

Julius-Maximilians-Universität Würzburg
Institut für Informatik
Lehrstuhl für Informatik I
Effiziente Algorithmen und wissensbasierte Systeme

Bachelorarbeit

Zeichnen von Netzwerken unter Berücksichtigung von Kantenlängen

Matthias Neumann

Eingereicht am 21.01.2015

Betreuer:
Prof. Dr. Alexander Wolff
Fabian Lipp, M. Sc.

Inhaltsverzeichnis

| | |
|--|-----------|
| 1. Einleitung | 3 |
| 2. Definitionen und mathematische Grundlagen | 8 |
| 2.1. Ungerichtete Graphen | 8 |
| 2.2. Beziérkurven | 9 |
| 2.3. Längenberechnung von Kurven | 10 |
| 3. Erstellen von Streckenplänen | 12 |
| 3.1. Anfertigung einer geradlinigen Zeichnung | 12 |
| 3.2. Anfertigung einer Zeichnung mit Beziérkurven als Kanten | 13 |
| 3.2.1. Vorstellung des Zeichenalgorithmus | 14 |
| 3.2.2. Erstellen des Ausgangsgraphen | 14 |
| 3.2.3. Kräfte und deren Berechnung | 16 |
| 3.2.4. Zeichnung und Vereinfachung von Kanten | 20 |
| 4. Ergebnisse des Algorithmus | 22 |
| 4.1. Entstandene Zeichnungen | 22 |
| 4.2. Abweichungen von der gewünschten Kantenlänge | 27 |
| 4.3. Einfluss des Vorfaktors auf im Graphen wirkende Kräfte | 28 |
| 5. Fazit und Ausblick | 30 |
| A. Inhalt der beigelegten CD | 35 |
| B. Erklärung | 36 |

1. Einleitung

Im ersten Halbjahr 2014 fuhren etwa 3,24 Milliarden Personen in Deutschland mit Eisen- und Straßenbahnen. Dies entspricht einem Durchschnitt von ungefähr 18 Millionen Fahrgästen pro Tag im Nah- und Fernverkehr [Bun14]. Hierbei kann es durchaus vorkommen, dass Passagiere unbeabsichtigt in den falschen Zug steigen. Auch bei der Deutschen Bahn verpassen ICEs ihre Haltestellen oder fahren in Richtung Hannover statt nach Berlin [Bli13].

Um solche Irrfahrten zu vermeiden hängen, zumindest zur Orientierung der Reisenden, in Zügen und an Haltestellen schematische Karten der Bahnverbindungen aus. Solche Pläne sind oft *oktilinear* gezeichnet, einzelne Stationen sind über Strecken verbunden, die horizontal, vertikal oder diagonal, im Winkel von 45°, verlaufen. *Topographisch* sind diese schematischen Karten meist nicht korrekt, die dargestellten Haltestellen und deren Entfernung voneinander entsprechen nicht ihrer tatsächlichen geographischen Lage. Stattdessen sind aus Gründen der Übersichtlichkeit Stationen in gleichmäßigen Abständen über die Strecke verteilt, siehe Abbildung 1.1. Dadurch entzerrt sich die Darstellung in Bereichen mit vielen Haltestellen auf engem Raum, zum Beispiel in Innenstädten oder Ballungsräumen.



Abb. 1.1.: Liniennetzplan der Würzburger Straßenbahn [Sau10].

Vorangegangene Arbeiten. Mit der Erstellung von ansprechenden, schematischen Karten beschäftigen sich Nöllenburg und Wolff [NW11]. Hier wird die übersichtliche Auszeichnung von Haltestellen, ohne deren Überlappungen und der automatische Entwurf eingängiger Fahrpläne mit Hilfe linearer Optimierung thematisiert. Die oktilinearen Pläne können nach unterschiedlichen Kriterien, wie Minimierung der Anzahl der Knicke oder Treue zu den tatsächlichen Koordinaten erstellt werden.

Bei Fink et al. [FHN⁺13] wird die automatische Zeichnung von gut zu überblickenden Liniennetzplänen mit *Beziérkurven* erläutert, ohne dabei speziell auf die Visualisierung der Fahrtzeiten einzugehen. Um möglichst elegante Kurven zeichnen zu können, wird versucht möglichst viele Verbindungen zwischen den Haltestellen in einer Beziérkurve zusammenzufassen. Auf den dazu verwendeten *kräftebasierten* Algorithmus wird in Kapitel 3.2 noch genauer eingegangen.

Um den vielen Fahrgästen im öffentlichen Personenverkehr zusätzlich zu einer übersichtlichen Karte auch die Fahrtzeiten bestimmter Verbindungen zu veranschaulichen gibt es verschiedene Methoden. Eine Möglichkeit sind *isochrone* Karten, bei welchen, von einem Punkt ausgehend, Linien gleicher Reisezeit eingezeichnet sind, siehe Abbildung 1.2. Somit ist von der Station, von welcher die isochronen Linien ausgehen, gut ersichtlich welche Orte zur selben Zeit erreicht werden können. Sollen aber die Fahrtzeiten zwischen zwei beliebigen Haltestellen dargestellt werden, ist diese Art Karte ungeeignet.



Abb. 1.2.: Eine frühe Isochronenkarte von 1881 [Gal81]. Ausgehend von London wird die Reisezeit in verschiedenen Regionen der Welt in Tagen dargestellt.

Andere Methoden zur Veranschaulichung der Fahrtzeiten, ausgehend von beliebigen Orten als Startpunkt, werden von Haverkort [Hav14] vorgestellt. Eine naheliegende Vorgehensweise ist die numerische Darstellung der Reisezeit an den Verbindungen zwischen zwei Halten. Dies hat den Vorteil, dass schematische Karten nicht verändert werden müssen, es muss lediglich die Fahrtzeit an den einzelnen Streckenabschnitten hinzugefügt werden.

Weitere, auch von Haverkort beschriebene Möglichkeiten, Hinweise auf die Reisezeit zu geben, sind über die Verbindungslinien an sich. Beispielsweise wird die Reisezeit anhand der Stärke der Linie dargestellt, je kräftiger eine Verbindung gezeichnet ist, desto länger die Fahrtzeit. Ebenfalls möglich sind Hinweise über die Form der Verbindungslinien, je mehr sich ein Streckenabschnitt schlängelt, umso zeitintensiver ist die Fahrt. Auch über das Hinzufügen bestimmter Marker an einer Verbindung, wie einer Zacke, die für eine Verlängerung der Fahrt um einen bestimmten Betrag steht, lassen sich Informationen über die Reisezeit angeben.

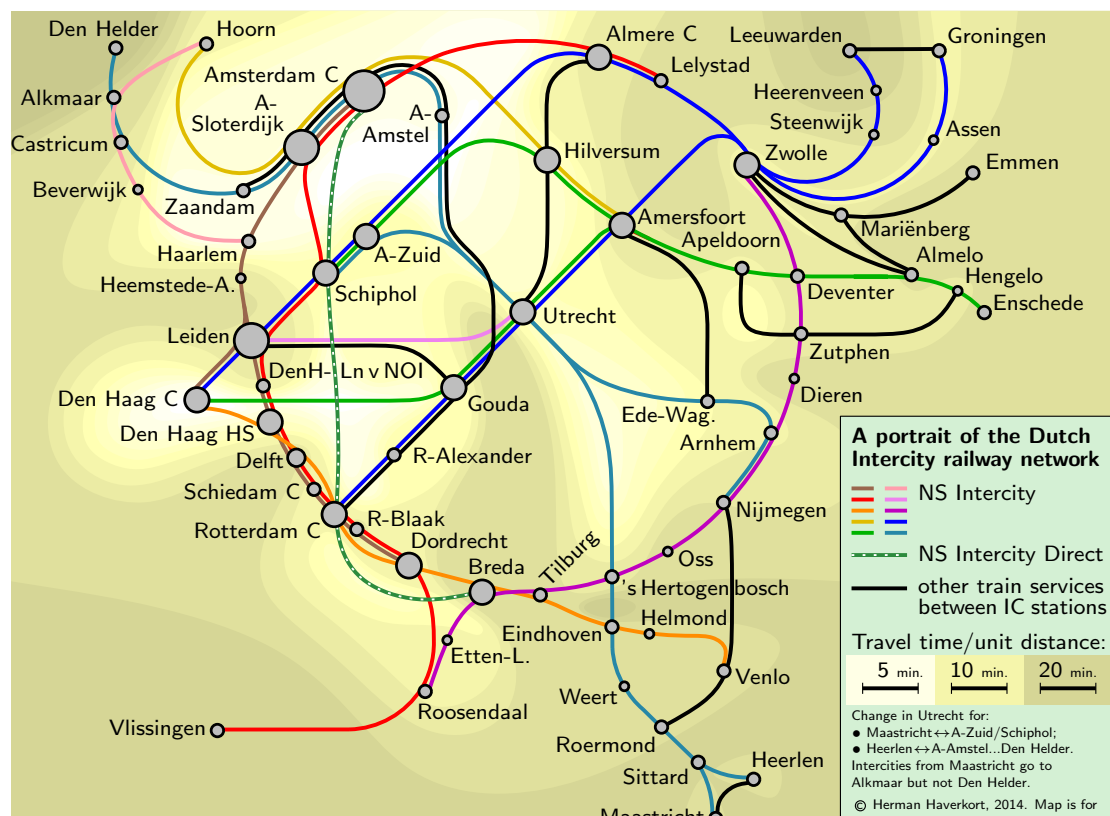


Abb. 1.3.: Wärmebild des niederländischen Intercity Liniennetzes [Hav14].

Das Einfärben des Kartenhintergrunds kann ebenfalls Rückschlüsse auf die Fahrtzeiten zulassen. Ist der Hintergrund zwischen zwei Stationen hell, so ist die Fahrtzeit zwischen ihnen gering. Wird der Farbton dunkler, bedeutet dies, dass sich die Fahrtzeit verlängert.

Ähnlich eines *Wärmebildes*, bei welchem Zonen hoher Temperatur durch warme Farben und kalte Gebiete durch kalte Farben dargestellt werden. In Abbildung 1.3 ist ein Wärmebild des niederländischen Intercity Liniennetzes zu sehen. Neben der Farbgebung einzelner Zonen fließt hier auch die Länge der Verbindung zwischen zwei Stationen in die Darstellung der Reisezeit mit ein.

Des Weiteren lässt sich eine Karte, wie ebenfalls bei Haverkort beschrieben, in einzelne Gebiete aufteilen. Liegen Stationen in der selben Zone, so sind sie maximal eine bestimmte Zeitspanne voneinander entfernt. Sind zwei Halte in unterschiedlichen Gebieten, so erhöht sich für jede Zone zwischen ihnen auch die Fahrtzeit. Solche Gebiete können unterschiedlich geformt sein, etwa wie Kreise oder Polygone, oder durch ein Gitternetz aufgespannt werden.

Stevens [Ste12] diskutiert die Darstellung der Reisezeit durch die unterschiedliche Farbgebung von Streckenabschnitten, genutzt zur Illustration der momentanen Verkehrssituation auf Straßen. Diese Art der Kartengestaltung richtet sich vor allem an Kraftfahrer mit internetfähigen Navigationssystemen, um ständig die aktuelle Verkehrslage überwachen zu können. Wie bei einer Ampel veranschaulichen die Farben rot, gelb und grün das aktuelle Verkehrsaufkommen auf einer bestimmten Strecke, grün steht hierbei für wenig Verkehr, rot für viel. Dieser Ansatz wird von Stevens um die sogenannte „*fixed-minute*“-Methode erweitert. Bei dieser werden Marker auf Straßen gesetzt, welche diese in Etappen von jeweils einer Minute Fahrtzeit unterteilt. Durch Aufsummieren der einzelnen Abschnitte kann die Reisezeit ermittelt werden. Die sichtbaren Auswirkungen dieser Marker auf einer Karte sind allerdings gering, da diese lediglich auf die bereits vorhandenen Routen gesetzt werden. Somit ist es für die Nutzer nicht auf den ersten Blick ersichtlich welche Route die schnellere ist. Hierzu müsste erst die Anzahl der Marker auf den zu vergleichenden Strecken gezählt werden.

Haunert und Sering [HS11] beschäftigen sich ebenfalls mit der Erstellung von Straßenkarten. Hier wird ein bestimmter Kartenabschnitt als Schwerpunkt betrachtet. Dieser wird um einen bestimmten Faktor vergrößert, während das Format der Karte gleich bleibt. Erreicht wird dies, indem Regionen außerhalb des Fokus nach unten skaliert werden, während die Verzerrung möglichst gering gehalten wird. Dieser Ansatz wird in Kapitel 5 noch einmal aufgegriffen. Er beschäftigt sich zwar nicht mit der Darstellung von Fahrtzeiten bestimmter Strecken, kann aber dazu genutzt werden, Zeichnungen von Liniennetzplänen ansprechender Darzustellen.

