)}, \frac{d(T_1(n), T_2(m))}{dl_{T_1}(n)} \right)$$

- (b) *r ist minimal, so dass das gemeinsame freie Zellenrandstück $L_{VW}^{i,j}$ oder $B_{VW}^{i,j}$ nicht leer ist für eine Zelle $C^{i,j}$ (Übergang zwischen benachbarten Zellen). Die kritischen Werte sind*

- *für $i \in \{2, \dots, n\}$, $j \in \{1, \dots, m\}$ die Abstände der Punkte $T_1(i-1)$ von den Segmenten $T_2|_{[j-1,j]}$ geteilt durch den an $T_1(i-1)$ gültigen lokalen maximalen punktweisen Abstand und*
- *für $j \in \{2, \dots, m\}$, $i \in \{1, \dots, n\}$ die Abstände der Punkte $T_2(j-1)$ von den Segmenten $T_1|_{[i-1,i]}$ geteilt durch den dort gültigen lokalen maximalen punktweisen Abstand $dl_{T_1}(i)$.*

- (c) *r ist minimal, so dass*

- für zwei Zellen C^{i,j_1}, C^{i,j_2} mit $j_1 < j_2$ und $B_{VW}^{i,j_1} = [b_l, b_r] \times \{j_1 - 1\}$ sowie $B_{VW}^{i,j_2} = [t_l, t_r] \times \{j_2 - 1\}$ gilt: $b_l = t_r$ (Vertikale zellenübergreifende Passage) oder
- für zwei Zellen $C^{i_1,j}, C^{i_2,j}$ mit $i_1 < i_2$ und $L_{VW}^{i_1,j} = \{i_1 - 1\} \times [l_b, l_t]$ sowie $L_{VW}^{i_2,j} = \{i_2 - 1\} \times [r_b, r_t]$ gilt: $l_b = r_t$ (Horizontale zellenübergreifende Passage).

Die entsprechenden kritischen Werte ermittelt man so:

- Im vertikalen Fall sind die Abstände der Punkte $T_2(j_1 - 1), T_2(j_2 - 1)$ zum Schnittpunkt der mittig zwischen ihnen verlaufenden Linie mit dem Segment $T_1|_{[i-1,i]}$, geteilt durch den lokal vorgegebenen maximalen punktweisen Abstand $dl_{T_1}(i)$, kritisch.
- Im horizontalen Fall fixiere zunächst die nach Definition 3.1 lokal an den Spaltengrenzen geltenden maximalen punktweisen Abstände d_1 bei $T_1(i_1 - 1)$ und d_2 bei $T_1(i_2 - 1)$.
 - Ist $d_1 = d_2$, so sind die kritischen Werte die Abstände der Punkte $T_1(i_1 - 1)$ und $T_1(i_2 - 1)$ zum Schnittpunkt der mittig zwischen ihnen verlaufenden Linie mit dem Segment $T_2|_{[j-1,j]}$, geteilt durch den lokal vorgegebenen maximalen punktweisen Abstand $d_1 = d_2$ analog zum vertikalen Fall.
 - Ist $d_1 \neq d_2$, so erhält man die kritischen Werte für alle möglichen Wahlen von i_1, i_2, j so: Sei $q := d_1/d_2$ der Quotient der beiden relevanten Abstände und

$$\ell : \mathbb{R} \rightarrow \mathbb{R}^2, \quad s \mapsto T_1(i_1 - 1) + s \cdot (T_1(i_2 - 1) - T_1(i_1 - 1))$$

die Linie durch die beiden den Spaltengrenzen entsprechenden Punkte $T_1(i_1 - 1)$ und $T_1(i_2 - 1)$. Ferner seien die ℓ -Parameter

$$s_1 := \frac{q}{q+1} \quad \text{und} \quad s_2 := \frac{q}{q-1}$$

und die Punkte

$$p_1 := \ell(s_1), \quad p_2 := \ell(s_2) \quad \text{und} \quad m := \ell\left(\frac{s_1 + s_2}{2}\right)$$

gegeben. Falls es einen Schnittpunkt $T_2(t)$ mit $t \in [j-1, j]$ des Kreises mit Mittelpunkt m und Radius $d(m, p_1) = d(m, p_2)$ mit dem Segment $T_2|_{[j-1,j]}$ gibt, so ist

$$\frac{d(T_1(i_1 - 1), T_2(t))}{d_1} = \frac{d(T_1(i_2 - 1), T_2(t))}{d_2}$$

ein kritischer Wert. Falls es zwei Schnittpunkte gibt, so ist der kleinere resultierende kritische Wert von Interesse. Siehe Abbildung 3.4.

Beweis. Die Aussagen sind zum größten Teil äquivalent zu den kritischen Werten der Standard-Fréchet-Distanz (Abschnitt 2.2.5), wenn man die freien Zellenrandstücke X_W durch die gemeinsam-freien Zellenrandstücke X_{VW} nach Definition 3.3 ersetzt und berücksichtigt, dass die kritischen Werte keine Werte für die Distanz selbst, sondern für einen Faktor der Distanzlokalisierung sind:

- (a) Die Ermittlung dieses Wertes wird durch die Lokalisierung überhaupt nicht beeinflusst. Bei der Berechnung muss lediglich die richtige lokale maximale punktweise Distanz ($dl_{T_1}(1)$ für den linken unteren Punkt, $dl_{T_1}(n)$ für den rechten oberen Punkt) angewendet werden. Der minimale Faktor r , für den beide Punkte frei sind, ist gerade das Maximum der Endpunktdistanzen geteilt durch die dort gültige lokale maximale punktweise Distanz:

$$r := \max \left(\frac{d(T_1(0), T_2(0))}{dl_{T_1}(1)}, \frac{d(T_1(n), T_2(m))}{dl_{T_1}(n)} \right)$$

Denn dann beträgt der Abstand von Start- und Endpunkten höchstens die jeweils lokale maximalen punktweise Distanz um den Faktor r skaliert:

$$\begin{aligned} d(T_1(0), T_2(0)) &= \frac{d(T_1(0), T_2(0))}{dl_{T_1}(1)} \cdot dl_{T_1}(1) \\ &\leq \max \left(\frac{d(T_1(0), T_2(0))}{dl_{T_1}(1)}, \frac{d(T_1(n), T_2(m))}{dl_{T_1}(n)} \right) \cdot dl_{T_1}(1) \\ &= r \cdot dl_{T_1}(1) \end{aligned}$$

(Die Rechnung für den Abstand der Endpunkte verläuft genau so.)

- (b) Da nur an horizontalen Zellengrenzen durch die Lokalisierung verschiedene maximale punktweise Distanzen auftreten können, ergibt sich im vertikalen Fall bis auf die Anpassung an die Skalierung der Distanzlokalisierung keine Änderung zum Standardfall. Da das gemeinsame freie Zellenrandstück

$$L_{VW}^{i,j} = R_W^{i-1,j} \cap L_W^{i,j}$$

für horizontale Zellenübergänge über das Minimum der beiden maximalen punktweisen Distanzen $d := \min(dl_{T_1}(i-1), dl_{T_1}(i))$ definiert ist, erhält man den zugehörigen kritischen Faktor r analog zum Standardfall als die Distanz des Punktes $T_1(i-1)$ vom Segment $T_2|_{[j-1,j]}$, geteilt durch die vorgegebene lokale maximale punktweise Distanz d .

- (c) Im Fall vertikaler zellenübergreifender Passagen gelten wiederum die selben maximalen punktweisen Distanzen, wodurch die zugehörigen kritischen Werte genau wie im Standardfall ermittelt werden können, jeweils geteilt durch den lokalen maximalen punktweisen Abstand gegeben durch die Distanzlokalisierung dl_{T_1} . Der Fall horizontaler zellenübergreifender Passagen ist interessanter: Eine horizontale spaltenübergreifende Passage zwischen Zellenrandstücken $L^{i_1,j}$ und $L^{i_2,j}$ mit $i_1 < i_2$ und lokalen maximalen punktweisen Distanzen

$$\min(dl_{T_1}(i_1-1), dl_{T_1}(i_1)) =: d_1 \quad \text{und} \quad \min(dl_{T_1}(i_2-1), dl_{T_1}(i_2)) =: d_2$$

öffnet sich realisiert durch Punkte $(\{i_1-1\}, t)$ und $(\{i_2-1\}, t)$ für einen Segmentparameter $t \in [j-1, j]$ dann, wenn zu minimal skaliert Distanzlokalisierung $r \cdot dl_{T_1}$ beide Parameterpaare auf dem Rand des Freespace liegen:

$$d(T_1(i_1-1), T_2(t)) = r \cdot d_1 \quad \text{und} \quad d(T_1(i_2-1), T_2(t)) = r \cdot d_2$$

Daraus folgt, dass der Punkt $T_2(t)$ auf dem Apollonischen Kreis

$$\left\{ (s, t) \in \mathbb{R}^2 \mid \frac{d(T_1(i_1-1), T_2(t))}{d(T_1(i_2-1), T_2(t))} = q \right\}$$

liegen muss. Eine konkrete Lösung erhält man als den Schnittpunkt dieses Kreises mit dem Segment $T_2|_{[j-1,j]}$, für den der Abstand zu den beiden Punkten minimal ist. Da im Falle von $q = 1$, also gleichen lokalen maximalen Distanzen $d_1 = d_2$, der Kreis zu der bereits bekannten mittig zwischen $T_1(i_1 - 1)$ und $T_1(i_2 - 1)$ verlaufenden Linie entartet, erhält man den dann entsprechenden kritischen Wert analog zum Standardfall durch Division durch $d_1 = d_2$.

Wie ein solcher Kreis im Fall $d_1 \neq d_2$ aussieht, kann man geometrisch konstruieren: Sei die Linie durch die Punkte $T_1(i_1 - 1)$ und $T_2(i_2 - 1)$ gegeben durch die Abbildung

$$\ell : \mathbb{R} \rightarrow \mathbb{R}^2, \quad s \mapsto T_1(i_1 - 1) + s \cdot (T_1(i_2 - 1) - T_1(i_1 - 1))$$

mit $\ell(0) = T_1(i_1 - 1)$ und $\ell(1) = T_1(i_2 - 1)$. Für Schnittpunkte des Kreises (gegeben durch ℓ -Parameter $s \in \mathbb{R}$ mit dieser Linie) muss gelten:

$$\frac{d(\ell(s), \ell(0))}{d(\ell(s), \ell(1))} = q \quad \text{und damit} \quad \frac{|s|}{|s - 1|} = q$$

Als Lösungen erhält man die ℓ -Parameter

$$s_{1,2} := \frac{q}{q \pm 1}$$

und damit die Punkte $p_1 := \ell(s_1)$ sowie $p_2 := \ell(s_2)$, als Mittelpunkt des Kreises schließlich $m := \ell((s_1 + s_2)/2)$ und als Radius $d(m, p_1) = d(m, p_2)$.

