Optimale Routenplanung bei der Müllabfuhr

Von Minh Duc Bui

Gliederung

1. Einleitung und Motivation
2. Beschreibung des Modells
3. Lösungsansatz
4. Numerische Umsetzung und Implementierung
5. Ausblick

1. Einleitung und Motivation

1. Einleitung und Motivation

- Kosten minimieren, indem wir die schnellstmögliche Route fahren
- Mannheim: 108.269 Tonnen Müll (Biomüll, Grünabfälle und Papier) aus privaten Haushälter und Geschäften im Jahr 2017 entsorgt -> 355 kg/Einwohner und Jahr oder auch ca. 500 Tonnen täglich
- Am Ende werden wir ein Stadtteil von Mannheim betrachten

2. Beschreibung des Modells

2.1 Mathematische Beschreibung

2. Beschreibung des Modells

- **Kriterien der Route:**
  1. Alle Straßen (mindestens) einmal befahren
  2. Einbahnstraßen und Sackgassen beachten
  3. Von der Müllhalde und wieder zurück

- In dieser Arbeit: Betrachten nur Straßennetze **ohne** Einbahnstraßen

2.1 Mathematische Beschreibung

- Straßennetz einer Stadt bzw. eines Stadtteiles als ein ungerichteten Graph modellieren
- Kante in einem ungerichtetem Graphen verlaufen immer nur in beiden Richtungen
- Graph soll zusammenhängend sein
- Knoten als Kreuzungen oder das Ende einer Sackgasse
- Zuweisungen von Kantengewichten: Verschiedene Arten von Variablen

-> Modellierung durch zusammenhängenden gewichteten ungerichteten Graphen

Definition (Kantengewicht). *Wir nennen eine Funktion* $c\colon E \to \mathbb{R}$ Kantengewichte *oder* Kostenfunktion *und schreiben*

$$c(G) = c(E(G)) = \sum_{e \in E(G)} c(e)$$

für das Gewicht des Graphen G*, wobei für unsere Zwecke* $c\colon E \to \mathbb{R}_{>0}$ *ausreicht.*


84
73
77
170
200
220
230
79
82
65

Unser Problem

- Finde einen Weg, der die folgenden Bedingungen erfüllt:

1. Jede Kante mindestens einmal durchläuft
2. Startpunkt identisch mit Endpunkt
3. Summe der Gewichte von den Kanten minimieren

Wichtige Definitionen

- **Eulerweg**: Weg, der jede Kante in zusammenhängendem Graphen genau einmal durchläuft
- **Eulertour:** Eulerweg muss zusätzlich ein Kreis sein

Wichtige Definitionen

Definition (Euler-Graph). *Ein Graph heißt* eulersch *oder ist ein* Euler-Graph, *wenn es in ihm eine* Euler-Tour *gibt. Ein* Euler-Kreis *oder eine* Euler-Tour *in einem Graphen ist ein Kreis, der jede Kante genau einmal enthält.*

3. Lösungsansatz

3.1 Konstruktion von Eulergraphen und –touren
3.2 Algorithmen zur Bestimmung von Eulertouren in Eulergraphen
3.3 Verbinden von ungeraden Knoten

3.1 Konstruktion von Eulergraphen und -touren

Definition (Grad). *Sei $G = (V,E,\gamma)$ ein Graph. Der* Grad $d(v)$ *eines Knoten* $v \in V$ ist die Anzahl des Auftretens von v als Endknoten einer Kante.

Satz. *Ein zusammenhängender Graph ist genau dann eulersch, wenn jeder Knoten einen geraden Grad hat.*