Bei Buchin et al. [BvGH⁺14] wird die Fahrtzeit in der Weise visualisiert, dass sich die Länge einer Verbindung proportional zur tatsächlichen Reisezeit verhält. Da die Knotenpunkte hier fest bleiben, mäandern die Verbindungslinien stark um auf eine gewünschte Länge zu kommen.

Auch Karten, die nicht primär darauf ausgelegt sind Hinweise auf die Reisezeiten zu liefern, können Anhaltspunkte über diese geben. So spiegeln maßstabsgetreue Karten die tatsächlichen Entfernungen wider und liefern Informationen über den Verlauf einer Fahrtstrecke, wobei hier die Reisegeschwindigkeit nicht berücksichtigt wird. Dennoch kann man sofort erkennen, ob eine Strecke kurvig oder geradlinig verläuft und Rückschlüsse auf deren Fahrtzeit ziehen.

Straßenkarten in denen die unterschiedlichen Kategorien der Fahrbahnen gekennzeichnet sind, sagen ebenfalls etwas über die Reisezeiten aus. Denn auf einer Autobahn kommt man in der Regel schneller voran als auf einem Feldweg. Folgerungen auf die Begrenzung der Geschwindigkeit können allerdings nicht gezogen werden.

Im öffentlichen Personennahverkehr gibt es zudem oft Karten in denen die Tarifzonen eingezeichnet sind. Über diese lässt sich eine erste Abschätzung der Reisezeit geben, da die Anzahl an durchquerten Zonen mit der Reisezeit korreliert. Allerdings sagen die einzelnen Zonen meist mehr über die räumliche Ausdehnung als über die tatsächlichen Fahrtzeiten aus.

Gliederung der Arbeit. Im Folgenden wird darauf eingegangen, Fahrtzeiten in Liniennetzplänen über die Kantenlängen zwischen zwei Haltestellen zu visualisieren. Dabei wird vom Prinzip der oktilinearen Karten Abstand genommen. Stattdessen werden Zeichnungen von Bahnnetzen mit Beziérkurven angefertigt. Grundlage hierfür ist der kräftebasierte Algorithmus von Fink et al. [FHN⁺13]. Dieser wird um eine Kraft erweitert, welche versucht Knoten in Abstände zu setzen, die proportional zur Fahrtzeit zwischen ihnen sind.

Zu Beziérkurven und ihren Eigenschaften wird in Kapitel 2 eine kurze Einführung gegeben. Zudem wird der Zusammenhang zwischen ungerichteten Graphen und Liniennetzplänen erläutert. Zuletzt wird ein Algorithmus vorgestellt, welcher die Länge einer Beziérkurve approximiert.

In Kapitel 3 wird die Vorgehensweise des Algorithmus von Fink et al. erklärt. Am Beispiel des Würzburg Straßenbahnnetzes wird gezeigt, wie man eine geradlinige Zeichnung als Eingabe für den Algorithmus erhält. Es wird weiter dargelegt, welche Vorverarbeitungsschritte für den Eingabegraphen nötig sind, um die anschließend skizzierten verschiedenen Kräfte auf den Graphen wirken zu lassen. Zum Abschluss des Kapitels wird ausgeführt, welche weiteren Vereinfachungen am Graphen vorgenommen werden und wie die Stationen später in der Zeichnung gesetzt werden.

Kapitel 4 stellt die Ergebnisse des zuvor eingeführten Algorithmus vor. Dabei handelt es sich um Zeichnungen der U-Bahn-Netze aus Wien und Montreal, der Vorortslinien von Sydney, sowie des Würzburger Straßenbahnnetzes. Die Liniennetzpläne werden auf die Abweichungen der Kantenlängen von ihren angestrebten Werten hin untersucht und bewertet.

Am Ende der Arbeit wird in Kapitel 5 ein Fazit gezogen und ein Ausblick darauf gegeben, welche weiteren Möglichkeiten bestehen, um das Erstellen von Liniennetzplänen zu optimieren.

2. Definitionen und mathematische Grundlagen

Im folgenden Kapitel 2 wird zuerst ein ungerichteter Graph definiert und erläutert weshalb Graphen geeignet sind Liniennetzpläne darzustellen. Weiter wird eine Einführung zu Beziérkurven gegeben. Es wird erklärt wie diese sich zeichnen lassen, wie man sie gewichtet und wie ihre Tangenten verlaufen. Abschließend wird darauf eingegangen wie die Länge von Kurven berechnet wird und es wird ein Algorithmus vorgestellt, der die Länge einer Beziérkurve näherungsweise bestimmt.

2.1. Ungerichtete Graphen

Eine Möglichkeit der mathematischen Interpretation von Streckenfahrplänen ist deren Auffassung als *ungerichteter Graph*, $G = (V, E)$, mit einer endlichen *Knotenmenge* $V \neq \emptyset$ und der *Kantenmenge* $E \subseteq \{\{u, v\} \mid u, v \in V, u \neq v\}$. Die Knoten V stehen hier für einzelne Haltestellen und die Kanten E für Streckenabschnitte zwischen den Stationen. Die Kanten zeigen bei einem ungerichteten Graphen in keine Richtung, wie Züge die zwischen zwei Stationen in beliebiger Richtung verkehren können.

Sind in einem Graphen parallele Kanten zugelassen, $e_1, e_2 \in E$, mit $e_1 = \{u, v\} = e_2$, so spricht man von einem *Multigraphen*. Im Liniennetzplan aus Abbildung 1.1 sind solche parallelen Kanten zu sehen, wenn zwei oder mehr der unterschiedlich eingefärbten Linien zwischen den gleichen zwei Haltestellen verkehren.

Wird nachfolgend der Grad eines Knotens u , mit der Notation $\deg(u)$ angeführt, so ist damit die Anzahl i an benachbarten Knoten v_1, \dots, v_i gemeint, die durch eine Kante mit u verbunden sind. Der Grad eines Knotens hängt also nur von der Anzahl benachbarter Knoten ab, nicht von der Anzahl an ein- und ausgehenden Kanten, siehe Abbildung 2.1.

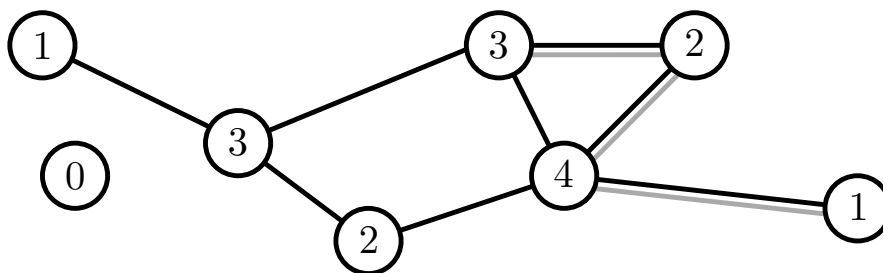


Abb. 2.1.: Ein ungerichteter Multigraph. Die Knoten sind mit ihrem Grad beschriftet.

2.2. Beziérkurven

Die einzelnen Knoten eines Graphen müssen in einer Zeichnung nicht durch geradlinige Strecken miteinander verbunden sein. Eine andere Option der Kantendarstellung sind Beziérkurven. Diese werden ausführlich bei Bartels et al. [BBB87] beschrieben. Eine Beziérkurve vom Grad n wird durch $n + 1$ *Beziéerpunkte* P_0, P_1, \dots, P_n bestimmt. Diese ziehen, über Strecken verbunden, das *Kontrollpolygon* einer Kurve auf. Formel 2.1 beschreibt eine Kurve vom Grad n für $t \in [0, 1]$.

$$C(t) = \sum_{i=0}^n B_{i,n}(t)P_i \quad \text{mit} \quad B_{i,n}(t) = \binom{n}{i} t^i (1-t)^{n-i} \quad (2.1)$$

Zudem kann man die einzelnen Punkte unterschiedlich stark *gewichten*, siehe Formel 2.2. Für Gewichte w größer eins wird die Beziérkurve stärker von einem Punkt angezogen und der Kurvenverlauf zu diesem Punkt hin beeinflusst. Bei Gewichten kleiner eins wird die Kurve von diesem Punkt abgestoßen.

$$C(t) = \frac{\sum_{i=0}^n B_{i,n}(t)P_i w_i}{\sum_{i=0}^n B_{i,n}(t)w_i} \quad (2.2)$$

Anschaulich erhält man für eine quadratische Beziérkurve vom Grad $n = 2$, mit den drei Beziéerpunkten P_0, P_1 und P_2 , aus Formel 2.1 die Gleichung 2.3. Mit Hilfe dieser Gleichung kann man jeden Punkt der Kurve bestimmen.

$$\begin{aligned} C(t) &= \sum_{i=0}^2 \binom{2}{i} t^i (1-t)^{2-i} P_i \\ &= (P_0 - 2P_1 + P_2)t^2 + (-2P_0 + 2P_1)t + P_0, \quad t \in [0, 1] \end{aligned} \quad (2.3)$$

In Abbildung 2.2 sind beispielhaft einige Kurven zu sehen. Links eine quadratische, nach Formel 2.3, in der Mitte eine kubische Beziérkurve, mit ihren jeweiligen Beziéerpunkten P_i und Q_i , sowie ihrem Kontrollpolygon (gestrichelt). Rechts sind zwei unterschiedlich stark gewichtete Beziérkurven dargestellt. Die Punkte R_1 und R_2 besitzen im Fall der schwarzen Kurve Gewichte gleich eins, im Fall der roten Kurve haben die Punkte Gewichte größer eins.

Sollen mehrere Beziérkurven möglichst glatt über einen ihrer Endpunkte miteinander verknüpft werden, so kann man dies über ein Angleichen ihrer *Tangenten* in diesem Punkt erreichen. Die Tangenten in den Endpunkten einer Kurve verlaufen entlang der Kanten $\overline{P_0 P_1}$ und $\overline{P_{n-1} P_n}$ des Kontrollpolygons. Aus Formel 2.4 kann man die Tangente jedes Punktes einer Beziérkurve bestimmen. Diese entspricht der Ableitung in diesem Punkt.

$$C'(t) = n \sum_{i=0}^{n-1} B_{i,n-1}(t)(P_{i+1} - P_i) \quad (2.4)$$

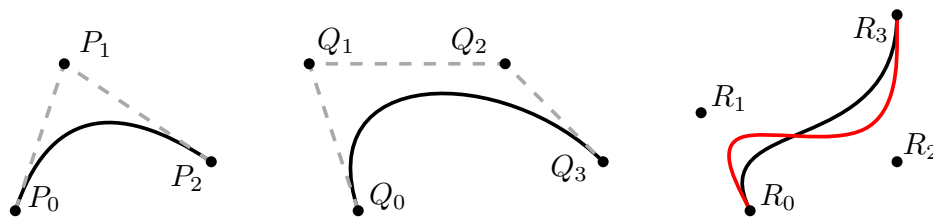


Abb. 2.2.: Links eine quadratische, in der Mitte eine kubische Beziérkurve. Rechts zwei Kurven mit unterschiedlicher Gewichtung der Punkte.

In Abbildung 2.3 ist ein Beispiel für zwei miteinander verknüpfte Kurven dargestellt. Der letzte Beziérpunkt der linken Kurve, P_3 , entspricht dem ersten Punkt der rechten Kurve, Q_0 . Die Tangenten der beiden Punkte gehen durch P_2 , $P_3 = Q_0$ und Q_1 und sind somit gleich. Die beiden Kurven erscheinen somit wie eine durchgehende Beziérkurve.

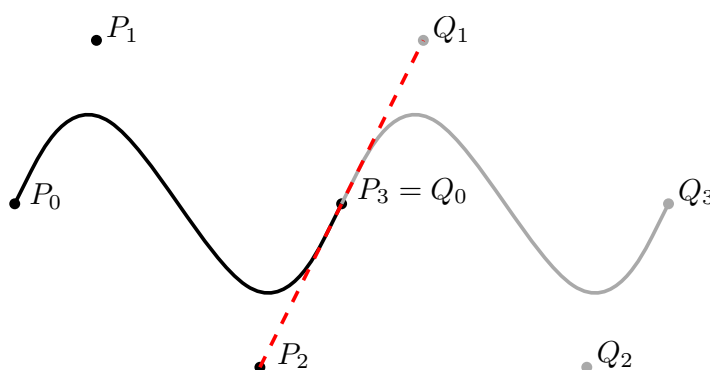


Abb. 2.3.: Ineinander übergehende Beziérkurven (grau und schwarz) mit der gleichen Tangente (rot gestrichelt) im Übergangspunkt.

2.3. Längenberechnung von Kurven

Die Länge einer Kurve und somit auch die Länge einer Beziérkurve im Zweidimensionalen, wird durch das Integral aus Formel 2.5 beschrieben.

$$L(C(t)) = \int_a^b \sqrt{f'_x(t)^2 + f'_y(t)^2} dt \quad (2.5)$$

Dabei entsprechen $f_x(t)$ und $f_y(t)$ den Funktionen der x- und y-Koordinate der Beziérkurve. Allerdings kommt es bereits ab Kurven des Grades $n = 3$ zu Polynomen fünften oder höheren Grades im Wurzelausdruck. Diese sind nach dem Satz von Abel-Ruffini nicht mehr allgemein lösbar.