□

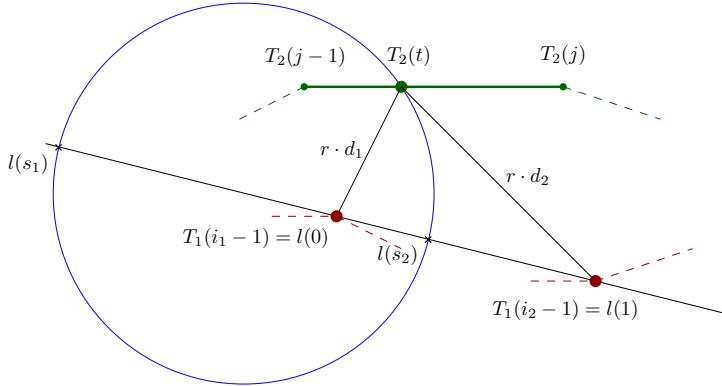


Abbildung 3.4: Eine horizontale spaltenübergreifende Passage zwischen gemeinsamen freien Zellenrandstücken im lokalisierten Freespace-Diagramm öffnet sich genau dann, wenn es einen Schnittpunkt des eingezeichneten Apollonischen Kreises bezüglich der Punkte $T_1(i_1 - 1)$ und $T_1(i_2 - 1)$ mit dem Segment $T_2|_{[j-1,j]}$ gibt, denn für alle Punkte auf diesem Kreis gilt, dass das Verhältnis ihrer Abstände zu den beiden Basispunkten dem Verhältnis der lokalen maximalen paarweisen Distanzen entspricht. Die Kenndaten Mittelpunkt und Radius des Kreises lassen sich über seine Schnittpunkte $\ell(s_1)$ bzw. $\ell(s_2)$ mit der Linie ℓ durch die zwei Basispunkte berechnen. Ist dieses Verhältnis der Distanzen 1, so entartet der Kreis zu einer Linie.

Mit den Anpassungen des Fréchet-Distanz-Entscheidungsproblems sowie der Parametersuche an den lokalisierten Fall, die beide den Standardfall umschließen, sind

konkrete Algorithmen bekannt, die lokalisierte Fréchet-Distanz als besondere Anpassung an den konkreten Anwendungsfall der astronomischen Spektren zu berechnen. Wie sinnvolle Werte für die verwendete Distanzlokalisierung aus einer Trainingsmenge von als positiv klassifizierten Spektren abgeleitet werden können, wird im folgenden Abschnitt diskutiert.

3.3 Die Lokalisierung der Leine

Im konkreten Anwendungsfall der Klassifikation sternbildender Galaxien war die Motivation einer Lokalisierung der Fréchet-Distanz darin begründet, dass die in der Trainingsmenge als positiv klassifizierten Spektren in einem bestimmten Wellenlängenbereich nach Normalisierung sehr große Ähnlichkeiten aufweisen, wohingegen bei größeren Wellenlängen teilweise extreme Abweichungen vorliegen (siehe Abbildung 3.1).

Das Ziel ist es nun, auf generische Weise und insbesondere ohne Expertenwissen solche Wellenlängenbereiche aus einer gegebenen Trainingsmenge abzuleiten und daraus eine Distanzlokalisierung zu generieren, die für (fast) alle Exemplare der Trainingsmenge eine geringe lokale Fréchet-Distanz (mit einem Distanzlokalisierungsfaktor nicht viel größer als 1) zu einer noch näher zu bestimmenden Referenzkurve realisiert. Hat ein Spektrum aber auch nur kleine Abweichungen in einem besonders sensiblen Wellenlängenbereich, so soll entsprechend die lokale Fréchet-Distanz groß sein. Dazu werden hier zwei Ansätze vorgestellt.

In beiden Fällen wird hier eine Referenzkurve benötigt, die sich auf einfache Weise aus den als positiv klassifizierten Spektren innerhalb der Trainingsmenge berechnen lässt. Allgemein kann man ein Spektrum auffassen als eine reelle Funktion einer endlichen Menge von Wellenlängen² W . Ist die als positiv klassifizierte Teilmenge der Trainingsmenge also gegeben als eine Menge T_p von Funktionen $W \rightarrow \mathbb{R}$, so kann man eine Referenzkurve naheliegend ermitteln über den punktweisen Durchschnitt oder Median³ med der Intensitäten:

$$\begin{aligned} \text{ref}_\emptyset : W &\rightarrow \mathbb{R}, \quad w \mapsto \emptyset \{s(w) \mid s \in T_p\} := \sum_{p \in T_p} \frac{p(w)}{|T_p|} \\ \text{ref}_m : W &\rightarrow \mathbb{R}, \quad w \mapsto \text{med}\{s(w) \mid s \in T_p\} \end{aligned}$$

Das Durchschnittsspektrum ref_\emptyset ist ebenfalls in Abbildung 3.1 zu sehen.

3.3.1 Lokalisierung über die punktweise Abweichung

Mehrere Strategien sind denkbar, zu jeder Wellenlänge ein Maß für die Abweichung innerhalb der Trainingsmenge zu berechnen. Zunächst kann man zu jeder Wellenlänge $w \in W$ alle Differenzen der Trainingsspektren von Referenzspektren errechnen. Man erhält so die Funktionen

$$\begin{aligned} \text{dists}_\emptyset : W &\rightarrow (\mathbb{R}^{\geq 0})^{|T_p|}, \quad w \mapsto \{|(\text{ref}_\emptyset(w) - s(w))| : s \in T_p\} \\ \text{dists}_m : W &\rightarrow (\mathbb{R}^{\geq 0})^{|T_p|}, \quad w \mapsto \{|(\text{ref}_m(w) - s(w))| : s \in T_p\} \end{aligned}$$

²Die genaue Diskretisierung der Wellenlängen ist ein nichttriviales Implementierungsdetail, da allein schon über verschiedene Rotverschiebungen unterschiedliche Auflösungen der Wellenlängen realisiert werden. Durch Interpolation kann hier aber eine einheitliche Auflösung erzeugt werden.

³Im Fall einer Menge mit geradzahlig vielen Elementen gibt es mehrere Möglichkeiten, den Median zu definieren. Manchmal wird gefordert, dass der resultierende Wert auch wirklich in der Menge vorkommt. Das ist für diese Zwecke nicht erforderlich, daher kann in diesem Fall mit dem arithmetischen Mittel der beiden mittleren Werte gearbeitet werden.

Sei zusätzlich zu jedem Anteil $q \in]0, 1]$ die folgende Funktion für (nichtnegative) Zahlenwert der Länge der Anzahl der positiv klassifizierten Spektren der Trainingsmenge definiert:

$$\max_q : (\mathbb{R}^{\geq 0})^{|T_p|} \rightarrow \mathbb{R}^{\geq 0}, \quad x = (x_1, x_2, \dots, x_{|T_p|}) \mapsto \pi_{\lceil q \cdot |T_p| \rceil}(\text{sort}(x)),$$

wobei sort ein Tupel auf eine entsprechend der totalen Ordnung von \mathbb{R} sortierte Version von sich selbst abbilde. Entsprechend ist \max_1 einfach das gewöhnliche Maximum und $\max_{\frac{1}{2}}$ eine Version des Median.

Mit dieser Funktion ist es elegant möglich, die Differenzmengen, auf die die dists -Funktionen abbilden, zu einem einzelnen Wert zusammenzufassen:

$$\begin{aligned} \text{dist}_{\emptyset, q} &= (\max_q \circ \text{dists}_{\emptyset}) : W \rightarrow \mathbb{R}^{\geq 0}, \quad w \mapsto \max_q(\text{dists}_{\emptyset}(w)) \\ \text{dist}_{m, q} &= (\max_q \circ \text{dists}_m) : W \rightarrow \mathbb{R}^{\geq 0}, \quad w \mapsto \max_q(\text{dists}_m(w)) \end{aligned}$$

Für $q = 1$ erhält man mit $\text{dist}_{\cdot, q}$ also Funktionen, die jeder Wellenlänge die maximale Distanz zur Referenzkurve innerhalb der Trainingsmenge an dieser Stelle zuordnen. Eventuelle lokale Ausreißer können mit einer kleineren Wahl für q ignoriert werden. Die Funktionen $\text{dist}_{\cdot, 9/10}$ etwa liefern zu jeder Wellenlänge den am weitesten von der Referenzkurve entfernten Wert, wenn man die am weitesten entfernten 10% ignoriert.

Um eine solche Funktion als Grundlage der Berechnung von lokализierten Fréchet-Distanzen nutzen zu können, bedarf es einer Abbildung $wl : \{1, \dots, n\} \rightarrow W$ von Segmentadressen auf Wellenlängen, die zur Verwendung von Spektren als polygonale Kurven ohnehin benötigt wird. Durch Komposition mit den obigen Funktionen erhält man dann die folgenden Distanzlokalisierungen:

$$\begin{aligned} (\text{dist}_{\emptyset, q} \circ wl) &: \{1, \dots, n\} \rightarrow \mathbb{R}^{\geq 0}, \quad i \mapsto \max_q(\text{dists}_{\emptyset}(wl(i))) \\ (\text{dist}_{m, q} \circ wl) &: \{1, \dots, n\} \rightarrow \mathbb{R}^{\geq 0}, \quad i \mapsto \max_q(\text{dists}_m(wl(i))) \end{aligned}$$

Diese Distanzlokalisierungen nehmen genau an den Wellenlängen große Werte an, an denen es innerhalb der positiv klassifizierten Spektren der Trainingsmenge große Abweichungen vom Referenzspektrum gibt. Da die Fréchet-Distanz (und ihre Varianten) aber gerade dadurch ausgezeichnet ist, dass die punktweisen Distanzen zwischen Punkten gemessen werden, die durch Variation der Parametrisierungen in ihrer Position auf den Segmenten der Kurven abweichen können, erscheint dieser Ansatz noch nicht optimal (siehe Abbildung 3.5). Ein direkt aus der Fréchet-Distanz abgeleiteter Ansatz zur Bestimmung einer besseren Distanzlokalisierung soll hier Abhilfe schaffen.

3.3.2 Ein divide-and-conquer-Ansatz zur Lokalisierung auf Basis der Fréchet-Distanz

Die zentrale Frage, die man sich hier stellt, ist nämlich im Kontrast zur oben diskutierten Methode nicht die Frage, wie weit die Kurven auf gemeinsamen Segmentadressen $i \in \{1, \dots, n\}$ durch die Abstände der Segmente $T_1|_{[i-1, i]}$ und $T_2|_{[i-1, i]}$ voneinander abweichen⁴, sondern lediglich, auf welchen Segmenten der Referenzkurve welche maximalen punktweisen Distanzen realisiert werden. Das lässt in Relation dazu eine Variation der Parametrisierung der anderen Kurve zu, wie sie für die Berechnung einer Fréchet-Distanz-Variante typisch ist.

⁴Dies natürlich unter der Annahme, dass die zu vergleichenden polygonalen Kurven die gleiche Anzahl von Segmenten haben, was aber im Kontext des Vergleichs von Spektren eine einfach zu realisierende Voraussetzung ist.