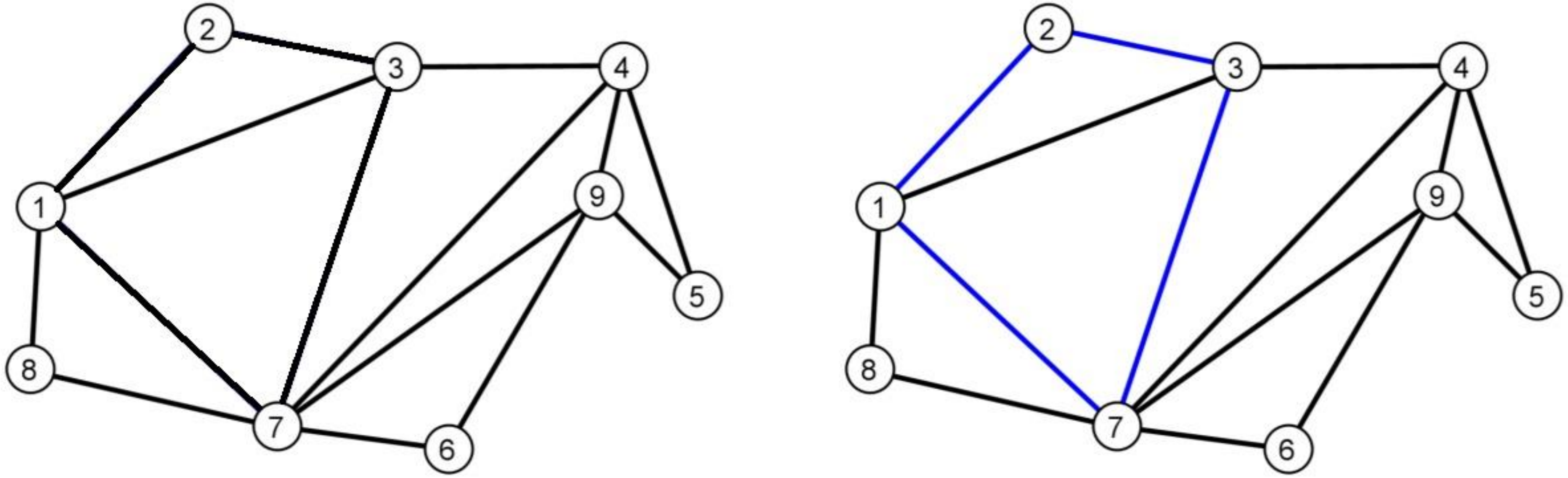
3.2 Algorithmen zur Bestimmung von Eulertouren in Eulergraphen

- Zwei verschiedene Algorithmen
- Algorithmus von Hierholzer
- Algorithmus von Fleury
- Satz hilft uns Korrektheit der Algorithmen zu beweisen
- Annahme: Graph G ist ein eulerscher Graph

Algorithmus von Hierholzer

1. Schritt:

- Wähle einen beliebigen Startknote $v_0 \in V$
- Wähle unbesuchte Kanten bis der Ausgangsknoten wieder erreicht ist. Der konstruierte Weg ist ein Kreis K.

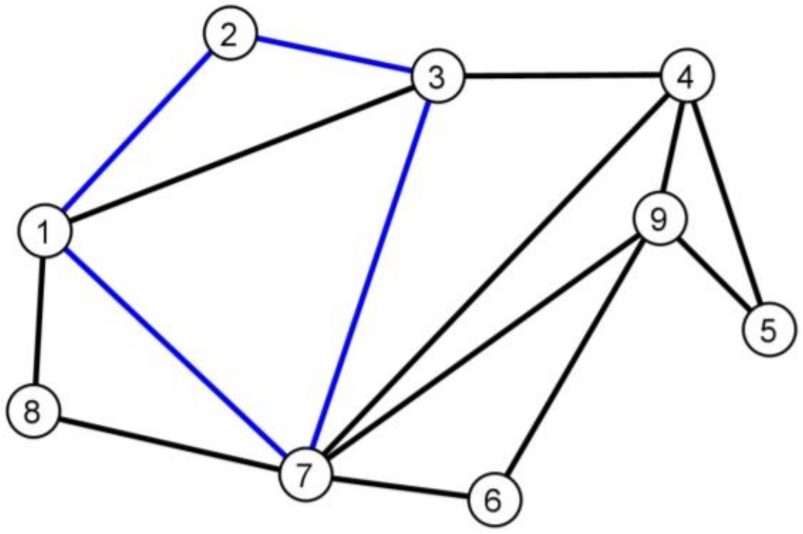


Knoten 2 als Startknoten

Algorithmus von Hierholzer

2. Schritt:

- Wenn K eine Eulertour ist, breche ab und K ist unsere Lösung. Ansonsten gehe zu Schritt 3.