Zur Längenberechnung einer Beziérkurve wird demzufolge ein Näherungswert benötigt. Dazu wird die Kurve durch Strecken bekannter Länge angenähert, die sich einfach aufsummieren lassen. Algorithmus 1 bestimmt so näherungsweise die Länge einer Beziérkurve.

Die Länge einer Beziérkurve wird nachfolgend benötigt, um auf einer Kurve liegende Haltestellen in die gewünschten Abstände zueinander zu setzen.

Algorithmus 1: BeziérLänge(Beziérkurve B)

Eingabe : Beziérkurve B , Genauigkeitsgrenze ε , maximale Anzahl von Durchgängen I

Ausgabe : Länge der Beziérkurve B

```
1 counter = 0
2 N = 1
3 lengthA = Summe der Länge der N = 1 Strecken, die B annähern
4 lengthB = Summe der Länge der N · 10 Strecken, die B annähern
5 while |lengthA - lengthB| > ε and counter < I do
6   N = N · 10
7   lengthA = lengthB
8   lengthB = Summe der Länge der N · 10 Strecken, die B annähern
9   counter = counter + 1
10 return lengthB
```

3. Erstellen von Streckenplänen

In diesem Kapitel wird erläutert wie aus einer Eingabe von Bahnstationen und deren Verbindungen zueinander eine optimierte Zeichnung des dazugehörigen Liniennetzplanes entsteht. Der dazu genutzte Algorithmus stammt aus der Arbeit von Fink et al. [FHN⁺13] und wurde um den Einbezug der Fahrtzeiten erweitert. Es wird erörtert wie aus den Eingabedaten eine geradlinige Zeichnung entsteht und diese in eine Zeichnung mit Beziérkurven als Kanten transformiert wird. Zudem wird beschrieben welche Modifikationen weiter vorgenommen werden, um ein ansprechendes optischen Erscheinungsbild zu gewährleisten und wie über die Kantenlängen Hinweise auf Fahrtzeiten gegeben werden können.

3.1. Anfertigung einer geradlinigen Zeichnung

Bevor man die Zeichnung einer Bahnstrecke anfertigen kann, benötigt man die Koordinaten ihrer Haltestellen. Im Folgenden soll zunächst mit dem Liniennetzplan der Würzburger Straßenbahn gearbeitet werden, um grundlegende Vorgehensweisen zu erläutern. Hierzu wurden zu jeder Haltestelle der Name, die Linien, welche die jeweilige Haltestelle anfahren sowie die geographischen Koordinaten, in Grad nördlicher Breite und östlicher Länge, aus *OpenStreetMap* [GO14] ausgelesen. Die Gradangaben wurden zum Erstellen der Zeichnung in kartesische Koordinaten umgerechnet. Zudem wurden die Fahrtzeiten, welche die Straßenbahnen für ihre Strecken benötigen, den Plänen des *Verkehrsunternehmens-Verbundes Mainfranken (VVM)* [Gmb09] entnommen.

Aus den Koordinaten sämtlicher Haltestellen lässt sich dann ein *geradliniger* Graph erstellen, dessen Kanten durch Strecken dargestellt werden. Dieser Graph entspricht dem Würzburger Straßenbahnnetz, siehe Abbildung 3.1 und ist topographisch insofern korrekt, dass die Positionen der einzelnen Stationen zueinander, denen ihrer tatsächlichen Lage entsprechen. Die als Strecken dargestellten Kanten können von ihrem wirklichen Verlauf abweichen, da kurvenförmig angelegte Linienabschnitte nicht berücksichtigt werden. Die Zeichnung beinhaltet zusätzliche Informationen zur Orientierung, zum einen die Fahrtzeiten zwischen den Stationen in Minuten, zum anderen die Namen der Stadtteile. Außerdem gibt eine Legende weitere Angaben zu den einzelnen Linien.

Neben den Angaben der Reisezeit als Zahl, kann man teilweise aus den unterschiedlichen Streckenlängen zwischen zwei Stationen Rückschlüsse auf die Fahrtzeit ziehen. Hierbei gilt mit einigen Ausnahmen, je weiter zwei Haltestellen voneinander entfernt sind, desto länger ist die Fahrtzeit zwischen ihnen. Zum Teil würde dies aber zu falschen Vermutungen über die Reisezeit zwischen zwei Halten führen, da für das Zurücklegen kurzer Strecken teilweise mehr Zeit benötigt wird, als der Abstand vermuten lässt. Ebenso wird an manchen Stellen für den Weg zwischen zwei weiter voneinander entfernt liegenden Stationen weniger Zeit benötigt als man annehmen könnte.

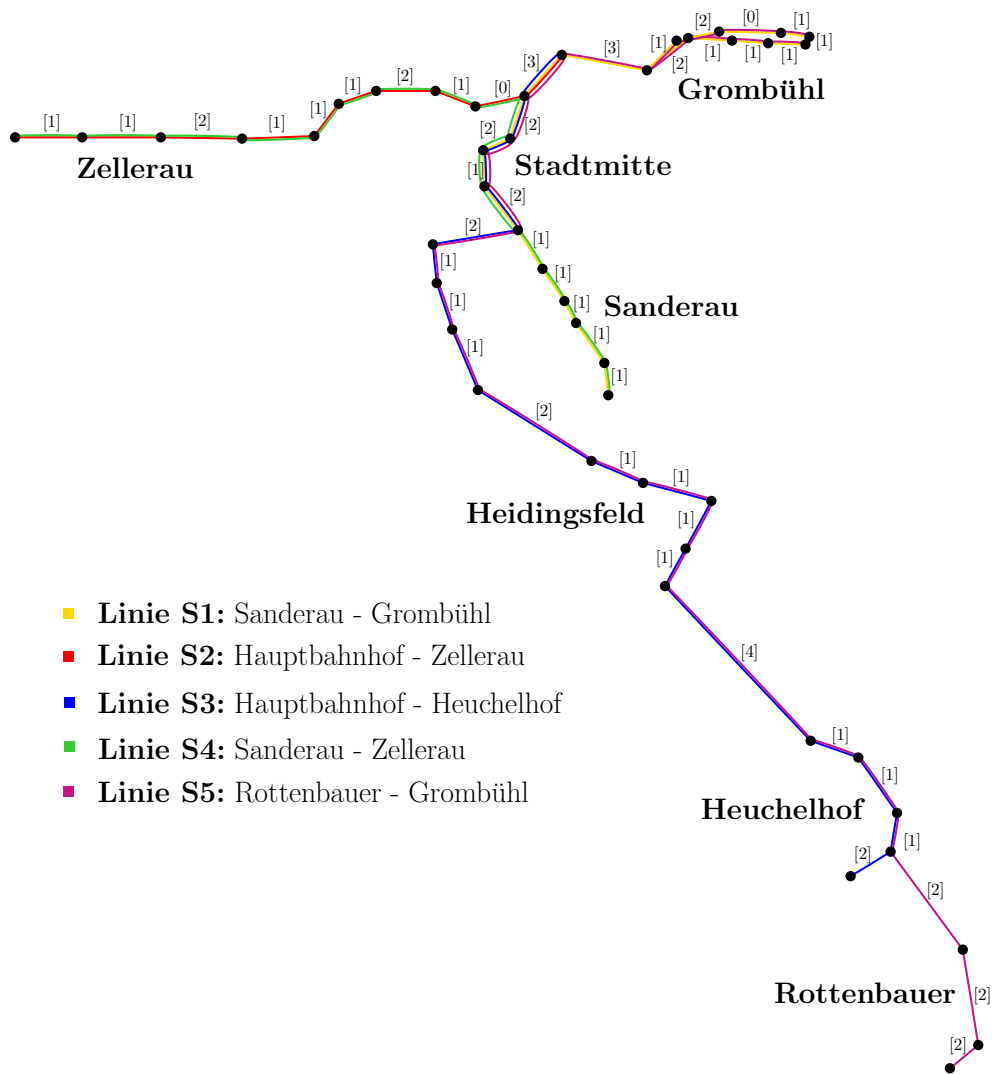


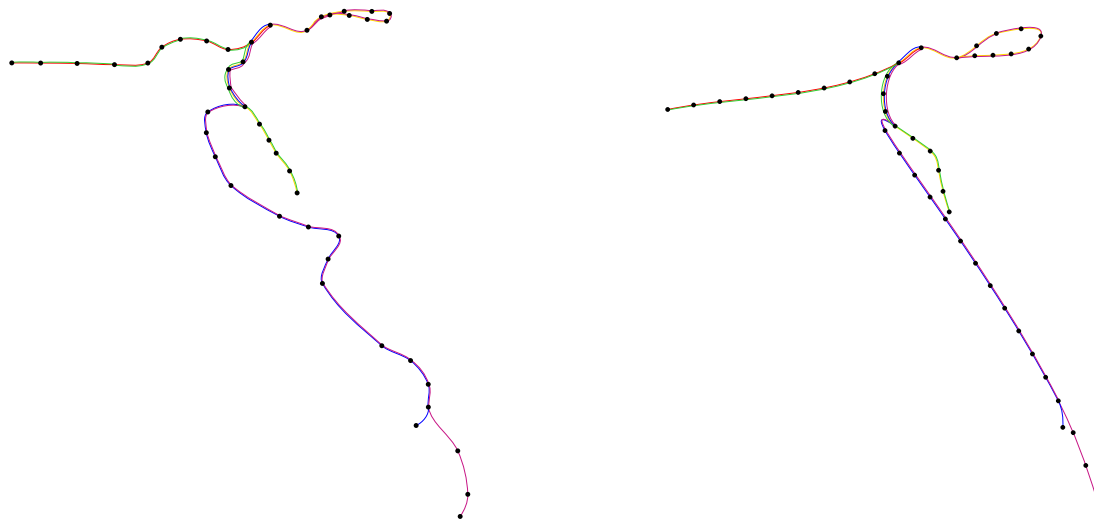
Abb. 3.1.: Plan des Würzburger Straßenbahnliniennetzes mit Reisezeiten. Fahrtzeit [0] entspricht einer Fahrtdauer von weniger als einer Minute.

3.2. Anfertigung einer Zeichnung mit Beziérkurven als Kanten

Nachfolgend wird auf den kräftebasierten Zeichenalgorithmus eingegangen. Es wird beschrieben wie die Eingabe in Form eines Graphen mit Strecken als Kanten in einen kreuzungsfreien Graphen mit beziérkurvenförmigen Kanten umgeformt wird. Weiter wird auf die Kräfte eingegangen, welche auf den Graphen wirken, um diesem ein gewünschtes Aussehen zu verleihen. Zuletzt erfolgt die Vereinfachung des Graphen, indem möglichst viele Kanten zu einer einzigen wohlgeformten Kante vereint werden.

3.2.1. Vorstellung des Zeichenalgorithmus

Ausgehend von einer geradlinigen Zeichnung lässt sich mit dem von Fink et al. [FHN⁺13] vorgestellten Algorithmus ein Graph mit Beziérkurven als Kanten zeichnen, vergleiche Abbildung 3.2. Hierbei wird entweder jede Kante durch eine eigene Kurve dargestellt, wie in Abbildung 3.2a, oder es wird versucht, möglichst viele Kanten in einer Beziérkurve zusammenzufassen, siehe Abbildung 3.2b. Auf das Zusammenfassen mehrerer Kanten wird in Abschnitt 3.2.4 näher eingegangen. Alle Beziérkurven in den beiden nachfolgenden Zeichnungen sind vom Grad $n = 3$.



(a) Darstellung jeder Kante durch eine eigene Beziérkurve.

(b) Zusammenfassung möglichst vieler Kanten zu einer Kurve.

Abb. 3.2.: Würzburger Straßenbahnnetz mit Beziérkurven als Kanten.

Wird jede Kante durch eine eigene Beziérkurve dargestellt, bleiben die geographischen Informationen zur Lage der Stationen zueinander teilweise erhalten. Durch die unterschiedlichen Abstände zwischen den Haltestellen ist, wie schon beim geradlinigen Graphen, ein erster Hinweis auf die Fahrtzeit gegeben. Werden dagegen möglichst viele Kanten zu einer Kurve vereint, gehen diese Informationen verloren, da die Haltestellen dann gleichmäßig auf der zusammengefassten Kurve verteilt liegen.

Die grundsätzliche Vorgehensweise zur Anfertigung eines optisch ansprechenden Graphen mit Beziérkurven als Kanten, aus einem geradlinigen Ausgangsgraphen, ist im kräftebasierten Algorithmus 2 beschrieben.