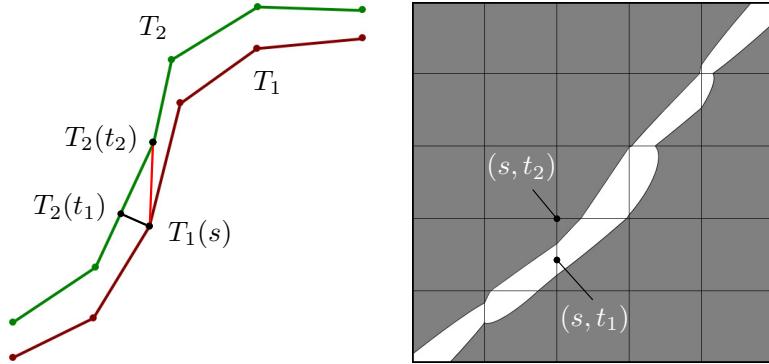


Abbildung 3.5: Besonders in Bereichen mit großer Steigung unterscheidet sich die Fréchet-Distanz von Spektren wesentlich von der punktweisen Distanz. Der Abstand der vom Parameterpaar (s, t_1) induzierten Punkte auf T_1 und T_2 ist deutlich geringer als der vom Paar (s, t_2) induzierte. Letzterer entspricht der punktweisen Distanz $d(T_1(s), T_2(t_2))$. Eine auf punktweisen Distanzen basierende Distanzlokalisierung wäre also in Bereichen großer Steigung viel toleranter als eigentlich nötig.

In Abbildung 3.5 wird deutlich, dass die punktweise Distanzmessung Aussagen entlang der Diagonalen mit gleichen Punkt- und Segmentadressen im Freespace-Diagramm trifft ($(s, t_2) = (s, s)$). Was man aber für eine Distanzlokalisierung lediglich benötigt, ist eine Aussage darüber, welche maximalen Distanzen in Relation zur Referenzkurve gelten, also in welchen Spalten des Freespace-Diagramms welche maximalen punktweisen Distanzen gelten sollen.

Die Beantwortung dieser Frage kann man ebenfalls durch die Berechnung von Fréchet-Distanz-Varianten realisieren, wie die hier folgende Diskussion zeigt. Der vorzustellende divide-and-conquer-Ansatz beruht auf der Zielformulierung, *scharfe Distanzlokalisierungen* bezüglich einer Referenzkurve und einer weiteren zu erzeugen:

Definition 3.10 Eine Distanzlokalisierung dl_{T_1} einer Kurve $T_1 : [0, n] \rightarrow \mathbb{R}^2$ heißt **scharf bezüglich einer weiteren Kurve T_2** , falls sie an jeder Stelle $i \in \{1, \dots, n\}$ minimal ist mit der Eigenschaft, dass das zugehörige lokalisierte Fréchet-Distanz-Entscheidungsproblem positiv gelöst werden kann, also $\delta_F(T_1, T_2, dl_{T_1}) \leq 1$ gilt.

Scharfe Distanzlokalisierungen besitzen die Eigenschaft, bezüglich einer unbekannten Kurve auf den Segmenten der Referenzkurve tolerant zu sein, auf denen auch große Distanzen zwischen den bei der Erzeugung beteiligten Kurven realisiert wurden und umgekehrt dort große Abweichungen sensibel zu erkennen, wo diese Kurven sich stark ähnelten. Sie sind damit die idealen Bausteine für eine auf einer Trainingsmenge von Spektren beruhende Distanzlokalisierung, die Wellenlängen großer bzw. kleiner Abweichungen treffsicher widerspiegelt.

Zur Berechnung einer scharfen Distanzlokalisierung bezüglich einer Referenzkurve T_1 und einer weiteren Kurve T_2 kann man sich zunächst intuitiv überlegen, dass die Spalten des Freespace-Diagramms, in denen die Fréchet-Distanz dieser Kurven realisiert werden, die Maxima der Distanzlokalisierung repräsentieren, denn dort wird der maximale punktweise Abstand (nämlich die Fréchet-Distanz selbst) benötigt, um das Entscheidungsproblem zu lösen. Wäre der Wert einer scharfen Distanzlokalisierung in einer anderen Spalte höher, so hätte der vorher betrachtete Wert nicht die Fréchet-Distanz sein können.

Als kleines Hilfsmittel wird nun informell eine Variante der lokalisierten Fréchet-Distanz eingeführt, die sich von der lokalisierten nur dadurch unterscheidet, dass einzelne Segmente der Referenzkurve als *fixiert* ausgezeichnet werden können, also bei der Skalierung einer Distanzlokalisierung nicht verändert werden. Die für konkrete Berechnungen benötigten Begriffe der freien gemeinsamen Zellenrandstücke und des erreichbaren Raumes werden dadurch nicht berührt. Für nicht erwähnte Spalten sei die Distanzlokalisierung standardmäßig auf 1 festgelegt, um dort keine Unterschiede zur Standard-Fréchet-Distanz zu generieren. Im weiteren Verlauf dieses Abschnitts wird implizit diese Variante verwendet.

Mit diesem Hilfsmittel lässt sich die folgende divide-and-conquer-Strategie zur Ermittlung einer scharfen Distanzlokalisierung zu einer Referenzkurve T_1 und einer weiteren Kurve T_2 formulieren:

Wird an einer „kritischen Stelle“ im Freespace-Diagramm die Fréchet-Distanz von T_1 und T_2 als kritischer Wert realisiert, so erhält man eine scharfe Distanzlokalisierung von T_1 und T_2 als Kombination der scharfen Distanzlokalisierungen der Teildiagramme links unterhalb und rechts oberhalb der der betrachteten Stelle samt Fixierung der Spalte(n) an der kritischen Stelle auf den kritischen Wert.

Die Aufteilung des Diagramms für einen kritischen Wert vom Typ II ist in Abbildung 3.6 illustriert. Eine solche Strategie impliziert ein rekursives Vorgehen und das Vorhandensein eines erreichbaren Rekursionsanfangs. Das oben beschriebene Verfahren kann aber für ein Teildiagramm abgebrochen werden, sobald alle zu berücksichtigenden Spalten fixiert sind. Aus der Endlichkeit der Spalten und der Endlichkeit der überhaupt zur Verfügung stehenden kritischen Werte folgt aber sofort die Fixierbarkeit aller Spalten (in jeder Spalte werden zumindest die kritischen Werte vom Typ I bezüglich des zugehörigen Segments realisiert) und damit die Gültigkeit dieser Strategie.

Wie die Unterteilung des Diagramms in Teildiagramme bzw. die Unterteilung der Kurven in Teilkurven aber konkret aussehen kann, soll im Folgenden diskutiert werden.

Die Fréchet-Distanz der Kurven wird immer durch kritische Werte nach Abschnitt 2.2.5 (im Standardfall) bzw. Lemma 3.9 (im lokalisierten Fall) realisiert. Für den Moment sei angenommen, dass solche kritischen Werte immer eindeutig sind, also nur an einer Stelle im Freespace-Diagramm auftreten können⁵. Wenn ein solcher kritischer Wert also die Fréchet-Distanz zweier Kurven realisiert, dann kann man ihm eindeutig einen Typ und eine Position im Freespace-Diagramm zuordnen.

Je nach Typ des kritischen Werts lassen sich eine oder mehrere Spalten des Freespace-Diagramms dann mit diesem kritischen Wert fixieren, denn laut Definition der Fréchet-Distanz gibt es keine Parametrisierungen, die das Entscheidungsproblem für die selbe Distanz lösen *und* in diesen Spalten eine kleinere punktweise Distanz realisieren. Folgende Typen können auftreten:

- Die Distanz ist der punktweise Abstand, für den Start- bzw. Endpunkt gerade eben noch frei sind (Typ I). In dem Fall wird die erste bzw. letzte Spalte auf den ermittelten kritischen Wert fixiert und die resultierende Fréchet-Distanz für das ganze Diagramm neu berechnet.

⁵Das ist im Allgemeinen nicht der Fall, spielt für das grundlegende Verständnis der hier präsentierten Idee aber zunächst keine Rolle. Weiter unten wird auf den allgemeinen Fall mit nichtein-deutigen kritischen Werten eingegangen

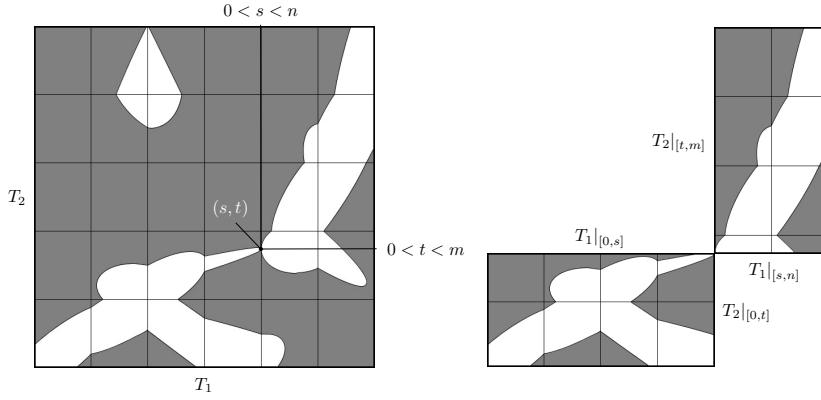


Abbildung 3.6: Eine Freespace-Diagramm-Aufteilung für einen eindeutigen kritischen Wert δ des Typs 2 (horizontaler Zellenübergang) an der Stelle (s, t) . Zur Suche der nächstkleineren relevanten kritischen Werte werden die benachbarten Spalten $s - 1$ und s auf den kritischen Wert δ fixiert; mit dieser Fixierung werden die Fréchet-Distanzen zu den Teilkurven $T_1|_{[0,s]}$ und $T_2|_{[0,t]}$ sowie $T_1|_{[s,n]}$ und $T_2|_{[t,m]}$ und damit die scharfen Teildistanzlokalisierungen berechnet.

- Die Distanz ist der punktweise Abstand, für den ein freies Zellenrandstück $L_W^{i,j}$ bzw. $B_W^{i,j}$ gerade eben nicht leer ist, also aus genau einem Punkt (s, t) besteht (Typ II). Im Fall von $L_W^{i,j}$ werden die Spalten $i - 1$ und i auf den ermittelten kritischen Wert fixiert, im Fall von $B_W^{i,j}$ nur die Spalte i . Die Teildiagramme für die Teilkurven $T_1|_{[0,s]}$ und $T_2|_{[0,t]}$ sowie $T_1|_{[s,n]}$ und $T_2|_{[t,m]}$ sind dann Grundlage der Berechnung der scharfen Teildistanzlokalisierungen (Abbildung 3.6).
- Die Distanz ist der punktweise Abstand, für den eine horizontale oder vertikale Passage zwischen freien Zellenrandstücken $L_W^{i_1,j}$ und $L_W^{i_2,j}$ bzw. B_W^{i,j_1} und B_W^{i,j_2} existiert (Typ III), realisiert durch die Punkte $p_1 := (s_1, t_1)$ und $p_2 := (s_2, t_2)$. Im Fall einer horizontalen Passage gilt $t_1 = t_2$ und die Spalten $i \in \{i_1 - 1, \dots, i_2\}$ werden auf den ermittelten kritischen Wert fixiert, im Fall einer vertikalen Passage ($s_1 = s_2$) nur die Spalte i . Die scharfen Teildistanzlokalisierungen werden dann für die Teilkurven $T_1|_{[0,s_1]}$ und $T_2|_{[0,t_1]}$ sowie $T_1|_{[s_2,n]}$ und $T_2|_{[t_2,m]}$ berechnet (siehe Abbildung 3.7).