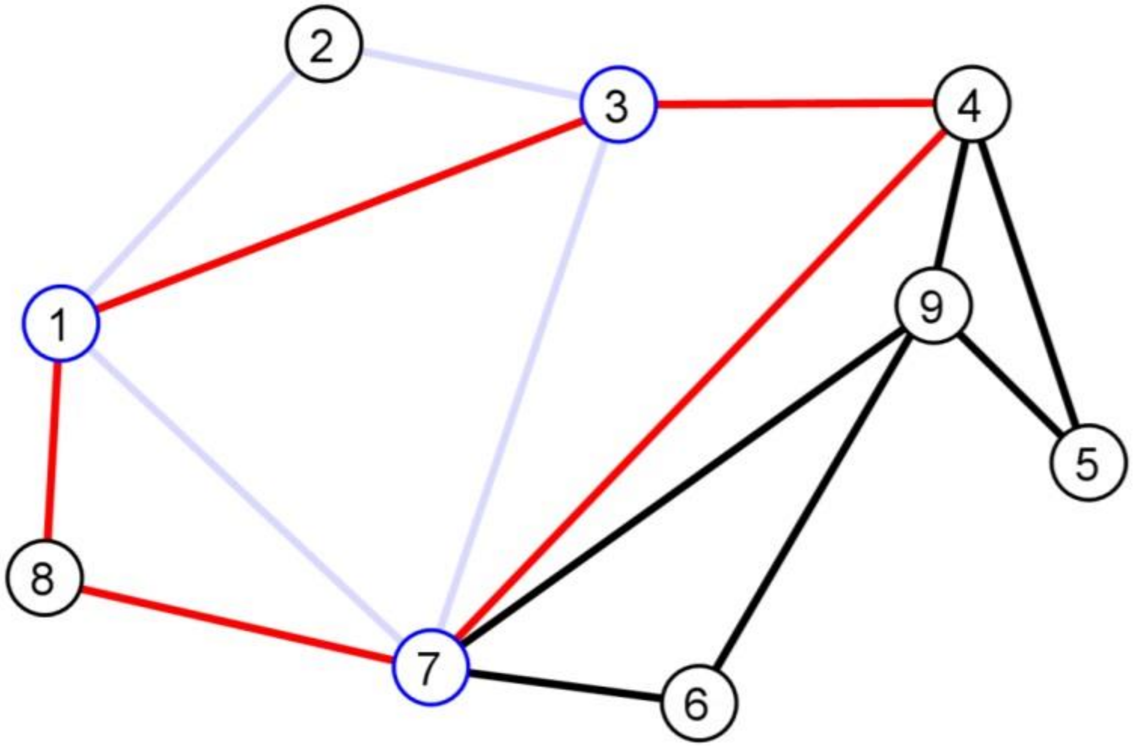
Knoten 2 als Startknoten

Algorithmus von Hierholzer

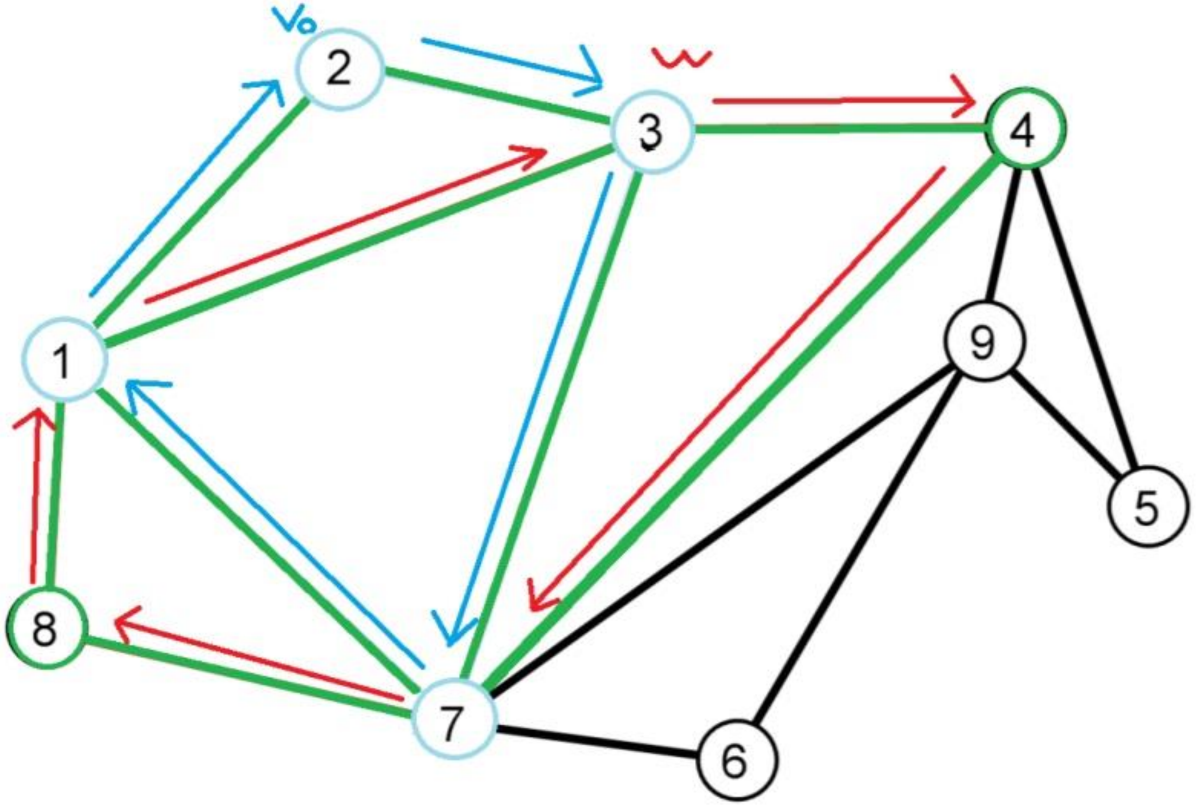
3. Schritt:

- Setze $K' = K$.
- Wähle Knoten w aus dem Kreis K', der eine unbesuchte Kante besitzt.
- Konstruiere wie im 1. Schritt für w einen Kreis K'', wobei wir nur den Graphen ohne Kanten vom Kreis K' betrachten.
- Nun wollen wir den Kreis K neu konstruieren, indem er die Kreise K' und K'' enthält: Gehe von $v_0 \in K'$ entlang K' bis zum Knoten w. Durchlaufe dann den neuen Kreis K'' einmal komplett durch und wieder bei w angekommen, durchläuft man den Rest von K'. Bezeichne diesen Weg als K.
- Gehe zu Schritt 2.

Algorithmus von Hierholzer



Algorithmus von Hierholzer



Satz. *Gegeben sei ein zusammenhängender Graph, in dem alle Kanten geraden Grad haben. Dann berechnet der Algorithmus von Hierholzer eine Euler-Tour.*