3.2.2. Erstellen des Ausgangsgraphen

Ausgangspunkt des Algorithmus zum Zeichnen der Bahnstrecken ist ein als ungerichteter Graph eingegebenes Streckennetz, wie aus Abbildung 3.1. Des Weiteren werden dem Algorithmus Informationen über die Zuordnung von Bahnlinien zu Kanten, die den Kanten

Algorithmus 2: Kräftebasierter Algorithmus(Ungerichteter Graph $G = (V, E)$)

Eingabe : Geradliniger, ungerichteter Graph $G = (V, E)$
Ausgabe : Optimierter Graph $G = (V, E)$

- 1 $I =$ maximale Anzahl Iterationen
- 2 $\varepsilon =$ Genauigkeitsgrenze
- 3 $counter = 0$
- 4 Erstelle kreuzungsfreien Graphen mit Beziérkurven als Kanten.
- 5 **while** $counter < I$ **and** maximale Knotenbewegung $> \varepsilon$ **do**
 - 6 Berechne Kräfte auf Knoten.
 - 7 Berechne Kräfte auf Kanten.
 - 8 Wende Kräfte auf Graphen G an.
 - 9 $counter = counter + 1$
- 10 **return** Optimierter Graph G .

zugeordneten Fahrtzeiten und die geographischen Positionen der Stationen übergeben. Das eingegebene Streckennetz wird zunächst in eine Zeichnung eines Graphen mit Beziérkurven als Kanten transformiert. Möglich ist dies, indem bei jeder Kante $e = \{u, v\}$ in sehr kurzer Distanz zu den Endpunkten zusätzlich die Punkte u' beziehungsweise v' auf die Strecke $[uv]$ gesetzt werden. Dadurch erhält man pro Kante vier ihr zugehörige Punkte. Diese vier Punkte spannen das Kontrollpolygon einer Beziérkurve vom Grad $n = 3$ auf. Die Tangenten in den Endpunkten dieser Kurve verlaufen demnach durch den jeweils gegenüberliegenden Endpunkt der Kante. Die Tangente durch u und u' verläuft also auch durch v . Nach dem Entfernen der ursprünglichen Strecken zwischen den Knoten erhält man eine Zeichnung des Graphen mit Beziérkurven als Kanten, welcher aussieht wie ein geradliniger Graph.

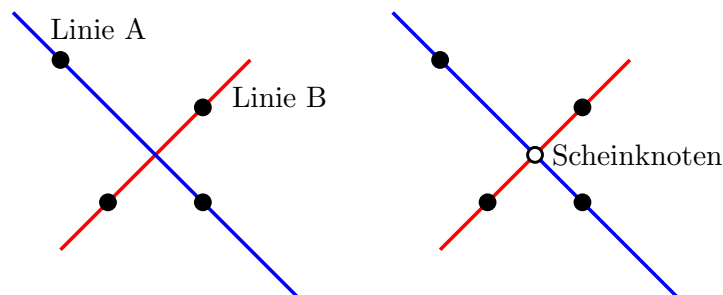


Abb. 3.3.: Sich kreuzende Linien A und B, rechts mit eingefügtem Scheinknoten.

Ist der Eingabegraph nicht, wie für den Algorithmus benötigt, *kreuzungsfrei*, es schneiden sich also mindestens zwei Kanten, so werden an die Schnittpunkte *Scheinknoten* gesetzt, siehe Abbildung 3.3. Diese repräsentieren keine Haltestelle, verhindern aber ein Kreuzen der Linien, da diese nun in einem Knoten zusammenlaufen. Die nachfolgend beschriebenen Kräfte wirken auf die Scheinknoten ebenso wie auf Knoten, welche Haltestellen repräsentieren. Allerdings werden die Scheinknoten nicht in die Ausgabe

ingezeichnet. Kreuzungen treten jedoch weder im Würzburger Straßenbahnnetz auf, noch in einem anderen in dieser Arbeit verwendeten Liniennetz. Aus praktischen Gründen werden an Kreuzungen meist tatsächlich Haltestellen errichtet, da diese das einfache Umsteigen zu anderen Linien ermöglichen. Selbst im ältesten und einem der ausgedehntesten U-Bahn-Netze der Welt, der London Underground, treten gerade einmal vier Kreuzungen auf. Diese stammen aus der Zeit als konkurrierende Unternehmen einzelne U-Bahnlinien betrieben [Wol04].

3.2.3. Kräfte und deren Berechnung

Kräfte zur Positionierung der Knoten. Der aus dem vorigen Schritt erhaltene, kreuzungsfreie Ausgangsgraph mit Beziérkurven als Kanten soll jetzt in eine optisch ansprechende Form gebracht werden. Dazu werden in mehreren Iterationen auf den Graphen wirkende Kräfte berechnet und auf diesen angewandt. Diese Kräfte ziehen Knoten und Kanten des Graphen in eine ansprechende Form. In jedem Schleifendurchlauf werden die berechneten Kräfte auf den momentanen Graphen angewandt. Für den als Ergebnis erhaltenen Graphen werden in der darauf folgenden Iteration erneut die entsprechenden Kräfte berechnet um daraufhin Anwendung zu finden. Alle Kräfte werden über experimentell ermittelte Vorfaktoren gewichtet. Diese gewährleisten einen stimmigen Ausgabegraphen.

Die Kräfte, denen die Knoten und Kanten ausgesetzt sind, setzen sich aus mehreren Faktoren zusammen. Auf die Knoten wirken unter anderem die von Fruchterman und Reingold [FR91] definierten Kräfte, vergleiche Formeln 3.1 und 3.2. Die abstoßenden Kräfte wirken auf alle Paare von Knoten u und v . Die anziehenden Kräfte streben an, dass benachbarte Knoten immer einen festen Abstand l zueinander haben. Um dies zu erreichen wirken auf benachbarte Knoten u und v sich entgegengesetzte, anziehende Kräfte.

$$F_{AB}(u, v) = \left(\frac{l}{d(u, v)} \right)^2 \cdot \vec{uv} \quad \text{mit} \quad d(u, v) = \sqrt{(x_u - x_v)^2 + (y_u - y_v)^2} \quad (3.1)$$

$$F_{AN}(u, v) = \frac{d(u, v)}{l} \cdot \vec{vu} \quad (3.2)$$

Liegen zwei Knoten nahe beieinander, so überwiegt die abstoßende Komponente der Kraft. Bei weit voneinander entfernt liegenden Knoten dominiert die anziehende Komponente der Kraft und die Knoten nähern sich einander an. In Abbildung 3.4 wird die Wirkungsweise der Kraft über mehrere Iterationen hinweg veranschaulicht.

Außerdem wirkt auf sämtliche Knoten eine Kraft, die anstrebt diese an ihrer ursprünglichen geographischen Position u_{geo} im Graphen zu halten, siehe Formel 3.3. Weicht die Position der Knoten im zu erstellenden Streckenplan zu stark von der eigentliche Lage der Haltestelle ab, ist es für den Betrachter mühsamer sich zurechtzufinden. Denn befindet sich eine im Interesse liegende Station nicht an der zu erwartenden Position in der

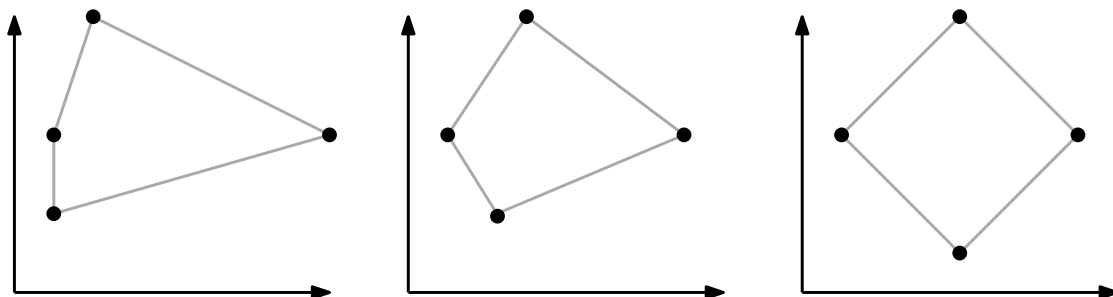


Abb. 3.4.: Anwendung der anziehenden und abstoßenden Kräfte von Fruchterman und Reingold. Die vier Punkte werden über mehrere Schritte hinweg in gleichmäßige Abstände zueinander gebracht.

Zeichnung, muss diese erst gesucht werden. Zudem gehen Informationen über die Lage der Halte zueinander verloren wenn sie ihre Position beliebig verändern können.

$$F_{\text{GEO}}(u) = \overrightarrow{uu_{\text{geo}}} \quad (3.3)$$

Zur Einbeziehung der Fahrzeit gibt es eine weitere Kraft, welche nicht wie die anderen Kräfte schon bei Fink et al. beschrieben wird. Diese versucht die Abstände zwischen den Knoten an die Fahrzeiten anzupassen, vergleiche Formel 3.4, in welcher die auf einen Knoten u wirkende Kraft beschrieben ist. Dabei ist $z(u, v)$ der Abstand, welcher der Fahrzeit entspricht. F_{ZEIT} wird auf alle benachbarten Knoten ausgeübt. Bei Kanten $e = \{u, v\}$, die einen Knoten mit dem Grad $n = 1$ enthalten, wird die Kraft ausschließlich auf diesen angewandt um die von ihm ausgehende Linie auf die gewünschte Länge $z(u, v)$ zu bringen. Der Betrag der Kraft auf den einzelnen Knoten ist hierbei größer, als sie es bei der Anwendung auf beide Endpunkte einer Kante wäre. Die Beschränkung der Kraft auf den Knoten mit Grad $n = 1$ hängt mit deren Position zusammen. Diese liegen meist in den äußeren Gebieten der Zeichnung, da es sich oft um Endhaltestellen in Vororten handelt. Die Positionierung dieser Knoten hat dadurch keinen Einfluss auf andere Knoten und Kräfte, die auf diese wirken. Endhaltestellen im Zentrum der Zeichnung haben gewöhnlich Grade größer als eins, da dort mehrere Linien enden. Also werden bei diesen und allen anderen Kanten, ohne Knoten vom Grad $n = 1$, die Kräfte auf beide Endpunkte angewandt, wobei die auf u angewandte Kraft der auf v angewandten Kraft entgegenwirkt. Der Betrag der Kraft für benachbarte Knoten einer Kante ist demnach gleich. Die Berechnung dieser Kraft orientiert sich an der bereits erwähnten Kraft von Fruchterman und Reingold.

$$F_{\text{ZEIT}}(u, v) = \frac{d(u, v)}{z(u, v)} \cdot \overrightarrow{vu} + \frac{z(u, v)^2}{d(u, v)^2} \cdot \overrightarrow{uv} \quad (3.4)$$

Tangenten und Kontrollpunkte beeinflussende Kräfte. Neben auf Knoten wirkenden Kräften existieren Kräfte, welche die Lage der Kontrollpunkte und Tangenten beeinflussen.

Die Kontrollpunkte u' und v' zum Erstellen einer Beziérkurve der Kante $e = \{u, v\}$, sollen nach Möglichkeit von u , beziehungsweise v im Abstand von einem Drittel der Länge der Strecke $[uv]$ liegen. Dieser Wert wurde experimentell ermittelt und sorgte hierbei für eine ansprechende Form der Beziérkurven. Liegen die Kontrollpunkte, im Vergleich zur Kurvenlänge, nahe an den Endpunkten der Kante, so erhält die Kurve an den Endpunkten eine stärkere Krümmung. Diese erschwert es Kurven glatt ineinander übergehen zu lassen. Sind die Kontrollpunkte hingegen zu weit von den Endpunkten entfernt, so wird die Kurve unnötig in die Länge gezogen, siehe Abbildung 3.5.

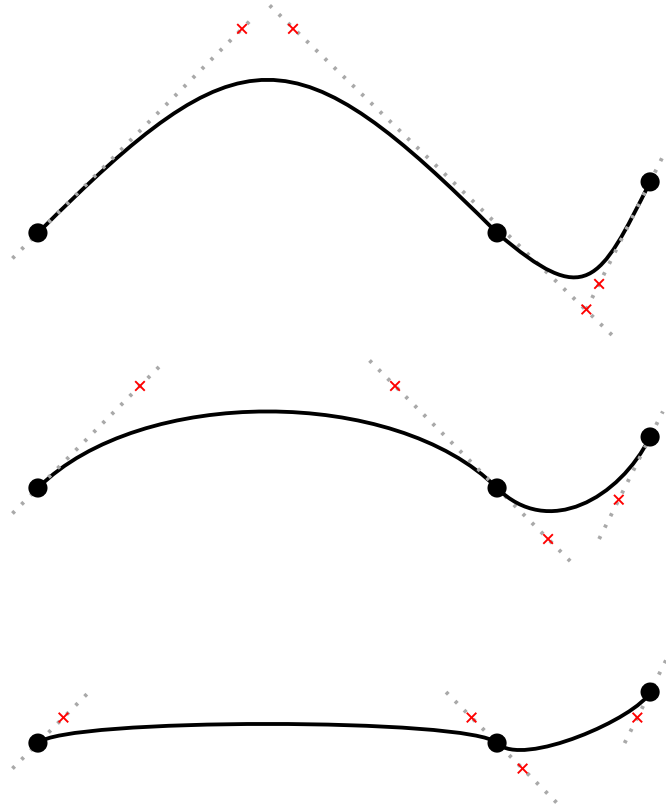


Abb. 3.5.: Zwei ineinander übergehende Beziérkurven mit von den Knoten unterschiedlich weit entfernten Kontrollpunkten.