Bei der Kombination scharfer Teildistanzlokalisierungen ist zu beachten, dass in einem Unteraufruf bereits fixierte Spalten nicht mehr durch kleinere Werte überschrieben werden dürfen, da der ursprünglichen Wert zur Realisierung einer Fréchet-Distanz im jeweiligen Schritt groß genug sein muss.

Bei den obigen Ausführungen wurde angenommen, dass ein kritischer Wert, der als Fréchet-Diagramm von (Teil-)Kurven realisiert wird, einen eindeutigen Typ und eine eindeutige Position im Diagramm hat. Dies ist im Allgemeinen nicht der Fall, wodurch das Problem der Generierung einer scharfen Distanzlokalisierung nicht mehr eindeutig lösbar ist. Trotzdem kann man bei Mehrfachrealisierung der Fréchet-Distanz einfach eine Stelle auswählen und für die oben vorgestellte divide-and-conquer-Strategie nutzen, um eine scharfe Distanzlokalisierung zu berechnen. Ein vergleichbares Vorgehen wird in [BBMS] vorgestellt um *lokal korrekte Parametrisierungen* für zwei polygonale Kurven zu finden im Gegensatz zum hier präsentierten

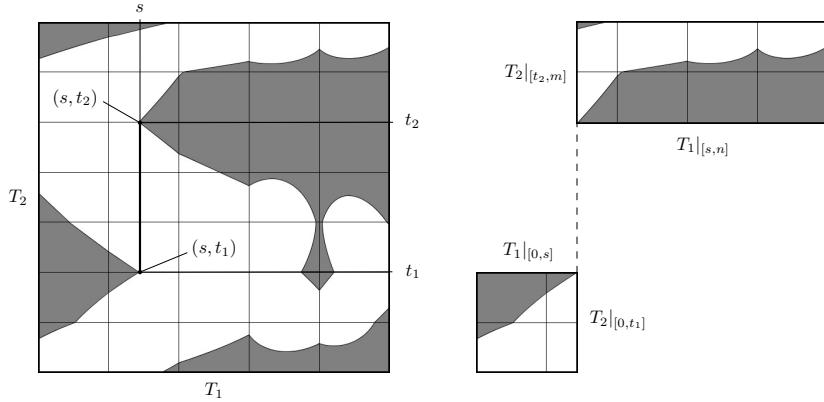


Abbildung 3.7: Eine Freespace-Diagramm-Aufteilung für einen eindeutigen kritischen Wert δ des Typs 3 (vertikale zeilenübergreifende Passage) zwischen den Stellen (s, t_1) und (s, t_2) . Zur Suche der nächstkleineren relevanten kritischen Werte wird die beteiligte Spalte $[s]$ auf den kritischen Wert δ fixiert; mit dieser Fixierung werden die Fréchet-Distanzen zu den verbleibenden Teilkurven $T_1|_{[0, s]}$ und $T_2|_{[0, t_1]}$ sowie $T_1|_{[s, n]}$ und $T_2|_{[t_2, m]}$ und damit die scharfen Teildistanzlokalisierungen berechnet.

Ziel der konkreten Berechnung einer optimalen Distanzlokalisierung für die lokalisierte Fréchet-Distanz.

Die durch nichteindeutige Realisierungen einer Fréchet-Distanz zweier Kurven implizierte nichteindeutige Berechnung von scharfen Distanzlokalisierungen legt die Formulierung eines Gütebegriffs für scharfe Distanzlokalisierungen nahe. Um mehrere Teillösungen dieses Problems vergleichen zu können, müssen aber zunächst bei der Berechnung alle realisierenden kritischen Stellen im Diagramm berücksichtigt werden. Dies wird in Abbildung 3.8 illustriert.

Ein Problem, das dabei auftreten kann, ist die Mehrfachberechnung der scharfen Distanzlokalisierung für ein Teildiagramm, was aber mit Hilfe von Memoisation gelöst werden kann. Dazu ist es zunächst nötig, die Berechnung der scharfen Distanzlokalisierung für zwei Teilkurven als (referentiell transparente) Funktion zu formulieren. Unter Berücksichtigung der Argumente dieser Funktion wird dann das Ergebnis dieser Berechnung in einer Tabelle gespeichert und kann bei eventuellem Neuaufruf in konstanter Zeit zurückgeliefert werden. Aufgrund der Anzahl überhaupt möglicher kritischer Werte und der damit verbundenen Begrenztheit möglicher Funktionsaufrufe kann diese Tabelle auch nicht beliebig groß werden.

Bei der Rekombination der scharfen Distanzlokalisierungen für mehrere Unteraufrufe bei gleichen kritischen Werten kann man sich dann anhand des erwähnten Gütekriteriums für eine Belegung entscheiden, etwa über eine lexikalische Sortierung anhand der sortierten Werte der Distanzlokalisierungen. Damit wäre sichergestellt, dass eine mit diesem Verfahren ermittelte scharfe Distanzlokalisierung zusätzlich nach diesem Kriterium optimal ist.

Die Kombination mehrerer dieser scharfen Distanzlokalisierungen entsprechend der als positiv klassifizierten Spektren aus einer Trainingsmenge würde schließlich in einer gemeinsamen Distanzlokalisierung münden, die Abweichungen und Ähnlichkeiten von typischen Kurvenverläufen innerhalb der Trainingsmenge, repräsentiert durch eine Referenzkurve, zuverlässig erkennt.

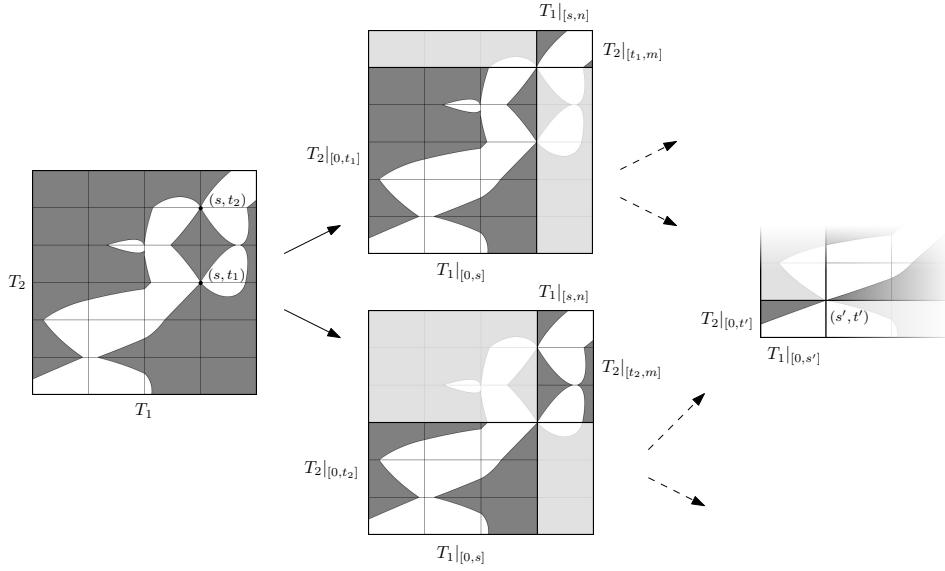


Abbildung 3.8: Wird eine Fréchet-Distanz durch einen kritischen Wert an mehr als einer Stelle realisiert (links), so können für beide möglichen Unterteilungen scharfe Distanzlokalisierungen für die resultierenden Teilkurven berechnet werden (mitte). Bei der Rekombination dieser Lokalisierungen kann man sich entsprechend eines vorher definierten Gütekriteriums für eine Belegung entscheiden um insgesamt ein nach diesem Kriterium optimales Ergebnis zu erhalten. Bei diesem Ansatz kann es vorkommen, dass scharfe Distanzlokalisierungen für Teildiagramme mehrfach berechnet werden müssen (rechts), was durch Memisation bezüglich Start- und Endpunkt der Teildiagramme bzw. der zugehörigen Teilkurven aber umgangen werden kann.

Eine genauere Untersuchung dieses interessanten Verfahrens sprengt den zeitlichen Rahmen der vorliegenden Arbeit. Schließend kann man aber zusammenfassen, dass die vorgestellten Werkzeuge zur Wellenlängen-sensiblen Messung von Fréchet-Distanzen zwischen astronomischen Spektren sehr passgenau erscheinen und durch die verschiedenen Kombinationsmöglichkeiten zu einer Referenzkurve bzw. einer gemeinsamen „ausreichend scharfen“ Distanzlokalisierung ein großes Potential zur problemspezifischen Anpassung bieten.

Kapitel 4

Eine Implementierung von Fréchet-Distanz-Varianten für die astronomische Datenanalyse

Ein wesentlicher Teil meiner Diplomarbeit besteht in der Implementierung der in dieser schriftlichen Ausarbeitung diskutierten Methoden zur Messung von Fréchet-Distanzen zwischen astronomischen Spektren. Diese Implementierung dient nicht nur der Verifikation der diskutierten Algorithmen, sondern lässt sich ganz konkret in der astronomischen Datenanalyse einsetzen, denn die implementierten Klassen sind eingebettet in das Framework Clastro, an dessen Entstehung ich im Rahmen eines Projektseminars sowie in späteren Weiterentwicklungsphasen als studentische Hilfskraft maßgeblich beteiligt war. Clastro selbst ist kein Bestandteil des praktischen Teils dieser Diplomarbeit, die Interaktion mit dem System wird hier dennoch beschrieben, um die Integration der Fréchet-Distanz-Messung in die astronomische Anwendung zu verdeutlichen.

Im ersten Abschnitt dieses Kapitels wird die Architektur und Funktionalität des Frameworks kurz diskutiert, um die Einbettung der für diese Arbeit implementierten Methoden in Clastro vorzustellen. Die dazu verwendeten vom restlichen Framework abgrenzbaren Klassen werden im zweiten Abschnitt dieses Kapitels vorgestellt.

Ein wichtiges Merkmal der Java-Implementierung von Clastro und der für diese Arbeit realisierten Methoden ist der Einsatz testgetriebener Entwicklung. Durch den modularen Aufbau war es möglich, alle relevanten Methoden durch den Einsatz umfangreicher Unit-test-Suites automatisiert verifizierbar zu machen und ihre Spezifikationen noch vor Fertigstellung der Implementierung durch JUnit-Testscases festzuschreiben. Auch Integrationstests sind Bestandteil der großen Testsuite und definieren und sichern das Zusammenspiel der einzelnen Komponenten. Obwohl durch den Einsatz solcher Tests nicht die Korrektheit von Programmcode bewiesen werden kann, hat sich das automatisierte Testen und die Entwicklung entlang bereits identifizierter kritischer Fälle bewährt und ließ nicht nur die frühzeitige Erkennung von Fehlern, sondern auch augenblickliche Testläufe des gerade entwickelten Codes ohne eigens entworfene Programme zu.