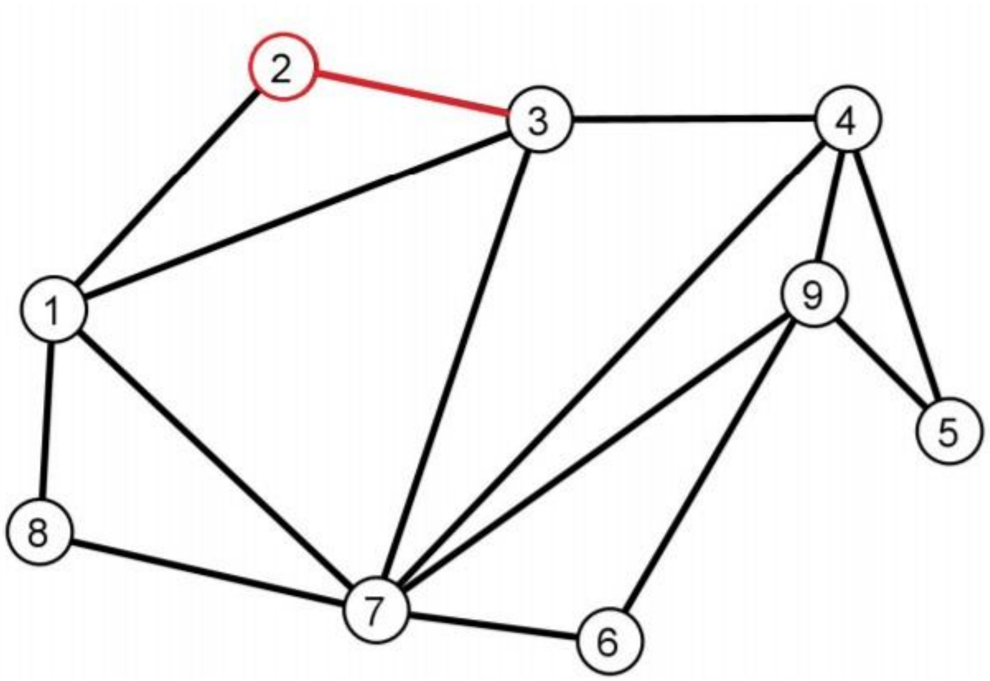
Algorithmus von Fleury

Definition (Brücke). Eine Brücke ist eine Kante $e \in E$ bei deren Löschung der Graph in zwei Komponenten zerfallen würde.

Lemma. *Wenn in einem Graphen alle Knotengrade gerade sind, dann hat er keine Brücke.*

Algorithmus von Fleury

1. Schritt: Beginne mit einer beliebigen Startknoten und wähle eine inzidente Kante.



Algorithmus von Fleury

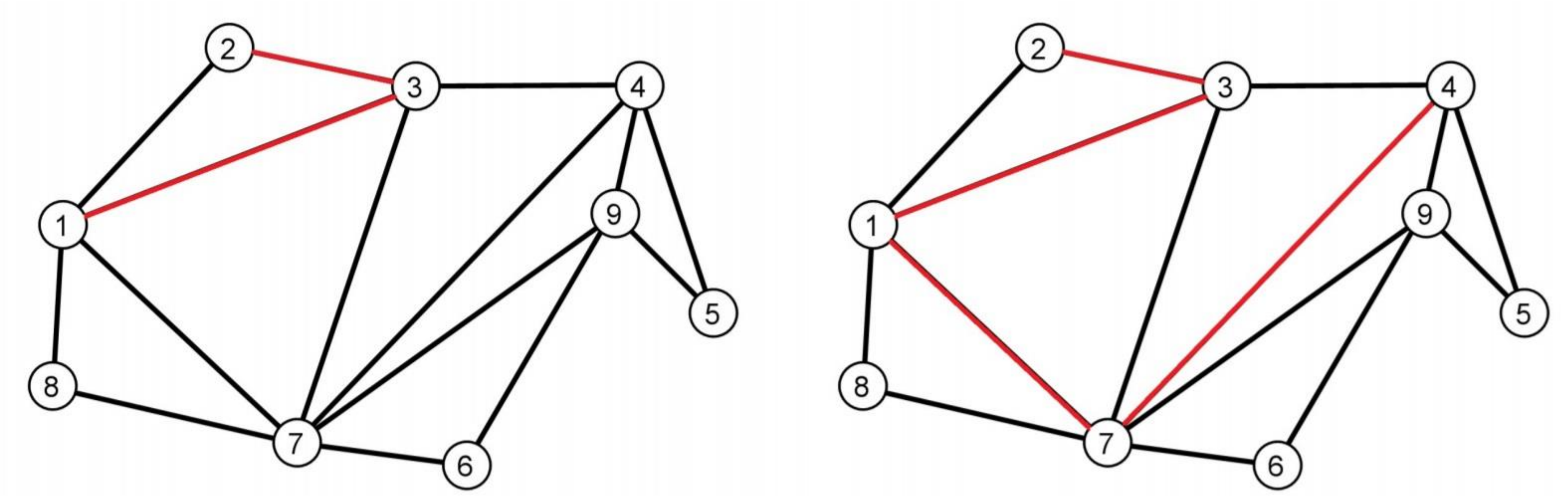
2. Schritt: Wähle nächste unbesuchte Kante aus. Dabei sind Kanten zu wählen, die die 2 Bedingungen erfüllen, wobei die 2. Bedingung verletzt werden darf, wenn es keine andere Kanten gibt, die diese erfüllt:

1. Inzident zu der zuletzt besuchten Kanten.

2. Kante (u, v) ist keine Brücke in dem Restgraphen, der aus allen noch nicht besuchten Kanten besteht.
Wir überprüfen diese Bedingung, indem wir nach einem (u, v)-Weg suchen im Restgraphen ohne die Kante (u, v).
Falls wir solch ein Weg finden können, z.B. mit der Tiefensuche, dann wissen wir, dass die Kante eine Brücke sein muss, denn wir können den gefundenen Weg (u, v) erweitern zu einem Kreis mit der Kante u, v.
Aber eine Kante ist keine Brücke, wenn sie auf einem Kreis liegt!

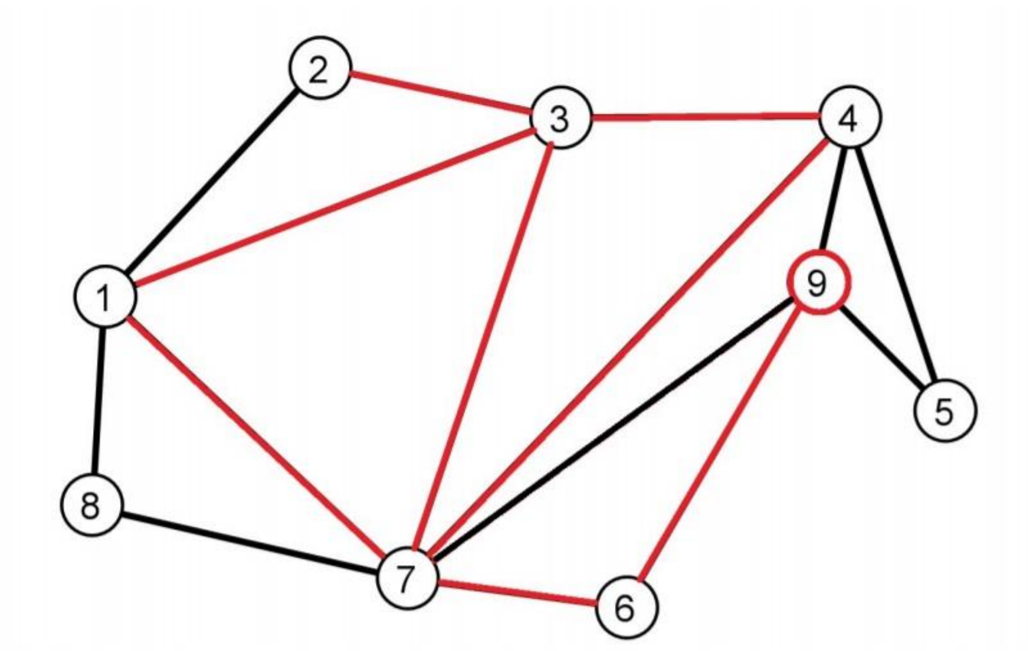
3. Schritt: Wiederhole Schritt 2 bis alle Kanten besucht worden sind.