Die Kraft, welche dazu verwendet wird die Kontrollpunkte in einen angemessenen Abstand zu setzen, orientiert sich an den von Fruchterman und Reingold definierten Kräften, vergleiche Formel 3.1 und 3.2. Sie hat eine anziehende und eine abstoßende Komponente, ist aber kein Vektor, sondern skalar. Die Kraft soll nämlich lediglich den Abstand zwischen einem Endpunkt und dem benachbarten Kontrollpunkt verändern und nicht die Orientierung der Tangente. Sie ändert also nur den Betrag der auf den Kontrollpunkt wirkenden Kraft.

$$F_{\text{KTR}}(u, u') = \left(\frac{(d(u, v)/3)^2}{d(u, u')} - \frac{d(u, u')^2}{d(u, v)/3} \right) \quad (3.5)$$

Um die Krümmung der Beziérkurven möglichst gering zu halten wirken begradigende Kräfte auf Knoten und Tangenten. Bei jeder Kante $e = \{u, v\}$ wird eine Kraft auf v ausgeübt, um sie der Tangente t durch u anzunähern, siehe Formel 3.6. Die Kraft zieht den Punkt v zum Punkt v_t auf der Tangente t . Die Punkte v und v_t haben zum Punkt u den gleichen Abstand, es gilt also $d(u, v) = d(u, v_t)$, siehe Abbildung 3.6a.

Zugleich wirkt eine Kraft auf die Tangente t durch den Knoten u , um diese an den Knoten v anzunähern, siehe Formel 3.7. Diese Kraft ist ebenfalls skalar, wie die Kraft aus Formel 3.5, da sie den Kraftvektor um einen bestimmten Winkel rotiert. Dabei handelt es sich um eine eindimensionale Bewegung. Zur Berechnung dieser Kraft werden zuerst alle Kanten bestimmt, welche durch u verlaufen und die t als Tangente besitzen. Für jede dieser Kanten wird der Winkel α zwischen ihrem geradlinigen Verlauf, der Strecke $[uv_i]$ und der Tangente t bestimmt. Die Winkel werden entsprechend der Kantenlänge gewichtet. Längere Kanten sorgen für eine größere Gewichtung, kürzere Kanten für eine geringere Gewichtung, vergleiche Abbildung 3.6b. Der ungünstige Verlauf einer kurzen Kante ist eher hinzunehmen, als der einer langen Kante. Diese Gewichtung erfolgt über den Abstand vom Knoten u zu seinem jeweiligen Kontrollpunkt u'_i

$$F_{\text{KRÜ-KNO}}(u, v) = \overrightarrow{vv_t} \quad (3.6)$$

$$F_{\text{KRÜ-TAN}}(t, u, v_1, \dots, v_k) = \frac{\sum_{i=1}^k \alpha_i \cdot d(u, u'_i)}{\sum_{i=1}^k d(u, u'_i)} \quad (3.7)$$

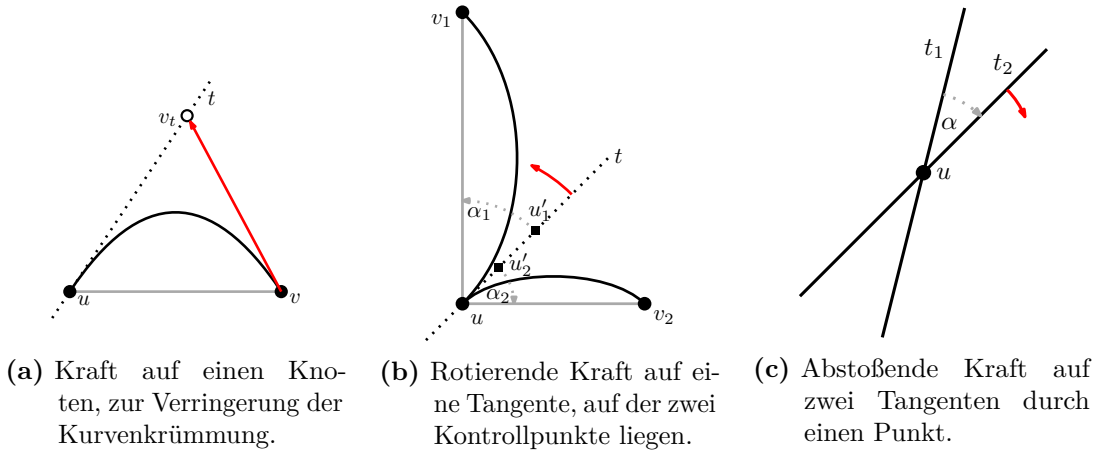


Abb. 3.6.: Auf Tangenten und Knoten wirkende Kräfte.

Zuletzt existiert eine Kraft, welche auftritt wenn mehrere Tangenten verschiedener Beziérkurven durch einen Punkt laufen. Diese Kraft wirkt in dem Maße, dass sich verschiedene Tangenten durch einen Punkt u voneinander abstoßen, vergleiche Formel 3.8. Dies soll verhindern, dass zwei Kanten die durch den selben Knoten verlaufen zu dicht aneinander liegen. Dazu wird auf jedes Tangentenpaar t_1, t_2 durch u eine abstoßende, skalare Kraft in Abhängigkeit des Winkels $\alpha(t_1, t_2)$ zwischen den Tangenten ausgeübt, siehe Abbildung 3.6c.

$$F_{\text{AB-TAN}}(t_1, t_2) = \frac{1}{\alpha(t_1, t_2)} \quad (3.8)$$

Vermeidung von Kreuzungen. Um zu verhindern, dass nach dem Anwenden der Kräfte Kreuzungen im Graphen auftreten, müssen die Kräfte eventuell eingeschränkt werden. Dazu wird zu jedem Paar von Kanten überprüft, ob die durch die momentanen Kräfte angestrebten Änderungen in deren Verlauf zu einer Kreuzung führen würden. Ist dies der Fall, so wird der Betrag aller Kräfte, welche auf Tangenten, Knoten und Kontrollpunkte dieser Kanten wirken halbiert. Dieser Schritt wird so oft wiederholt, bis die neu bestimmten Kräfte keine Kreuzung der Kanten nach sich ziehen.

3.2.4. Zeichnung und Vereinfachung von Kanten

Zusammenfassen von Kanten. Bei zwei Kanten $e_1 = \{u, v\}$ und $e_2 = \{v, w\}$, auf denen die selben Linien verkehren, entscheidet der Grad n des Knotens v darüber, ob die beiden Kanten zusammengefasst werden können. Ist v vom Grad zwei, wird geprüft ob eine neu eingefügte geradlinig gezeichnete Kante $e = \{u, w\}$ eine bereits vorhandene Kante kreuzen würde. Ist dies nicht der Fall, so werden e_1 und e_2 aus dem Graphen entfernt und e wird diesem hinzugefügt. Würde eine Kreuzung entstehen so werden e_1 und e_2 beibehalten und e wird verworfen. In sämtlichen Testszenerarien wurden aber nahezu alle Kanten mit einem Knoten vom Grad $n = 2$ als Endpunkt entfernt. Die Reisezeit für die neu eingefügte Kante e wird aktualisiert und die für e aus dem Graphen entfernten Kanten werden in einer Liste gespeichert. Im Idealfall lassen sich so sämtliche, zu einer Bahnlinie gehörenden Kanten zwischen zwei Stationen zu einer Beziérkurve zusammenfassen. Diese würde dann von der ersten Haltestelle der Strecke bis zu deren Endstation verlaufen. Auf einer Linie liegen aber meist auch Knoten mit einem Grad größer als zwei. Hier ist das Vereinen der Kanten deutlich komplizierter. Deswegen wird sich im Algorithmus auf das Zusammenfügen von Kanten, über einen Knoten vom Grad $n = 2$ beschränkt. In Abbildung 3.2 ist der Liniennetzplan der Würzburger Straßenbahn mit möglichst vielen zusammengefassten Kanten und ohne Zusammenfassung der Kanten zu sehen.

Einzeichnen von Haltestellen auf einer Kante. Nach sämtlichen Anwendungsschritten der Kräfte auf die zu Beziérkurven zusammengefügte Kanten werden die einzelnen Stationen eingefügt. Diese werden nun nicht mehr wie im Algorithmus von Fink et al. gleichmäßig auf der Beziérkurve verteilt. Die Stationen werden stattdessen in Abständen,

die der Fahrtzeit z zwischen ihnen entsprechen, platziert. Dazu wurden beim Vereinen der Kanten die Fahrtzeiten der entfernten Kanten zwischengespeichert. Mit Hilfe der zuvor gespeicherten Zeiten lassen sich nun die Abstände zwischen den einzuzeichnenden Stationen auf der Kurve bestimmen. Ausgehend von einem Endpunkt u als Ausgangspunkt wird der Abstand zum nächsten Knoten k auf der Kurve mit dem anderen Endpunkt v durch $d_{\text{BEZ}}(u, k) = d_{\text{BEZ}}(u, v) / z_{\{u, v\}} \cdot z_{\{u, k\}}$ bestimmt. Aus den bereits für die Approximation der Länge der Beziérkurve berechneten Punkten wird nun jener als Punkt k gewählt, dessen Abstand von u der Länge $d_{\text{BEZ}}(u, k)$ am ehesten entspricht. So lassen sich alle zuvor entfernten Knoten vom Grad zwei wieder in den Graphen einzeichnen. Der Unterschied von gleichmäßig verteilt liegenden Knoten, zu Knoten in Abständen ihrer Fahrtzeit entsprechend, kann in Abbildungen 3.2b und 4.1, des Würzburger Straßenbahnnetzes betrachtet werden.

4. Ergebnisse des Algorithmus

Im nun folgenden Kapitel werden Zeichnungen präsentiert, welche vom zuvor erläuterten Algorithmus angefertigt wurden. Dabei handelt es sich um Liniennetzpläne aus den Städten Würzburg, Montreal, Wien und Sydney. Die entstandenen Pläne werden auf die Abweichung von der gewünschten Länge bei Kanten analysiert und dementsprechend bewertet.

4.1. Entstandene Zeichnungen

Würzburger Straßenbahnnetz. Angewandt auf den Streckenplan der Würzburger Straßenbahn aus Abbildung 3.1 erhält man durch den Algorithmus die in Abbildung 4.1 dargestellte Ausgabe. Streckenabschnitte, für welche die Fahrtzeit weniger als eine Minute beträgt, werden dabei so behandelt, als wäre die ihnen zugeordnete Fahrtzeit gleich einer Minute. In diesem und in den Streckenplänen von Wien und Montreal, vergleiche Abbildungen 4.2 und 4.3, sind Kanten, welche kaum von der gewünschten Länge abweichen schwarz eingefärbt. Kanten deren Länge um mehr als zehn Prozent von der angestrebten Länge differieren sind farbig markiert. Für das Erstellen aller Zeichnungen wurden die Kräfte über 500 Iterationen hinweg angewandt. Der Algorithmus benötigte dabei im langsamsten Fall, der Erstellung der Zeichnung des Graphen der Vorortslinien von Sydney, etwa fünf Sekunden.

Bei der Anwendung des Algorithmus auf das Würzburger Straßenbahnnetz konnten sämtliche Kanten mit Knoten vom Grad eins auf die gewünschte Länge gebracht werden. Probleme traten im zyklischen Teilabschnitt des Graphen im rechten oberen Bereich auf. Hier liegen eine zu kurz und eine zu lang gezeichnete Kante beieinander. Diese besitzen einen gemeinsamen Endpunkt. Würde man diesen verschieben um die zu kurz gezeichnete Kante zu verlängern, so würde sich auch die bereits zu lange Kante vergrößern. Mit den ebenfalls bereits korrekt eingezeichneten anderen benachbarten Kanten wird eine weitere Anpassung der Kantlänge verhindert. Der Grund, weshalb die zentral im oberen Bereich der Zeichnung gelegene Kante nicht länger eingezeichnet wird, konnte nicht eindeutig bestimmt werden. Es scheinen Kräfte vorzuliegen, die ihrer Ausdehnung entgegen wirken.

U-Bahn-Netz der Stadt Montreal. Der Eingabegraph der *Metro Montreal* aus der Arbeit von Fink et al. [FHN⁺13] ergibt die in Abbildung 4.2 zu sehende Zeichnung. Die Fahrtzeiten der Linien des U-Bahn-Netzes der Stadt Montreal wurden dem Routenplaner von *Google Maps* entnommen [Inc15]. Die Fahrzeiten zwischen den einzelnen Stationen in Montreal sind sich sehr ähnlich, deswegen sind auch die Abstände zwischen den Haltestellen sehr gleichmäßig verteilt.