Die Abwägung zwischen dem Einsatz von bewährten Bibliotheken zur Lösung von für sich genommen nichttrivialen Problemen und der Entwicklung von passgenauen Individuallösungen war und ist ein zentrales Thema beim Entwurf von Clastro

und den für diese Arbeit entwickelten Klassen. Während für das Framework selbst einige Fremdlösungen eingesetzt werden (etwa für das Training von Support Vector Machines oder die Visualisierung von Spektren), haben sich die für diese Arbeit relevanten Probleme als sehr spezifisch herausgestellt und wurden komplett durch selbstentwickelte Klassen gelöst.

4.1 Clastro: ein Framework zur Klassifikation astronomischer Objekte aus dem SDSS

Clastro ist eine Sammlung von Werkzeugen zur komfortablen Suche von Spektren aus dem Sloan Digital Sky Survey (SDSS) sowie ihrer Weiterverarbeitung und Klassifikation anhand verschiedener Merkmale. Es werden verschiedene Methoden zur visuellen Verifikation bereitgestellt, die sich über eine zentrale Benutzerschnittstelle in Form einer gut dokumentierten Eingabeaufforderung bedienen lassen. Die in Clastro verwendeten Klassen implementieren im Wesentlichen, aber nicht ausschließlich, die in Kapitel 1 vorgestellten Methoden. Im Projektbericht [OCSW12b] wird näher auf Details zu Design und Implementierung eingegangen.

4.1.1 Repräsentation astronomischer Objekte

Im internen Datenspeicher von Clastro werden astronomische Objekte durch Objekte der Klasse `StellarObject` repräsentiert, die im wesentlichen ein `Spectrum`-Objekt sowie einige Metadaten (etwa zur Beobachtungsposition am Himmel sowie zur Rotverschiebung), aber auch photometrische Daten beinhalten. Die `Spectrum`-Objekte wiederum sind der zentrale Bestandteil der Datenverarbeitung in Clastro, denn sie enthalten die Daten, die im Teil des maschinellen Lernens durch Merkmale repräsentiert stellvertretend für die astronomischen Objekte klassifiziert werden. Um die Verarbeitung effizient zu gestalten und Heap-Overflows zu vermeiden, verwenden `Spectrum`-Objekte intern Felder von `double`-Zahlen. Ferner enthalten Sie Hilfsmethoden zur Zuordnung und Umrechnung von Wellenlängen.

Die Rohdaten der Spektren sind aber nur Ausgangspunkt einer Verarbeitungskette für die Klassifikation. Mit Hilfe verschiedener Methoden werden von den Spektren vereinfachte und bereinigte Versionen berechnet und schließlich ein Kontinuum extrahiert, auf dessen Grundlage viele Merkmale berechnet werden. Alle diese Daten werden in Objekten der Klasse `FeaturedObject` um die den Daten aus der Datenquelle entsprechenden `StellarObjects` dekoriert. Durch den Einsatz von *lazyness* werden diese Spektren jedoch erst zum letztmöglichen Zeitpunkt berechnet. Durch verschiedene Abhängigkeiten von Parametern dieser Berechnungen kann es bei Modifikationen zu aufwendigen Update-Kaskaden kommen, die aber alle vom Benutzer der Objekte verborgen sind. Ein Spezialfall von diesen Objekten sind die `ClassifiedFeaturedObjects`, die zusätzlich über eine Information verfügen, ob das zugehörige Objekt entsprechend bestimmter Kriterien als positiv oder negativ klassifiziert wurde. Die Objekte von Trainingsmengen etwa liegen als `ClassifiedFeaturedObjects` vor.

Schließlich wird die mehrfache Verarbeitung von solchen Objekten durch die Verwendung entsprechender `-Set`-Varianten dieser Klassen ermöglicht. Alle Objekte der hier erwähnten Klassen (mit Ausnahme der Spektren selbst) werden unter einem eindeutigen Namen im internen Datenspeicher von Clastro vorgehalten und können mit Hilfe der Benutzerschnittstelle zur Laufzeit inspiert werden.

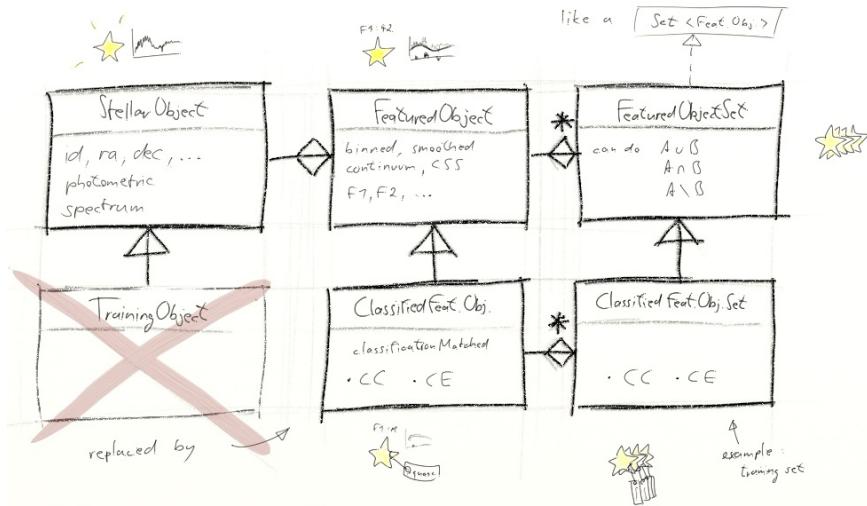


Abbildung 4.1: Repräsentation von astronomischen Objekten in Clastro: ein **StellarObject** kapselt ein Spektrum bestehend aus den Rohdaten aus dem SDSS sowie eine Auswahl der verfügbaren Metadaten. Daraus abgeleitete Spektren werden in einem **FeaturedObject** gesammelt und um das **StellarObject** „herumdekoriert“. Bereits klassifizierte Objekte sind **ClassifiedFeaturedObjects** und mehrere dieser Objekte können in entsprechenden Mengenobjekten gemeinsam verarbeitet werden. Bildquelle: [OCSW12b, Abschnitt 4.1]

4.1.2 Datentransport in Clastro

Es gibt in Clastro im Wesentlichen zwei Methoden, Spektren astronomischer Objekte zur Weiterverarbeitung zu erhalten. Der Import- und Export von CSV-Dateien für einzelne Objekte und ganze Objektmengen ist nützlich, wenn Arbeitsbeispiele für spätere Verwendung ohne Anbindung an die Datenbank vorgehalten werden sollen. Eine spezifische Suche nach Objekten wird aber nur über die Datenbank realisiert. Anfangs arbeitete Clastro dazu mit einem MSSQL-Server bei der TU-Dortmund, wo eine geeignete Teilmenge vom Data Release 6 des SDSS gespiegelt war. In der jüngeren Zeit wurde an der WWU Münster eine geeignete Teilmenge vom aktuellen Data Release 9 des SDSS gespiegelt, was den Zugriff auf mehr Spektren sowie eine schnellere Transferzeit ermöglicht.

Zunächst war es geplant, ein erprobtes ORM-Framework zur Interaktion mit der relationalen Datenbank einzusetzen. Es hat sich allerdings herausgestellt, dass eine eigens programmierte Lösung mit genau den richtigen Möglichkeiten für die aus technischer Sicht sehr übersichtlichen Interaktionsmöglichkeiten vollkommen ausreichte. Ein besonderer Vorteil der eingesetzten Lösung ist die Möglichkeit, eine Suche nach Objekten zwischenspeichern und schrittweise zu modifizieren, etwa bis eine gut handhabbare Anzahl von Objekten für einen einmaligen Transfer selektiert ist.

4.1.3 Training und Klassifikation

Anhand der in den vorigen Abschnitten erläuterten Methoden ist es möglich, eine definierte Menge von Objekten aus CSV-Dateien zu importieren und zusätzlich die

Information, welche Objekte als positiv klassifiziert sind, im Speicher zu halten. Auf diese Weise werden die in Zusammenarbeit mit Astrophysikern erstellten Trainingsmengen in Clastro verfügbar gemacht. Um einen Klassifizierer entsprechend einer Trainingsmenge zu erzeugen, müssen genaue Informationen darüber bekannt sein, mit welchen Parametern die für die Klassifizierung verwendeten Spektren vorverarbeitet werden und welche Merkmale für den Merkmalsraum ausgewählt werden sollen. Diese Informationen werden vom Benutzer ausgewählt und ermöglichen in Zusammenhang mit einer späteren Qualitätsmessung die schrittweise Verbesserung von Klassifizierern.

In einer `ClassifierFactory` werden entsprechend der übergebenen Parameter und Merkmalsmengen kNN- oder SVM-Klassifizierer trainiert wie in Kapitel 1 beschrieben. Zur sofortigen Qualitätsbeurteilung wird von der `ClassifierFactory` ebenfalls eine Kreuzvalidierung auf Basis der Trainingsmenge durchgeführt. Daraus ergeben sich neben dem MCC als Qualitätsmaß auch Mengen von Objekten, die in der Kreuzvalidierung als falsch-positiv oder falsch-negativ klassifiziert wurden. Diese Objektmengen können dem Benutzer wertvolle Hinweise über eventuelle Ungenauigkeiten der Trainingsmenge oder schlecht gewählte Parameter geben.

Ein so generierter Klassifizierer kann in Clastro, einmal trainiert, serialisiert und immer wieder verwendet werden. Er transformiert die Objekte einer zu übergebenen Menge von zu klassifizierenden Objekten entsprechend der beim Training übergebenen Transformationsparameter und entscheidet zu jedem Objekt, ob es entsprechend der beim Training gewählten Fragestellung als positiv oder negativ angesehen wird.

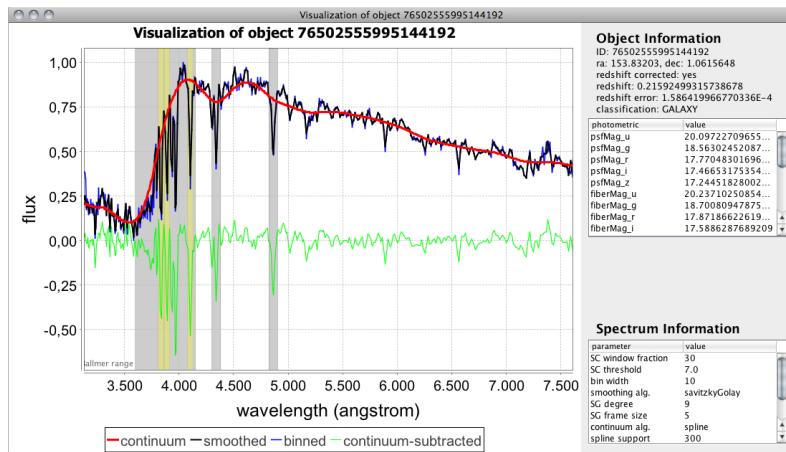


Abbildung 4.2: Die Visualisierung eines einzelnen Objekts. Im Bild sichtbar sind neben einem leicht vereinfachten Rohspektrum (blau) auch die geglättete Version (schwarz) sowie das extrahierte Kontinuum (rot) und die Differenz des geglätteten Spektrums vom Kontinuum. In der rechten Spalte sind Photometriedaten des Objekts und Transformationsparameter sichtbar.

4.1.4 Die Benutzerschnittstelle

Die von Clastro verwendete Benutzerschnittstelle ist im Wesentlichen eine komfortable Eingabeaufforderung, mit der sich aber auch GUI-Komponenten laden lassen. Das Ziel bei der Entwicklung war, alle Funktionen von Clastro in einer möglichst

leichtgewichtigen und einfach zu erweiternden Oberfläche zu vereinen. Über diese Eingabeaufforderung können auch die Objekte im internen Zwischenspeicher inspiert und ggf. modifiziert werden. Das Suchkommando etwa generiert und modifiziert gekapselte Datenbank-Querys ohne dass der Benutzer über SQL-Kenntnisse verfügen muss. Mit dem Visualisierungskommando können je nach Kontext einzelne Objekte oder ganze Objektmengen visualisiert werden. Die Visualisierung eines einzelnen Objekts etwa generiert eine Grafik, die alle für die zuletzt angewandten Parameter generierten Spektren des Objekts enthält (siehe Abbildung 4.2).

Für die Visualisierung von Objektmengen gibt es die Möglichkeit, einzelne Grafiken im Dateisystem zu speichern oder alle gemeinsam in einer Grafik anzuzeigen (siehe Abbildung 3.1). Ferner gibt es zur Unterstützung von Astrophysikern ein Tool, mit dem man schnell eine Objektmenge in zwei Teilmengen aufspalten kann. Zur Beurteilung von Merkmalen kann man sich Trainingsmengen in zweidimensionale Streudiagramme für zwei Merkmale einzeichnen lassen (siehe Abbildung 1.12 auf Seite 16).

4.2 Implementierung von Fréchet-Distanz-Varianten in Clastro

Die Realisierung der in dieser Ausarbeitung vorgestellten Methoden verlief in mehreren Abschnitten. Als Grundlage aller Berechnungen sowie für die Tests der Fréchet-Distanz-spezifischen Programmteile diente dabei eine ausführliche Implementierung verschiedener geometrischer Grundlagen in Form von Klassen für Punkte und Linien bzw. Liniensegmenten. Das Fréchet-Distanz-Entscheidungsproblem sowie die Distanzberechnung selbst wurden schließlich exakt an den mathematischen Definitionen entlang in Form von Freespace-Diagrammen zusammen mit ihren jeweiligen Zellen implementiert. Wie in Kapitel 3 diskutiert kann die Standard-Fréchet-Distanz als Spezialfall der lokalisierter Variante behandelt werden. So wurde der Code im zeitlichen Verlauf so modifiziert, dass alle implementierten Varianten der Fréchet-Distanz mit Hilfe einer Distanzlokalisierung berechnet werden, die in den Standardfällen eben konstant ist.

Zur optischen Verifizierung der Korrektheit schließlich wurde eine grafische Benutzerschnittstelle entworfen, in der sich interaktiv polygonale Kurven erstellen und verändern lassen und in der der Zusammenhang mit dem zugehörigen Freespace-Diagramm in Echtzeit verdeutlicht wird. Die meisten der Visualisierungen von Freespace-Diagrammen in dieser Ausarbeitung wurden von dieser Komponente erzeugt.

4.2.1 Geometrische Grundlagen

Obwohl sich die Aufgaben der Klassen nicht vollständig trennen lassen, kann man doch grob zusammenfassen, dass die geometrischen Grundlagen der Berechnung von Fréchet-Distanzen im Paket `geometry` implementiert wurde, während die konkreten Berechnungen mit Freespace-Diagrammen und kritischen Werten im Paket `geometry.frechet` organisiert sind. In diesem Abschnitt werden die allgemeinen geometrischen Operationen vorgestellt, die zwar für die weiteren Berechnungen nötig, aber noch allgemein nutzbar sind.

Punkte und Vektoren zugleich: die Klasse Point

Die naheliegendste Realisierung von Punkten (x, y) in der Ebene \mathbb{R}^2 erhält man in Java als die Zusammenfassung zweier `double`-Werte für die einzelnen Koordinaten. Aufgrund der Verwendung von Rechnerarithmetik hat man es hier mit einer äußerst kleinen (da endlichen) Teilmenge dieses Raumes zu tun, die meisten Operationen lassen sich jedoch für reale Anwendungsfälle genau genug durchführen. Da Punkte in der Ebene und zweidimensionale \mathbb{R} -Vektoren als hinreichend gleich¹ aufgefasst werden können, erhalten `Point`-Objekte eine Doppelrolle. Sie können nicht nur Punkte repräsentieren, sondern auch wie die zugehörigen Ortsvektoren addiert und skalar multipliziert werden. Diese Mehrdeutigkeit erwies sich zu keinem Zeitpunkt als problematisch.

Aus den aus dieser Eigenschaft erwachsenden Operationen Addition `add(Point)` und skalare Multiplikation `scale(double)` lassen sich weitere Operationen wie Subtraktion `subtract(Point)` und Invertierung `inverse()` ableiten. Diesen Operationen ist gemein, dass sie stets als Ergebnis der Operation ein neues `Point`-Objekt zurückliefern, anstatt das benutzte Objekt per Seiteneffekt zu modifizieren, um bei Berechnungen anhand von Punkten, die bereits in weiteren Objekten in Verwendung sind, keine versteckten Änderungen zu verursachen.

Aus der Eigenschaft, die Computerarithmetik-Näherung eines metrischen Raumes zu sein, erwächst die Notwendigkeit, eine Distanz zwischen zwei Punkten berechnen zu können. Verschiedene Metriken sind hier denkbar, für die Implementierung der hier diskutierten Algorithmen wurde jedoch stets die Euklidische Metrik eingesetzt. Das Ergebnis ist eine Methode `distance(Point)`, die die Euklidische Distanz zwischen dem zugrundeliegenden `Point`-Objekt und dem übergebenen Objekt berechnet. Daraus abgeleitet gibt es zu jedem `Point`-Objekt stets auch die Länge des zugehörigen Ortsvektors `length`, indem man die Distanz zum Punkt $(0, 0)$ berechnet.

Die komplexeste Methode schließlich, die mit `Point`-Objekten direkt durchführbar ist, ist die Berechnung der Distanz zu einem Liniensegment implementiert durch die Methode `segmentDistance(Point, Point)`, wobei das Liniensegment durch die zwei als Argument übergebenen Punkte repräsentiert wird. Als Hilfsmethoden wird dazu noch das Skalarprodukt `scalarProduct(Point)` verwendet. Die Methode `segmentDistance(Point, Point)` berechnet den Abstand des Punktes zu der induzierten Linie nur, falls dieser innerhalb des Liniensegmentes realisiert wird. Ansonsten wird die Distanz zum näherliegenden Punkt zurückgeliefert. Die Methode stützt sich dabei auf eine andere Methode `lineDistance(Point, Point)` ab, die immer den Abstand zur durch die zwei Punkte induzierten Linie berechnet. Die letztgenannten Methoden werden in der als nächstes zu dokumentierenden Klasse gespiegelt:

Linien und Abschnitte davon: die Klasse LineSegment

Ebenfalls eine Doppelrolle können Objekte der Klasse `LineSegment` übernehmen. Obwohl durch zwei bei der Konstruktion zu übergebene Punkte die Parametrisierung sowie Start und Ende eindeutig festgelegt sind, wurde zusätzlich die Möglichkeit berücksichtigt, die induzierte Linie durch die beiden Punkte als ganzes zu betrachten, wie dies auch in der Anwendung zur Behandlung von polygonalen Kurven gelegentlich vorgesehen ist.

¹bis auf Isomorphie

Durch die eindeutig vergebenen Start- und Endpunkte a und b wird festgelegt, dass es sich um eine Linie mit folgender Parametrisierung handelt:

$$\ell : \mathbb{R} \rightarrow \mathbb{R}^2, \quad t \mapsto a + t(b - a),$$

also das Liniensegment durch das Einheitsintervall I mit $\ell(0) = a$ und $\ell(1) = b$ parametrisiert wird. Diese Parametrisierung wird durch die Methoden `eval(double)` und `getParameter(Point)` widergespiegelt. Während die erste Methode zu einem gegebenen Parameter den zugehörigen Punkt auf der Linie zurückliefert, versucht die zweite Methode, den Parameter für einen zu übergebenen Punkt zu berechnen. Liegt dieser Parameter nicht auf der Linie, so muss eine Ausnahme ausgelöst werden.

Für dazu genutzte Tests, ob Punkte auf der induzierten Linie liegen, musste aufgrund der verwendeten Computerarithmetik eine gewisse Toleranz eingeführt werden, denn ansonsten sind Fälle konstruierbar, in denen in einem bestimmten Bereich kein Punkt auf der Linie repräsentiert werden kann. Dies wird durch eine Klassenvariable `EPSILON` realisiert, die bei dem Test, ob ein Punkt auf der induzierten Linie liegt, als obere Schranke für die im vorigen Abschnitt besprochene Punkt-Liniendistanz dient.

Ähnliche Toleranzen werden bei einer ganzen Klasse von Hilfsmethoden verwendet, die Linien miteinander vergleichen. Neben einer naheliegenden Implementierung der `equals(Object)`-Methode, die sich ggf. auf `equals(LineSegment)` abstützt und auch eine tolerantere Version `almostEquals(LineSegment)` besitzt, kann überprüft werden, ob Liniensegmente die gleiche Form haben (Gleichheit unter Vertauschung von Start- und Endpunkten durch `almostEqualShape(LineSegment)`) und schließlich ob die induzierten Linien via `isCoincident(LineSegment)` übereinstimmen.

Eine wichtige Operation zur komplexen Geometrie mit Linien und -segmenten ist die Berechnung von Schnittpunkten. Die für solche und ähnliche Berechnungen zugrundeliegende Methode ist `lineIntersection(LineSegment)`, die einen Punkt zurückliefert, oder eine Ausnahme auslöst, falls es keinen (eindeutigen) Schnittpunkt gibt. Die Ausnahme enthält dabei Informationen darüber, ob die induzierten Linien gleich (zu viele Schnittpunkte) oder parallel (keine Schnittpunkte) sind, oder ob der Schnittpunkt außerhalb der zugrundeliegenden Liniensegmente liegt. Die segmentsbereichsensitive letzte Ausnahme wird nur bei der darauf aufbauenden Methode `segmentIntersection(LineSegment)` ausgelöst.

Als Hilfsmethode bei der Berechnung von freien Zellenrandstücken wurde eine Methode `circleLineIntersection(Point, double)` implementiert, die die Schnittmenge der induzierten Linie mit einem durch Mittelpunkt und Radius definierten Kreis berechnet. Der Rückgabewert ist ein (möglicherweise leerer, durch `null` repräsentiert) Liniensegment. Durch Überprüfung der Parameter der Endpunkte dieses Liniensegments auf der induzierten Linie des ursprünglichen Liniensegments kann die Position der Schnitte festgestellt werden.