Auch hier ist gut zu erkennen, dass Kanten, welche im Außenbereich der Zeichnung enden, die angestrebten Längen erreicht haben. Im zentral gelegenen zyklischen Teilabschnitt sind dagegen eine Kanten zu kurz und eine Kante zu lang eingezeichnet. Der Grund dafür, dass eine Kante zu kurz eingezeichnet wird, könnte sein, dass eine andere Kante fast parallel zu ihr verläuft und diese bereits die korrekte Länge hat. Dadurch könnte eine weitere Anpassung der zu kurzen Kante verhindert werden.

U-Bahn-Netz der Stadt Wien. Wird der Algorithmus auf das Wiener U-Bahn-Netz angewendet, so ergibt sich Abbildung 4.3. Der Graph stammt ebenfalls aus der Arbeit von Fink et al. Die Fahrtzeiten wurden aus den Fahrplänen der *Wiener Linien (WL)* [KG14] entnommen.

Diese Zeichnung ist komplexer als die beiden Vorgegangenen und besitzt mehr als doppelt so viele Kanten wie diese. Dennoch konnten wie auch zuvor sämtliche Kanten mit Endhaltestellen vom Grad eins auf die beabsichtigte Länge gebracht werden. Im Zentrum der Zeichnung, in welchem viele Stationen auf engem Raum beieinander liegen, konnten Kantenlängen nicht mehr den ihnen zugeordneten Fahrtzeiten angepasst werden. Es überwiegen dennoch, wie auch bei den beiden vorab besprochenen Zeichnungen, die Kanten mit Maßen entsprechend ihrer Fahrtzeit.

Vorortslinien von Sydney. Im Liniennetzplan der Vorortslinien von Sydney, siehe Abbildung 4.4, wurden einzelnen Linien nicht entsprechend ihrer Kantenlänge eingefärbt. Stattdessen wurde in der Zeichnung ein Bereich markiert, in welchem viele Kanten liegen, deren Länge nicht auf den gewünschten Wert gebracht werden konnten. Zudem liegen in diesem Gebiet viele Stationen auf engem Raum zusammen, während sie in anderen Bereichen weitläufiger verteilt sind. Dies begründet sich darin, dass es sich bei der markierten Zone unter anderem um die Innenstadt Sydneys handelt. Dort liegen Haltestellen dicht beieinander, somit sind die Fahrtzeiten zwischen ihnen gering. Je weiter man in die Vororte von Sydney gelangt, umso weiter liegen die Stationen auseinander und die Fahrtzeiten zwischen ihnen erhöhen sich. Somit entzerrt sich die Darstellung in den Außenbereichen der Zeichnung. Zudem wurde bei der Erstellung des Graphen keine Rücksicht auf Kreuzungen von Kanten genommen, um eine geeignete Ausgabe zu erhalten.

Auch dieser Eingabegraph stammt aus der Arbeit von Fink et al., die Positionen der einzelnen Haltestellen wurden jedoch von ihrer Angabe in geographischen Koordinaten in kartesische Koordinaten umgerechnet. Die Fahrtzeiten der *Sydney Trains* wurden den offiziellen Fahrplänen entnommen [Tra14].

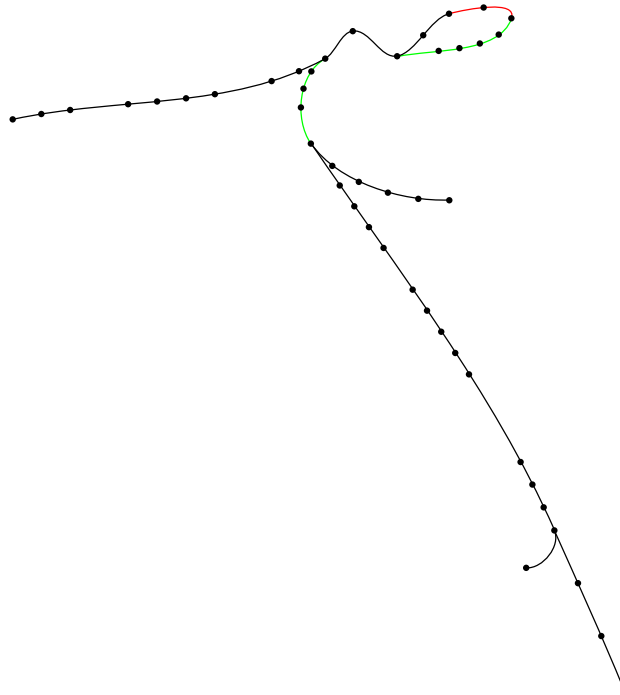


Abb. 4.1.: Liniennetzplan der Würzburger Straßenbahn mit Anpassung der Kantenlängen an die Fahrtzeiten. Rot eingefärbte Kanten sind mindestens 10% zu lang eingezeichnet, grüne Kanten mindestens 10% zu kurz.

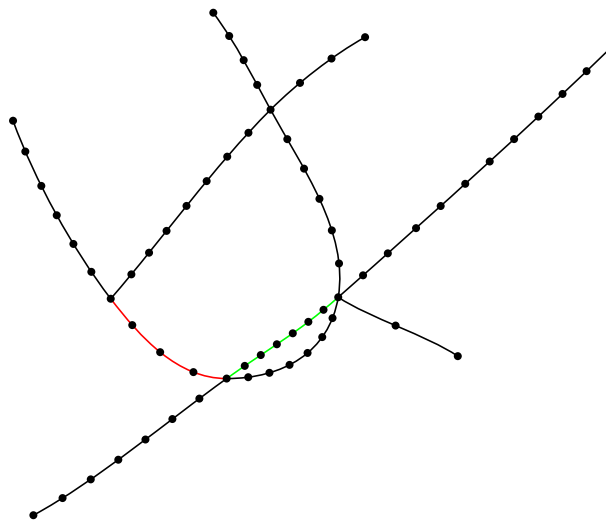


Abb. 4.2.: Liniennetzplan der U-Bahn von Montreal mit Anpassung der Kantenlängen an die Fahrtzeiten. Rot eingefärbte Kanten sind mindestens 10% zu lang eingezeichnet, grüne Kanten mindestens 10% zu kurz.

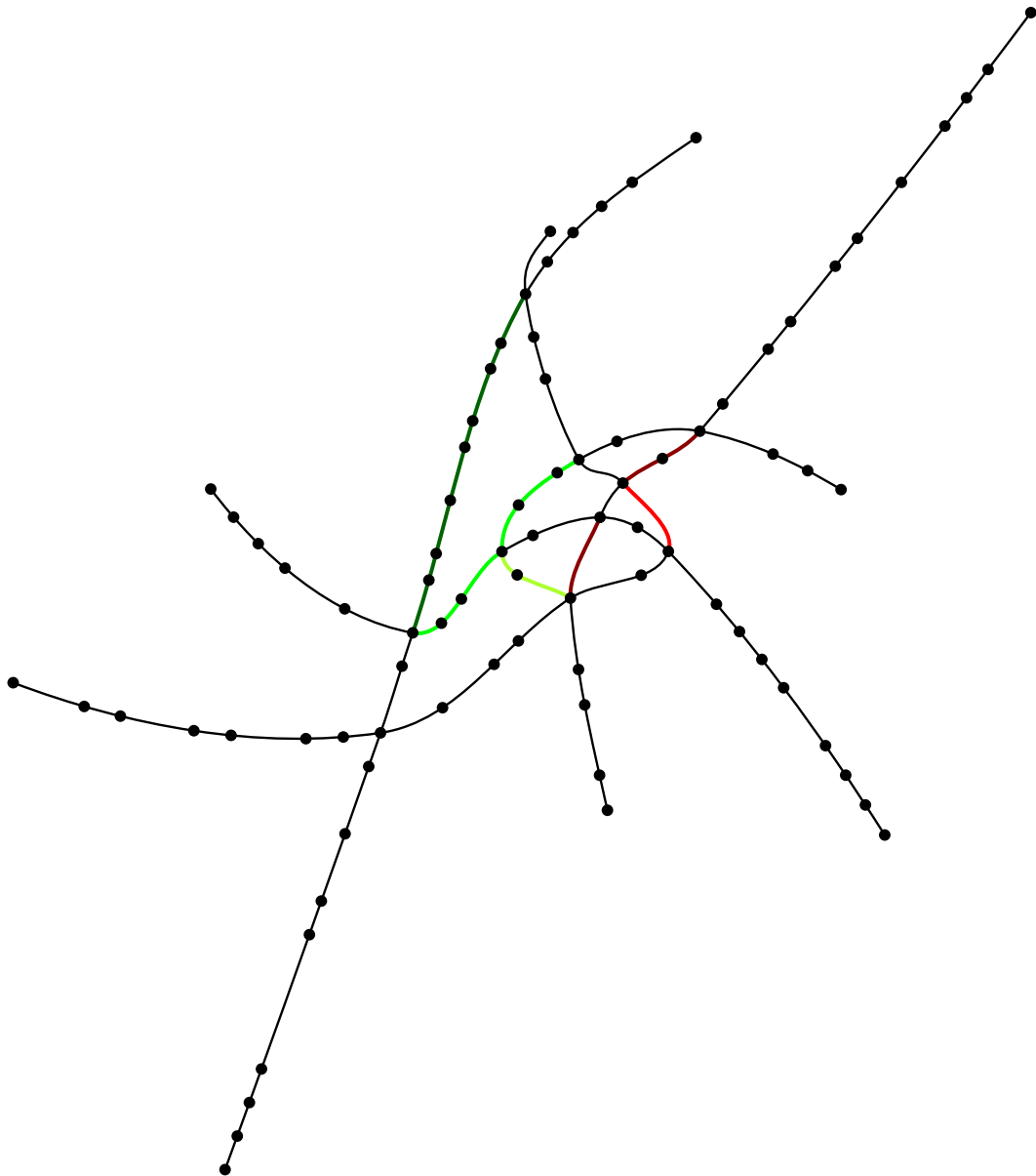


Abb. 4.3.: Liniennetzplan der U-Bahn von Wien mit Anpassung der Kantenlängen an die Fahrtzeiten. Hellrot eingefärbte Kanten sind mindestens 10% zu lang eingezeichnet, dunkelrote Kanten mindestens 15%. Grün gefärbte Kanten sind dagegen zu kurz eingezeichnet. Hellgrüne Kanten weichen um mindestens 10%, grüne Kanten um mindestens 15% und dunkelgrüne Kanten um mindestens 20% von der gewünschten Länge ab.

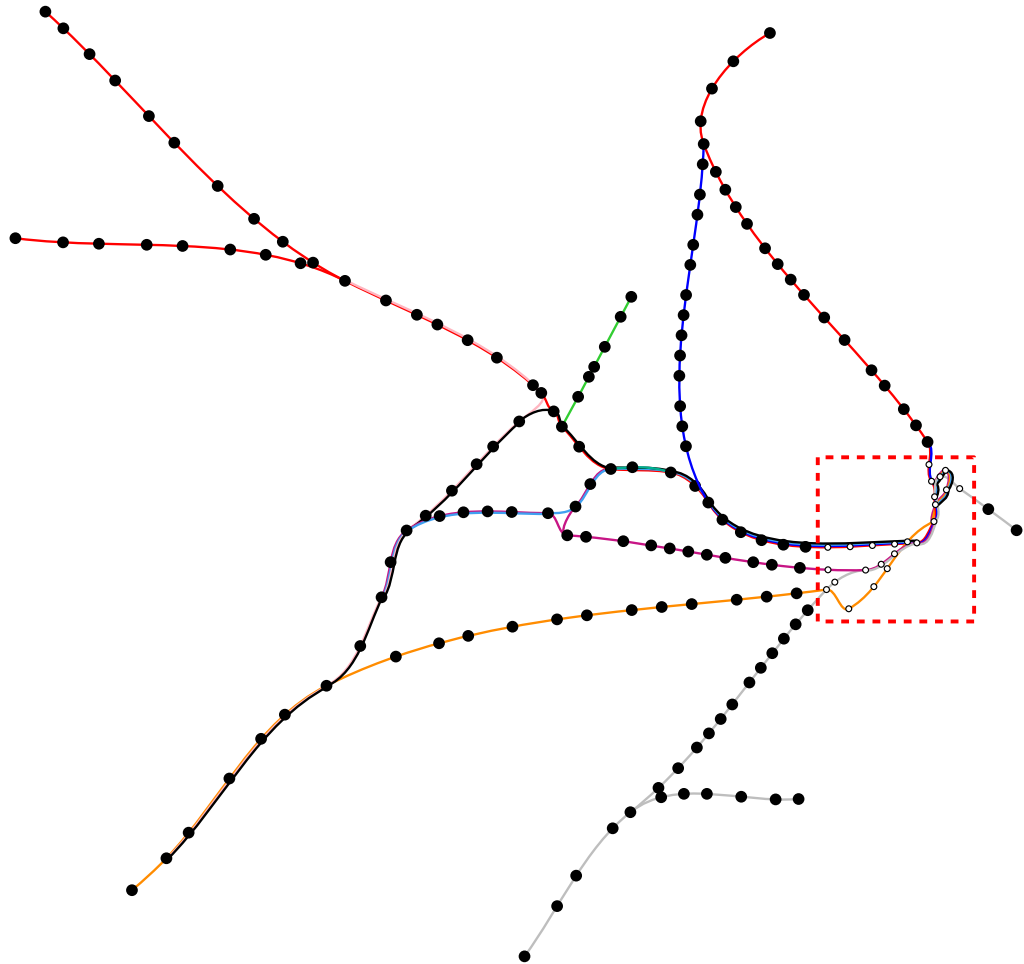


Abb. 4.4.: Liniennetzplan der Vorortslinien von Sydney mit Anpassung der Kantenlängen an die Fahrtzeiten. Kanten sind gemäß der auf ihnen verkehrenden Linie eingefärbt. Im rot markierten Bereich liegen überwiegend Kanten, deren gewünschte Länge nicht erreicht wird. Die Größe der darin liegenden Knoten wurden manuell angepasst.