Darauf aufbauend wurde für die Ermittlung kritischer Werte für die lokalisierte Fréchet-Distanz entsprechend Lemma 3.9 eine Methode zur Ermittlung der Schnittmenge der induzierten Linie mit einem Apollonischen Kreis, gegeben durch zwei Punkte und ein Verhältnis, implementiert. Die genaue Berechnung folgt unmittelbar dem Beweis des genannten Lemmas.

Zur Berechnung von Ellipsen in Freespace-Zellen schließlich war es nötig, die zugehörigen zwei Liniensegmente so umzuparametrisieren, dass der Schnittpunkt der induzierten Linien vom Parameterpaar $(0, 0)$ repräsentiert wird (vgl. Abschnitt 2.2.3). Nachdem bereits die Schnittpunkte zweier Linien sowie die zugehörigen Parameter berechenbar sind, fehlte noch eine Methode, um eine Parameterverschiebung

zu realisieren. Mit Hilfe der Zugriffsmethoden `setParameterOffset(double)` und `getParameterOffset()` kann eine solche Parameterverschiebung realisiert werden.

Eine Repräsentation polygonaler Kurven

Polygonale Kurven schließlich als zentrales mathematisches Konstrukt in dieser Ausarbeitung werden innerhalb von Objekten der Klasse `PolygonalCurve` durch eine Liste von Punkten implementiert. Obwohl diese Festlegung auch eine Liste von Liniensegmenten induziert, werden Liniensegmente aus den zugrundeliegenden Punkten nur bei Bedarf erzeugt, da die Liniensegmente jeweils mit Start- und Endpunkten übereinstimmen müssten und so redundante Information enthielten, deren Korrektheit mit zusätzlichen Tests sichergestellt werden müsste.

Neben einfachen Zugriffsmethoden wie etwa `getPoints()` (die die Liste der zugrundeliegenden Punkte zurückliefert und Modifikationen daran zulässt) und der Berechnung der gesamten Euklidischen Länge `getLength()` durch Addition der Punktdistanzen besteht ein großer Teil des Codes der Klasse `PolygonalCurve` bereits aus speziellen Methoden zur Distanzmessung, wird also im nächsten Abschnitt genauer behandelt.

Bereits im Vorfeld dieser Arbeit wurde aber als Distanz polygonaler Kurven die Hausdorff-Distanz (vgl. Abschnitt 2.1) zweier `PolygonalCurve`-Objekte als eine symmetrische Version von `maxPointSegmentDistance(PolygonalCurve)`, die die Methode `distance(Point)` verwendet, implementiert. Letztere berechnet den Abstand des übergebenen Punktes von der zugrundeliegenden polygonalen Kurve als das Minimum der Punkt-Segment-Distanzen.

Beliebige Ellipsen als AWT-Shapes

Keine geometrische Grundlage für die Berechnung von Fréchet-Distanzen, sondern eine Grundlage zur Visualisierung der exakten Gestalt des Freespace in Diagrammzellen, ist die Klasse `RotatedEllipse`. Sie wird trotzdem hier vorgestellt, da sie keine ausschließlich für Fréchet-Distanz-Anwendungen benutzbare Klasse ist.

Die Java-Klassenbibliothek bietet zwar die für die Visualisierung geeignete Ellipsenklaasse `Ellipse2D.Double` im AWT-Paket, deren Objekte über einen oberen linken Punkt, eine Höhe und eine Breite konstruiert werden, jedoch sind die Halbachsen dieser Ellipsen stets horizontal und vertikal ausgerichtet. Erst durch die Anwendung von `AffineTransform`-Instanzen (Verschiebung, Skalierung, Scherung, Rotation) kann eine Ellipse beliebig verdreht werden. Genau die dazu nötigen Daten (gegeben durch Mittelpunkt, Breite und Höhe sowie einen Drehwinkel) werden in dieser Klasse gekapselt, denn nach Anwendung der Drehung ohne kapselnde `RotatedEllipse`-Klasse liegt das Objekt nur noch als `Shape`-Objekt vor, ohne Zugriff auf die bei der Konstruktion beteiligten Daten.

4.2.2 Entscheidungsproblem und lokalisierte Fréchet-Distanz

Zur konkreten Berechnung von Entscheidungsproblem und Fréchet-Distanz selbst ist das in dieser Ausarbeitung beschriebene Hilfsmittel der Wahl das Freespace-Diagramm. Es besteht aus je einer Zelle für jedes mögliche Paar von Liniensegmenten der beiden zu vergleichenden polygonalen Kurven. Die Implementierung ist eine exakte Repräsentation des mathematischen Modells.

Repräsentation von Freespace-Zellen

Alle Daten, die zur Berechnung des Freespace innerhalb einer Zelle `FreeSpaceCell` des Freespace-Diagramms benötigt werden, lassen sich aus den zwei zugehörigen Liniensegmenten $\varphi, \psi : I \rightarrow \mathbb{R}^2$ und einem maximalen punktweisen Abstand δ ableiten, daher sind dies auch die einzigen bei der Konstruktion zu übergebenden Daten.

Die wesentliche definierende Funktion für den Freespace ist die Distanzfunktion für die beiden durch ein Parameterpaar der Zelle induzierten Punkte auf den zugehörigen Liniensegmenten `getPointPairDistance(double, double)`. Es wird an dieser Stelle keine Ausnahme ausgelöst, wenn das Parameterpaar nicht aus dem Inneren der Zelle $I \times I$ stammt, da die beschriebene Distanzfunktion problemlos auf dem Parameterraum der induzierten Linien $\mathbb{R} \times \mathbb{R}$ fortgesetzt werden kann.

Die Entscheidungsmethode `isInFreeSpace(double, double)`, die überprüft, ob ein übergebenes Parameterpaar im Freespace liegt, wertet die oben beschriebene Methode lediglich an dieser Stelle aus und vergleicht die resultierende Distanz mit der bei der Konstruktion angegebenen maximalen punktweisen Distanz δ . Um jedoch den gesamten Freespace der zugrundeliegenden Zelle effizient zu erfassen, ist eine Brute-Force-Auswertung „aller“ Parameterpaare der Zelle nicht geeignet. Hierzu werden zwei Lösungsansätze parallel verfolgt:

- Für die exakte Berechnung der Fréchet-Distanz bzw. zur Lösung des Entscheidungsproblems ist die Gestalt des Freespace im Inneren der Zellen nicht relevant. Es genügt zunächst die Berechnung der freien Zellenrandstücke, deren Parameter sich durch Schnitte eines Kreises vom Radius δ um den die Kante repräsentierenden Punkt mit dem anderen Liniensegment errechnen lassen.
- Zur Visualisierung des Freespace-Diagramms wird der gesamte Freespace der Zelle als Shape berechnet. Dies wird im kommenden Abschnitt näher vorgestellt.

Berechnung des Freespace als Shape

Entsprechend der Argumentationen zu Schläuchen und Ellipsen in Kapitel 2 wird die `Shape`-Berechnung des Freespace einer Zelle von der Lage der induzierten Linien durch die zugehörigen Liniensegmente abhängig gemacht. Gibt es einen Schnittpunkt dieser Linien, so hat der Freespace die Form einer Ellipse, sind die Linien parallel, so handelt es sich um einen Schlauch. Der Rückgabetyp der Methode `getFreeSpaceShape()` ist `FreeSpaceCellShape`. In dieser abstrakten Basisklasse werden einige Methoden gebündelt, die bei der Verwendung solcher Shapes in der Visualisierung benötigt werden, etwa die Berechnung eines `Area`-Objekts, das man durch den Schnitt mit der Zelle $I \times I$ erhält.

Die Klassen `FreeSpaceCellEllipse` und `FreeSpaceCellTube`, die von der `Shape`-Klasse erben, berechnen die Gestalt des Freespace schließlich auf Basis der Berechnungsvorschriften in den Abschnitten 2.2.3 und 2.2.2, das wichtigste Resultat dieser Berechnungen ist schließlich ein in der Visualisierung allgemein verwendbares `Shape`-Objekt.

Das Freespace-Diagramm als zentrales Berechnungswerkzeug

Sind zwei polygonale Kurven gegeben, so ist entsprechend der Ausführungen in dieser Ausarbeitung die Erzeugung des dazugehörigen Freespace-Diagramms das

zentrale Hilfsmittel. Dabei wird im Standardfall eine konstante maximale punktweise Distanz δ und im lokalisierten Fall eine Distanzlokalisierung dl für die ausgezeichnete Referenzkurve übergeben. Es wurde bereits gezeigt, dass die Standard-Fréchet-Distanz von der lokalisierten Fréchet-Distanz als Spezialfall umfasst wird, wenn man die Distanzlokalisierung konstant wählt. Das ist auch der Grund, warum intern in der Klasse `FreeSpaceDiagram` immer mit einer Distanzlokalisierung gearbeitet wird. Ein Konstruktor, der mit einem `double`-Argument für die maximale punktweise Distanz den Standardfall repräsentiert, erzeugt zunächst eine konstante Distanzlokalisierung mit eben diesem Abstand als einzigm Wert. Ferner wird bei der Konstruktion eines solchen Diagramms für jedes Paar von Liniensegmenten das dazugehörige `FreeSpaceCell`-Objekt erzeugt und mit der passenden ggf. lokalisierten maximalen punktweisen Distanz versehen.

Sowohl bei der Lösung des Entscheidungsproblems als auch bei der konkreten Berechnung einer Fréchet-Distanz steht der erreichbare Raum im Vordergrund, denn die Frage ist ja stets, ob zu einer gegebenen Distanzlokalisierung bzw. zu einem gegebenen maximalen punktweisen Abstand die rechte obere Ecke des Diagramms erreichbar ist. Diese Berechnung wird iterativ entsprechend der Berechnungsvorschrift aus Lemma 3.9 durchgeführt und jeweils zwischengespeichert. Das Ergebnis ist eine Methode `isReachable(int, int)`, die zu einem gegebenen Punkt auf den Eckpunkten der Freespace-Zellen positiv oder negativ beantwortet, ob dieser Punkt erreichbar ist. Die Lösung des Entscheidungsproblems ist dann schließlich mit `isPossible()` nur noch das Einsetzen des rechten oberen Punktes in diese Methode.

Durch die bei den Konstruktion fest vorgegebene Distanz oder Distanzlokalisierung für ein Freespace-Diagramm kann ein `FreeSpaceDiagram`-Objekt jedoch nicht bei der Suche nach kritischen Werten zur konkreten Distanzberechnung behilflich sein, was beim Entwurf aber auch nicht erforderlich erschien. Schließlich besteht die Hauptaufgabe von `FreeSpaceDiagram`-Objekten darin, den erreichbaren Raum zu berechnen, was bei jeder Änderung der Distanzlokalisierung wiederholt werden muss.

Berechnung kritischer Werte und der lokalisierten Fréchet-Distanz

Die kritischen Werte werden im Kontext von `PolygonalCurve`-Objekten in der Methode `criticalFreespaceDiagramValues()` berechnet, die optional mit einer Distanzlokalisierung eine Auskunft über die lokalen Verhältnisse der spaltenweisen maximalen punktweisen Abstände erhält. In einer Version ohne Distanzlokalisierung wird implizit mit einer konstanten Distanzlokalisierung vom Wert 1 gearbeitet, was genau dem Standardfall entspricht.