4.2. Abweichungen von der gewünschten Kantenlänge

Um den Grad der Abweichung der einzelnen Kanten von ihrer gewünschten Länge zu quantifizieren wurden zwei Größen genutzt. Zum einen die mittlere Abweichung einer Kante von ihrer gewünschten Länge, vergleiche Formel 4.1 und zum anderen die mittlere quadratische Abweichung, siehe Formel 4.2.

$$\bar{x} = \frac{\sum_{i=1}^n |g_i - l_i|}{n} \quad (4.1)$$

$$\bar{x}^2 = \frac{\sum_{i=1}^n |g_i - \bar{x}|^2}{n} \quad (4.2)$$

Um die mittlere Abweichung \bar{x} der Kantenlänge zu erhalten, wird für alle Kanten i die tatsächliche Länge l_i von der gewünschten Länge g_i abgezogen. Die Beträge der Differenzen werden aufsummiert und durch die Anzahl der Kanten n geteilt. Bei der quadratischen mittleren Abweichungen werden größere Abweichungen von der gewünschten Länge stärker berücksichtigt, kleinere Abweichungen von der angestrebten Kantenlänge fließen weniger stark in den Wert mit ein. Dazu werden die Differenzen der gewünschten Kantenlänge und der mittleren Abweichungen quadriert und aufsummiert.

Für die zuvor abgebildeten Graphen ergeben sich so die in Tabelle 4.1 aufgelisteten Abweichungen. Für kleine und mittelgroße Liniennetze, wie die von Würzburg, Montreal und Sydney, ergeben sich geringere Abweichungen als beim großen Liniennetz von Sydney. Betrachten man nur die kleineren Netze, liegen die mittleren Abweichungen jeweils bei etwa zehn Prozent. Die quadratischen Abweichungen schwanken stärker, abhängig von der durchschnittlichen Kantenlänge. Beim großen Netz von Sydney, mit annähernd doppelt so vielen Kanten wie dem von Wien, liegt die mittlere Abweichung bei etwa 20 %.

| NAME DER STADT | ANZAHL KANTEN | \bar{x} in % | \bar{x} | \bar{x}^2 |
|----------------|---------------|----------------|-----------|-------------|
| Würzburg | 11 | 11,08 | 24,22 | 918,92 |
| Montreal | 11 | 8,11 | 5,71 | 10,48 |
| Wien | 25 | 8,68 | 6,37 | 103,15 |
| Sydney | 42 | 22,23 | 13,28 | 231,75 |

Tab. 4.1.: Mittlere Abweichungen der Kantenlängen von der gewünschten Kantenlänge.

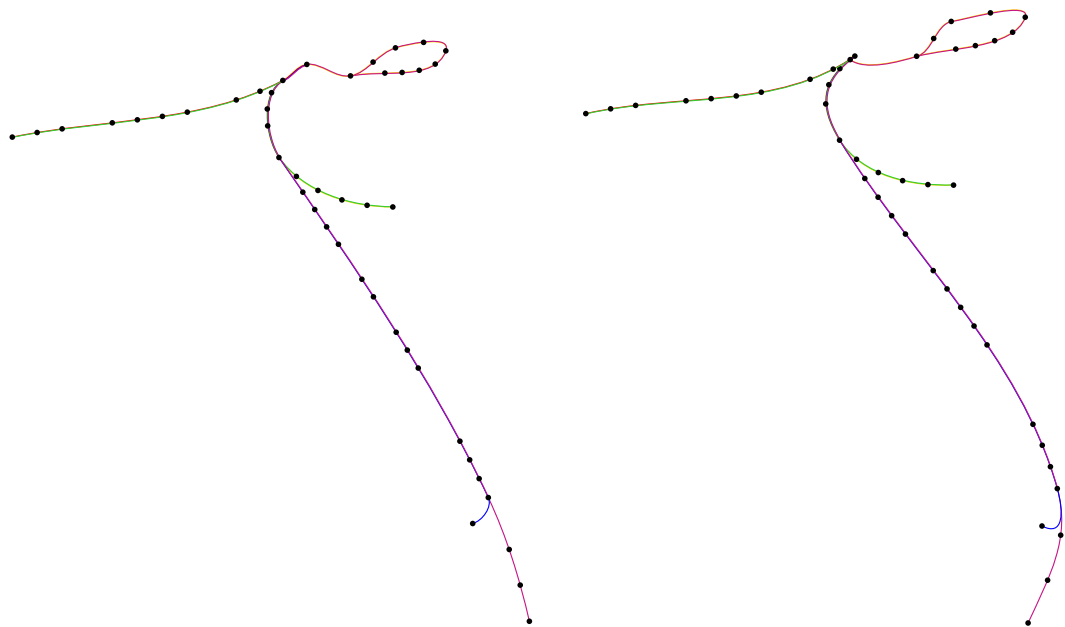
Trotz der Abweichungen von Kantenlängen vom gewollten Wert ist in den Zeichnungen der kleinen bis mittleren Netze gut zu erkennen, dass die Knoten nicht gleichmäßig auf den Kanten verteilt liegen. Aus den Abständen zwischen den Knoten lassen sich Schlussfolgerungen über die Fahrtzeit zwischen ihnen ziehen. So können Routen entsprechend ihrer Reisezeit bewertet werden.

Im Liniennetz von Sydney sind in den Bereichen der Vororte ähnliche Erkenntnisse wie in den kleineren Netzen möglich. Im Bereich der Innenstadt lassen sich jedoch anhand des Planes keine genauen Aussagen mehr über die Fahrtzeit zwischen einzelnen Stationen treffen.

4.3. Einfluss des Vorfaktors auf im Graphen wirkende Kräfte

Wie schon in Abschnitt 3.2.3 erwähnt, werden für alle Kräfte auf den Graphen Vorfaktoren ermittelt, um die Zeichnung möglichst ansprechend zu gestalten. Diese wurden für jede Kraft experimentell ermittelt und werden vor deren Anwendung mit dieser multipliziert. Die von Fink et al. ermittelten Vorfaktoren für im Graphen wirkende Kräfte wurden beibehalten. Für F_{ZEIT} wurde ein Faktor von etwa 1.1 ermittelt.

Wie sich eine Veränderung des Vorfaktors auf die Ausgabezeichnung auswirkt ist in Abbildung 4.5 zu sehen. Ein zu klein gewählter Vorfaktor bewirkt, dass die Kraft entsprechend der Fahrtzeit kaum berücksichtigt wird. Die Zeichnung entspricht dann in etwa der des ursprünglichen Algorithmus, vergleiche Abbildung 4.5a. Die durchschnittliche Abweichung einer Kante von der gewünschten Länge beträgt hier 16,19 %. Ist der Vorfaktor zu groß gewählt, werden Kanten zu sehr gestreckt oder verkürzt, vergleiche Abbildung 4.5b, die mittlere Abweichung beträgt hier 19,50 %. Dies ist am zyklischen Abschnitt im rechten oberen Bereich der Zeichnung gut zu erkennen.



(a) Vorfaktor nahe null gewählt.

(b) Vorfaktor dreimal größer als in Abbildung 4.1 gewählt.

Abb. 4.5.: Einfluss des Vorfaktors von F_{ZEIT} auf die Zeichnung des Straßenbahnnetzes von Würzburg.

In Abbildung 4.6 ist dargestellt, wie sich eine Änderung des Vorfaktors von F_{ZEIT} auf das U-Bahn Netz der Stadt Wien auswirkt. Ist der Vorfaktor sehr klein gewählt, so ist der Einfluss der Kraft, welche Kantenlängen ihren Fahrtzeiten anpasst, sehr gering., siehe Abbildung 4.6a. Die durchschnittliche Abweichung der Kantenlängen beträgt hier

etwa 13,23%. Wird der Vorfaktor sehr groß gewählt, wie in Abbildung 4.6b, so werden Kanten zu sehr in die Länge gezogen oder gestaucht. Dadurch ergibt sich eine mittlere Abweichung von der gewünschten Kantenlänge von 21,02%.

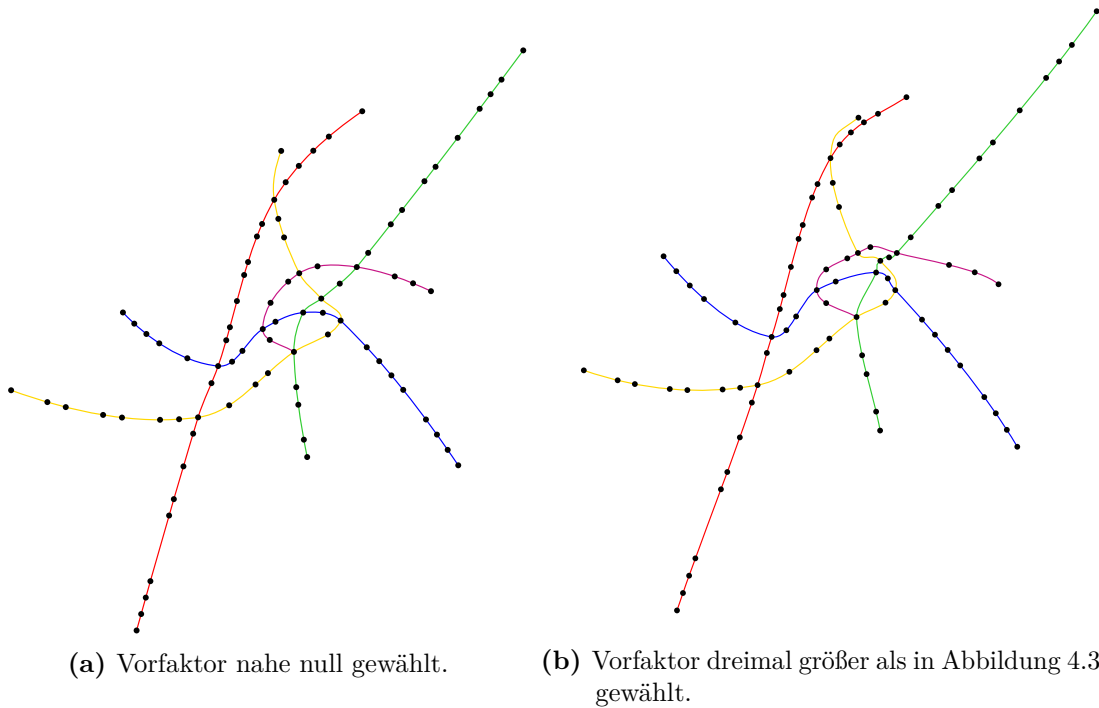


Abb. 4.6.: Einfluss des Vorfaktors von F_{ZEIT} auf die Zeichnung des U-Bahn Netzes der Stadt Wien.

5. Fazit und Ausblick

Mit dem weiterentwickelten Algorithmus von Fink et al., der in Kapitel 3 vorgestellt und erklärt wurde, ist es möglich die Kantenlängen eines Graphen zu beeinflussen. Der Algorithmus wurde auf Eingaben von Liniennetzplänen angewandt, um deren Kantenlängen entsprechend der Fahrtzeiten anzupassen. Dazu wurde der kräftebasierte Algorithmus um eine Kraft F_{ZEIT} erweitert. Zudem werden bei der Zusammenlegung von Kanten wegfallende Knoten abschließend nicht mehr gleichmäßig auf der neu eingefügten Kante verteilt, sondern in Abständen gemäß ihrer Fahrtzeiten zwischen einander gesetzt.

Die durch die Anwendung des Algorithmus entstandenen Zeichnungen wurden in Kapitel 4 dargestellt. Nicht alle Kanten haben bereits die gewünschte Länge, im Mittel weichen sie etwa zehn Prozent davon ab, doch geht aus den Abbildungen hervor, dass die Stationen unterschiedlich weit auseinander liegen. Es sind somit Rückschlüsse auf die Fahrtzeiten zwischen zwei Stationen möglich. Dadurch können zwei Routen auf ihre Fahrtdauer hin miteinander verglichen werden.

Jedoch wurde der Algorithmus nur an kleinen und mittelgroßen Netzwerken getestet. Für große Netzwerke, wie dem der Londoner U-Bahn, müssten noch weitere Optimierungen bezüglich der Wirkungsweise der Kräfte vorgenommen werden.

Der Algorithmus könnte weitergehend dahin verbessert werden, dass eine weitere Kraft eingeführt wird, welche Knoten von Kanten abstößt. Das führt dazu, dass von einem Knoten in ähnliche Richtungen ausgehende Kanten übersichtlicher dargestellt werden, veranschaulicht in Abbildung 5.1a. So würden sehr dicht aneinander liegende Knoten, wie sie beispielsweise in der Zeichnung des Liniennetzes von Sydney vorkommen, siehe Abbildung 5.1b, vermieden.

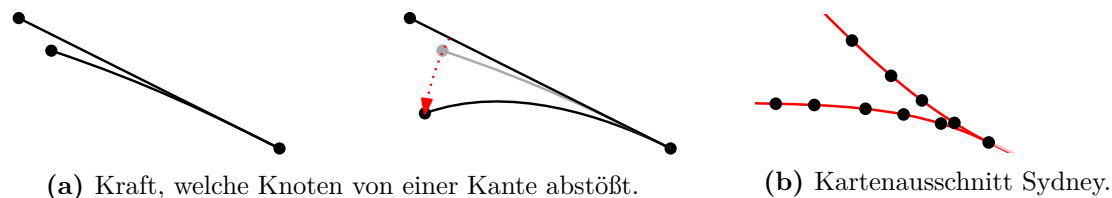


Abb. 5.1.: Veranschaulichung einer Kraft, welche Knoten von Kanten abstößt mit Beispiel für einen Anwendungsbereich.

Weiter könnten Kanten, welche durch den Algorithmus in der Zeichnung eines Graphen zu kurz dargestellt sind, durch eine zusätzlich eingefügte Krümmung verlängert werden, siehe Abbildung 5.2. Beim Einfügen der Krümmung muss allerdings erneut geprüft werden, ob der neue Kurvenverlauf zu einer Kreuzung mit einer anderen Kurve führt. Um dies zu vermeiden muss die Krümmung gegebenenfalls weiter angepasst werden.



Abb. 5.2.: Verlängerung einer Kurve durch das Einfügen einer zusätzliche Krümmung.

Zudem werden Linien oft durch mehrere Beziérkurven dargestellt. Schöner wäre es für jede Linie eine durchgehende Kurve zu erhalten. Interessant wäre es zu ermitteln, ob es generell möglich ist solche durchgehende Kurven zu erhalten. Falls dies nicht möglich ist könnte man weiter untersuchen, in welchen Fällen es nicht durchführbar ist sämtliche Kanten einer Linie zusammenzufassen. Eine besondere Rolle spielen hierbei Knoten vom Grad drei und höher, die eine solche Zusammenlegung beträchtlich erschweren. Verbesserungen an dieser Stelle würden zu optisch ansprechenderen Kurven mit gleichmäßiger Krümmung führen.

Bei Zeichnungen mit unüberschaubaren Ballungsräumen, wie der Innenstadt Sydneys, siehe Abbildung 4.4, könnte man als zusätzliche Option bestimmte Gebiete in den Fokus setzen, wie in der in Kapitel 1 beschriebenen Methode von Haurert und Sering. Das Verhältnis von Kantenlänge zu Fahrtzeit wird in dieser Region verfälscht, die Zeichnung gewinnt aber an Übersichtlichkeit.

Literaturverzeichnis

- [BBB87] Richard H. Bartels, John C. Beatty, and Brian A. Barsky: *An Introduction to Splines for Use in Computer Graphics and Geometric Modeling*. Morgan Kaufmann Publishers, Inc., San Mateo, 1987, ISBN 1-55860-400-6.
- [Bli13] Antje Blinda: Falsche Richtung: ICE verfährt sich auf dem Weg nach Berlin. Website, September 2013. <http://www.spiegel.de/reise/aktuell/fulda-ice-verfaehrt-sich-auf-dem-weg-nach-berlin-a-924753.html> [30. November 2014].
- [Bun14] Statistisches Bundesamt: 1. Halbjahr 2014: Mehr Fahrgäste in Bussen und Bahnen. *Pressemitteilung*, 333/14, September 2014.
- [BvGH⁺14] Kevin Buchin, Arthur van Goethem, Michael Hoffmann, Marc van Kreveld, and Bettina Speckmann: Travel-time maps: Linear cartograms with fixed vertex locations. In Matt Duckham, Edzer Pebesma, Kathleen Stewart, and Andrew U. Frank (editors): *Geographic Information Science - 8th International Conference, GIScience 2014, Vienna, Austria, September 24-26, 2014. Proceedings*, volume 8728 of *Lecture Notes in Computer Science*, pages 18–33. Springer-Verlag Berlin Heidelberg, September 2014.
- [FHN⁺13] Martin Fink, Herman Haverkort, Martin Nöllenburg, Maxwell Roberts, Julian Schuhman, and Alexander Wolff: Drawing metro maps using bézier curves. In Walter Didimo and Maurizio Patrignani (editors): *Graph Drawing - 20th International Symposium, GD 2012, Redmond, WA, USA, September 19-21, 2012, Revised Selected Papers*, volume 7704 of *Lecture Notes in Computer Science*, pages 463–474. Springer-Verlag Berlin Heidelberg, September 2013.
- [FR91] Thomas M. J. Fruchterman and Edward M. Reingold: Graph drawing by force-directed placement. *Software: Practice and Experience*, 21(11):1129—1164, November 1991.
- [Gal81] Francis Galton: On the construction of isochronic passage charts. *Proceedings of the Royal Geographical Society*, 11:657–658, November 1881.
- [Gmb09] Verkehrsunternehmens Verbund Mainfranken GmbH: Fahrplan für die Straßenbahnlinien und Omnibusnachtverbindungen. Website, April 2009. <http://www.vvm-info.de/fahrplaene/1767930.pdf> [17. Dezember 2014].
- [GO14] Geofabrik GmbH und OpenStreetMap: Kartendaten von OpenStreetMap, ODbL 1.0. Website, 2014. http://tools.geofabrik.de/map/#12/49.7945/9.9503&type=Geofabrik_Standard [6. Dezember 2014].

- [Hav14] Herman Haverkort: Embedding cues about travel time in schematic maps. Schematic Mapping Workshop 2014, February 2014.
- [HS11] Jan Henrik Haunert and Leon Sering: Drawing road networks with focus regions. *IEEE Transactions on Visualization and Computer Graphics*, 17:2555–2562, Dezember 2011.
- [Inc15] Google Inc.: Kartendaten von Google Maps: Routenplaner, © 2015 Google. Website, 2015. <https://www.google.de/maps/dir///@45.5601451,-73.7120832,11z> [11. Januar 2015].
- [KG14] Wiener Linien GmbH & Co KG: Fahrgastinfo - Service - Fahrpläne - U-Bahn. Website, Dezember 2014. <http://www.wienerlinien.at/eportal2/ep/channelView.do/pageTypeId/66526/channelId/-47208> [05. Januar 2015].
- [NW11] Martin Nöllenburg and Alexander Wolff: Drawing and labeling high-quality metro maps by mixed-integer programming. *IEEE Transactions on Visualization and Computer Graphics*, 17:626–641, Mai 2011.
- [Sau10] Petra Sauer: Liniennetz der Straßenbahn in Würzburg. Website, September 2010. http://www.vvm-info.de/vvm/media/files/liniennetzplaene/13022_lnp_wuerzburg_straba_din_a4_hoch_10_09_30.pdf [2. Dezember 2014].
- [Ste12] Joshua Stevens: Alternative segmentation schemes for the design of traffic maps. Master’s thesis, Michigan State University, 2012. Studienfach Geographie.
- [Tra14] Sydney Trains: Timetables - Suburban Lines. Website, Juni 2014. <http://www.sydneytrains.info/timetables/#landingPoint> [14. Januar 2015].
- [Wol04] Christian Wolmar: *The Subterranean Railway: How the London Underground Was Built and How It Changed the City Forever*. Atlantic Books, London, 2004, ISBN 1-84354-023-1.

Abbildungsverzeichnis

| | | |
|------|--|----|
| 1.1. | Liniennetzplan der Würzburger Straßenbahn [Sau10]. | 3 |
| 1.2. | Eine frühe Isochronenkarte von 1881 [Gal81]. | 4 |
| 1.3. | Wärmebild des niederländischen Intercity Liniennetzes [Hav14]. | 5 |
| 2.1. | Ein ungerichteter Multigraph. Die Knoten sind mit ihrem Grad beschriftet. | 8 |
| 2.2. | Darstellungen unterschiedlicher Beziérkurven. | 10 |
| 2.3. | Ineinander übergehende Beziérkurven mit gleicher Tangente im Übergangspunkt. | 10 |
| 3.1. | Plan des Würzburger Straßenbahnliniennetzes mit Reisezeiten. | 13 |
| 3.2. | Würzburger Straßenbahnnetz mit Beziérkurven als Kanten. | 14 |
| 3.3. | Sich kreuzende Linien A und B, rechts mit eingefügtem Scheinknoten. . . | 15 |
| 3.4. | Darstellung der anziehenden und abstoßenden Kräfte von Fruchterman und Reingold. | 17 |
| 3.5. | Zwei ineinander übergehende Beziérkurven mit von den Knoten unterschiedlich weit entfernten Kontrollpunkten. | 18 |
| 3.6. | Auf Tangenten und Knoten wirkende Kräfte. | 19 |
| 4.1. | Liniennetzplan der Würzburger Straßenbahn mit Anpassung der Kantenlängen an die Fahrtzeiten. | 24 |
| 4.2. | Liniennetzplan der U-Bahn von Montreal mit Anpassung der Kantenlängen an die Fahrtzeiten. | 24 |
| 4.3. | Liniennetzplan der U-Bahn von Wien mit Anpassung der Kantenlängen an die Fahrtzeiten. | 25 |
| 4.4. | Liniennetzplan der Vorortslinien von Sydney mit Anpassung der Kantenlängen an die Fahrtzeiten. | 26 |
| 4.5. | Einfluss des Vorfaktors von F_{ZEIT} auf die Zeichnung des Straßenbahnnetzes von Würzburg. | 28 |
| 4.6. | Einfluss des Vorfaktors von F_{ZEIT} auf die Zeichnung des U-Bahn Netzes der Stadt Wien. | 29 |
| 5.1. | Veranschaulichung einer Kraft, welche Knoten von Kanten abstößt. . . . | 30 |
| 5.2. | Verlängerung einer Kurve durch zusätzliche Krümmung. | 31 |

A. Inhalt der beigelegten CD

CDInhalt.txt

 /Arbeit:
NetzwerkeKantenlaenge.pdf

 /Input:
ipestylecrop.txt
montreal.graphml
sydney.graphml
wien.graphml
wuerzburg.graphml

 /Java:

 ./src:
metrocurves

 ./src/metrocurves:
AbstractForcedDirectedAlgorithm.java
Color.java
ForcedDirectedAlgorithm.java
GraphMLReader.java
IPEWriter.java
MyBezierCurve.java
MyEdge.java
MyHalfVirtualVertex1.java
MyVector.java
MyVertex.java
MyVirtualVertex.java
Point.java
Test.java

 /Quellen:
openstreetmap.html
spiegel-verfahren.html
wienerlinien-fahrplan.html
buchin_traveltimes.pdf
fink_mapdrawing.pdf

fruchterman_graphdrawing.pdf
galton_isochrones.pdf
google_maps.pdf
haunert_focusregions.pdf
haverkort_traveltime.pdf
sauer_strabawue.pdf
statis_fahrgaeste.pdf
stevens_trafficmaps.pdf
sydney_T1_north_shore_northern.pdf
sydney_T1_northern.pdf
sydney_T1_western.pdf
sydney_T2_airport.pdf
sydney_T2_inner_west_south.pdf
sydney_T3_bankstown.pdf
sydney_T4_eastern_suburbs_illawarra.pdf
sydney_T5_cumberland.pdf
sydney_T6_carlingford.pdf
sydney_T7_olympic_park.pdf
vvm_fahrplan.pdf
wienerlinien_linieU1.pdf
wienerlinien_linieU2.pdf
wienerlinien_linieU3.pdf
wienerlinien_linieU4.pdf
wienerlinien_linieU5.pdf
wolff_drawingandlabelling.pdf

B. Erklärung

Hiermit versichere ich die vorliegende Abschlussarbeit selbstständig verfasst zu haben, keine anderen als die angegebenen Quellen und Hilfsmittel benutzt zu haben, und die Arbeit bisher oder gleichzeitig keiner anderen Prüfungsbehörde unter Erlangung eines akademischen Grades vorgelegt zu haben.

Würzburg, den 21. Januar 2015

.....
(Matthias Neumann)