Entsprechend Lemma 3.9 werden alle möglichen kritischen Werte der drei Typen berechnet und in einem `CriticalValueMap`-Objekt gespeichert. Diese Klasse ist eine Spezialisierung einer `TreeMap` mit `Double`-Schlüsseln und `Set<CriticalValue>`-Werten, wodurch zu einem kritischen Wert gegeben durch den Distanzlokalisierungsfaktor eine Menge von kritischen Events repräsentiert werden kann. Diese `CriticalValue`-Objekte kapseln Informationen über ihren Typ und die Spalten, die durch das Auftreten dieses kritischen Werts auf den entsprechenden Zahlenwert fixiert werden müssen. Für die Parametersuche bei der Berechnung einer lokalisierten Fréchet-Distanz macht das keinen Unterschied zu der laut [AG92] zu verwendenden Liste. Will man jedoch mit Hilfe der Fréchet-Distanz eine Distanzlokalisierung berechnen wie in Abschnitt 3.3.2 beschrieben, benötigt man nach der erfolgten Suche nach dem kritischen Wert, der die Fréchet-Distanz zweier Kurven realisiert, die hier beschriebenen Meta-Daten.

Mit Hilfe einer binären Suche schließlich wird innerhalb einer sortierten Liste der resultierenden kritischen Zahlenwerte nach dem kleinsten Wert gesucht, für den das zugehörige (lokalierte) Fréchet-Distanz-Entscheidungsproblem noch positiv gelöst wird. Dies entspricht auch genau dem diskutierten Verfahren zur Berechnung der Fréchet-Distanz. Aufrufbar ist diese Berechnung schließlich mittels einer Methode `frechetDistance(PolygonalCurve)`, die optional eine Distanzlokalisierung als zusätzliches Argument erhalten kann.

4.2.3 Eine grafische Benutzerschnittstelle

Zur optischen Verifikation der Korrektheit mancher Berechnungen ist eine einfache interaktive grafische Benutzerschnittstelle implementiert worden. Ihr Bedienfeld ist in zwei Hälften aufgeteilt, einen Datenraum und einen Parameterraum. Im Datenraum kann der Benutzer zwei polygonale Kurven mit der Maus modifizieren, es können Punkte hinzugefügt (einfacher Klick) oder entfernt (Doppelklick) und verschoben werden. Die Aktivierung einer `snap`-Funktion erlaubt bei den Punktkoordinaten im Anzeigebereich ($[0, 10] \times [0, 10]$) nur ganzzahlige Werte, so dass die Punkte in einem Gitter „einrasten“. Der über einen Schieber einstellbare maximale punktweise Abstand wird über einen Kreis um den Mauszeiger visualisiert.

Im Parameterbereich wird das zugehörige Freespace-Diagramm der beiden Kurven in Echtzeit gezeichnet und aktuell gehalten. Die Gitterfarbe für die Freespace-Zellen ist dabei ein farblicher Indikator für die Lösbarkeit des zugehörigen Fréchet-Distanz-Entscheidungsproblems. Schwebt die Maus über dem Freespace-Diagramm, so werden die Koordinaten der Maus in Echtzeit in ein Parameterpaar für die beiden Liniensegmente umgerechnet und im Datenbereich eingezeichnet. Eine Verbindungsline zwischen den beiden resultierenden Punkten auf den polygonalen Kurven indiziert über ihre Farbe, ob der Abstand kleiner oder gleich dem gewählten maximalen punktweisen Abstand ist, ob das Parameterpaar also frei war.

Es ist möglich, eine Distanzlokalisierung beim Aufruf mit anzugeben, diese kann jedoch nicht zur Laufzeit modifiziert werden. In dem Fall übernimmt der Schieber für den maximalen punktweisen Abstand die Rolle der Skalierung für die Distanzlokalisierung. Schließlich gibt es einen Knopf, der eine Text-Ausgabe der aktuellen erreichbaren Zellenrandstücke auslöst.

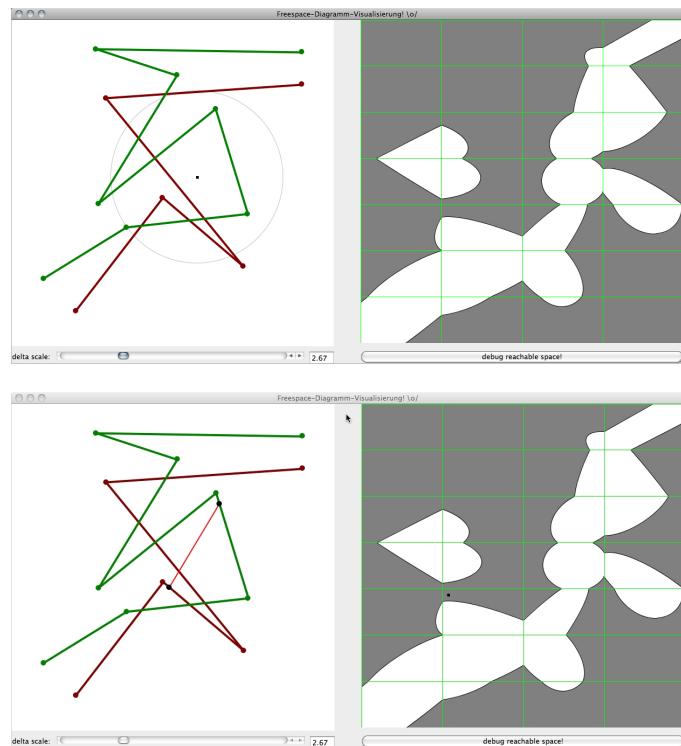


Abbildung 4.3: Das Visualisierungsprogramm. Im linken Bereich kann man mit der Maus interaktiv die polygonalen Kurven modifizieren. Der Mauszeiger wird dabei von einer Visualisierung des unten einstellbaren maximalen punktweisen Abstands umgeben (oben). Im rechten Bereich wird in Echtzeit das zugehörige Freespace-Diagramm angezeigt. Schwebt man mit der Maus über diesem Diagramm (unten), wird zu dem Parameterpaar „unter“ der Mausposition das Punktpaar auf den polygonalen Kurven errechnet und im linken Bereich eingezeichnet (rot: Parameterpaar „verboten“).

Kapitel 5

Reflexion und Ausblick

Die in dieser Diplomarbeit vorgestellten geometrischen Methoden zu verstehen und die Darstellung auszuarbeiten war äußerst interessant und lehrreich. Die zur genauen Gestalt des Freespace gemachten Aussagen sinnvoll aufeinander aufzubauen und zu beweisen war ebenso wie die eigenständige Definition des Begriffs der lokalisierten Fréchet-Distanz eine für mich neuartige mathematische Herausforderung.

Durch die Anwendung in der astronomischen Datenanalyse gemeinsam mit den bestehenden Anforderungen an die Zusammenarbeit der einzelnen Komponenten war auch der praktische Teil dieser Diplomarbeit nicht auf das beinahe automatisierbare Implementieren von bereits dokumentierten Algorithmen beschränkt, sondern gestaltete sich auch vom Standpunkt des Softwareentwurfs her als herausfordernde Aufgabe. Wie in jedem Softwareprojekt ist die Arbeit an der Implementierung von Fréchet-Distanz-Varianten in Clastro aber keineswegs abgeschlossen. Der Prozess der generischen Berechnung guter Distanzlokalisierungen zu bestehenden Trainingsmengen bietet sicherlich weiteres Optimierungspotential. Ebenso wäre die Frage von Interesse, ob die vorgestellten Methoden auf neue astronomische Klassifikationsaufgaben so flexibel anpassbar sind wie erhofft.

Auch im Bereich der algorithmischen Geometrie wurden Fragen aufgeworfen, die im Rahmen dieser Diplomarbeit nicht beantwortet werden konnten. Schon in Abschnitt 2.3 wurde angemerkt, dass es eine nichttriviale Aufgabe sein könnte, für zwei polygonale Kurven zu einem vorgegebenen maximalen Anteil der Weglänge im verbotenen Raum des Freespace-Diagramms Parametrisierungen zu finden, die eine minimale partielle Fréchet-Distanz realisieren. Auch eine Kombination der partiellen Variante mit der lokalisierten Fréchet-Distanz erscheint mir interessant.

Abstrakt definierte geometrische Methoden wie die Fréchet-Distanz in einem realen Forschungszusammenhang wie der Astrophysik zur Anwendung zu bringen und damit nichttriviale Probleme zu lösen, war jedenfalls eine spannende Tätigkeit und bot vielfältige Möglichkeiten, Kenntnisse aus verschiedenen Teilgebieten von Mathematik und Informatik gewinnbringend einzusetzen.

Literaturverzeichnis

- [AG92] Helmut Alt and Michael Godau. Computing the Fréchet Distance between two polygonal Curves, 1992.
- [BBMS] Kevin Buchin, Maike Buchin, Wouter Meulemans, and Bettina Speckmann. Locally Correct Fréchet Matchings.
- [BBW] Kevin Buchin, Maike Buchin, and Yusu Wang. Exact Algorithms for Partial Curve Matching via the Fréchet Distance.
- [BHW] Asa Ben-Hur and Jason Weston. A User's Guide to Support Vector Machines. pyml.sourceforge.net/doc/howto.pdf.
- [dCGM⁺13] Jean-Lou de Carufel, Amin Gheibi, Anil Maheshwari, Jörg-Rüdiger Sack, and Christian Scheffer. Similarity of polygonal curves in the presence of outliers, 2013.
- [DHP] Anne Driemel and Sariel Har-Peled. Jaywalking your Dog: Computing the Fréchet Distance with Shortcuts.
- [For06] Otto Forster. *Analysis 2. Differentialrechnung im \mathbb{R}^n , gewöhnliche Differentialgleichungen*. Vieweg-Verlag, 2006.
- [GPT⁺10] Fabian Gieseke, Kai Lars Polsterer, Andreas Thom, Peter Zinn, Dominik Bomans, Ralf-Jürgen Dettmar, Oliver Kramer, and Jan Vahrenhold. Detecting Quasars in Large-Scale Astronomical Surveys, 2010.
- [Hen] Jeff Henrikson. Completeness and Total Boundedness of the Hausdorff Metric.
- [HTF11] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer, 2011.
- [OCSW12a] Karen Elizabeth Andaviza Oblitas, Victor Cuadrado, Stephan Schmedding, and Mirko Westermeier. Clastro - Klassifikation astronomischer Objekte. Projektseminarpräsentation an der Ruhr-Universität Bochum, 2012.
- [OCSW12b] Karen Elizabeth Andaviza Oblitas, Victor Cuadrado, Stephan Schmedding, and Mirko Westermeier. Data Mining in Astronomy - Project Report, 2012.
- [Wal13] Richard Walker. Analyse und Interpretation astronomischer Spektren, 2013.

Eidesstattliche Versicherung

Hiermit erkläre ich, Mirko Westermeier, an Eides statt, dass ich die vorliegende Diplomarbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Münster, 23. Juni 2014