Algorithmus von Fleury

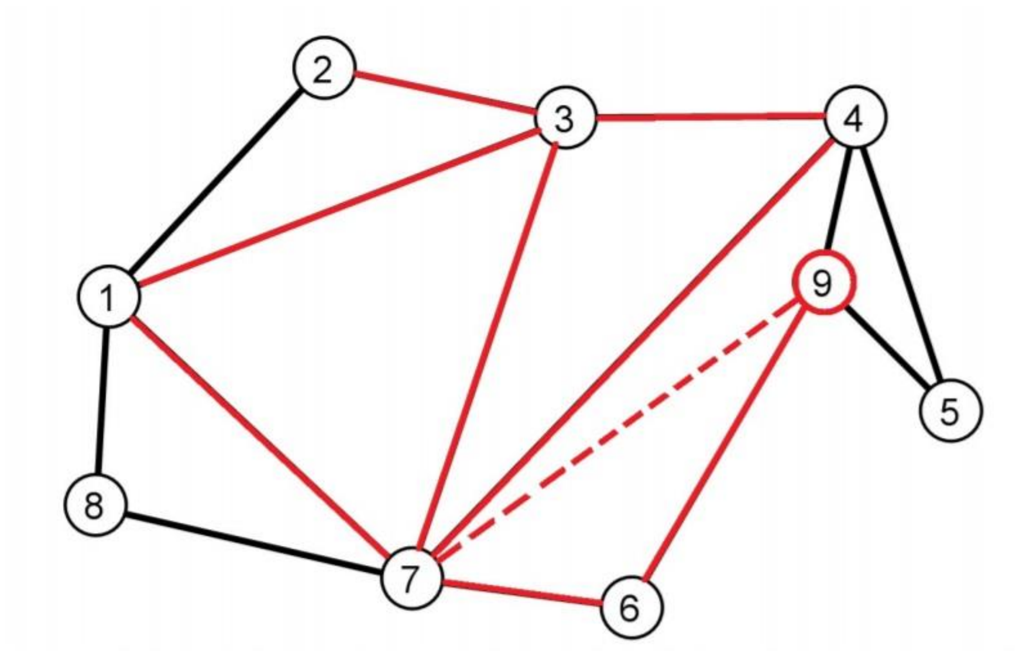


(a) Graph nachdem zwei Kanten gewählt wurden (b) Graph nachdem vier Kanten gewählt wurden

Algorithmus von Fleury

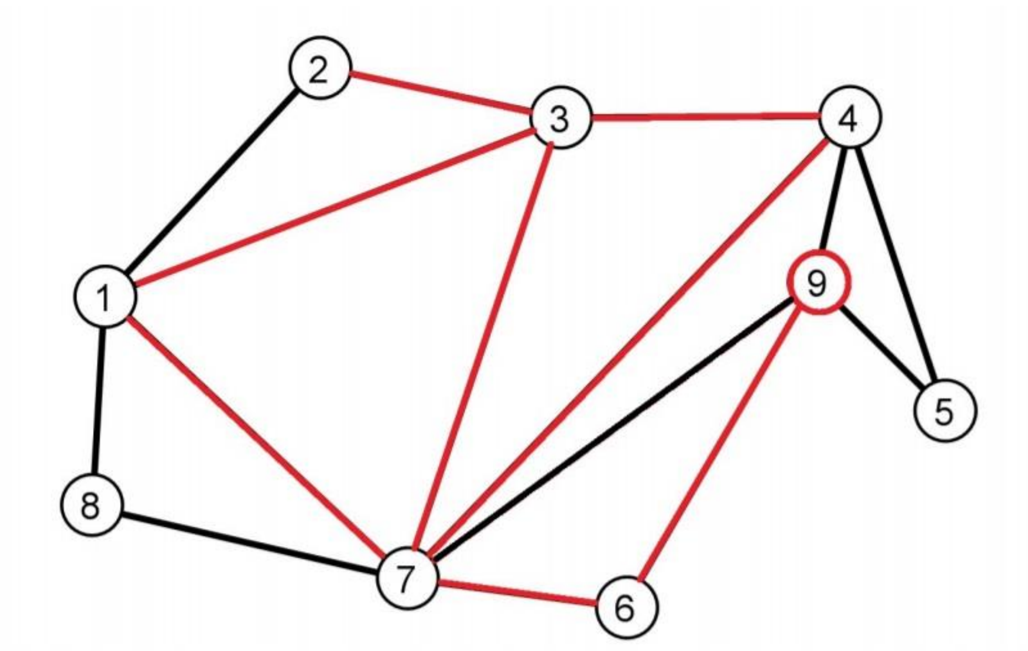


(c) Graph nachdem acht Kanten gewählt wurden. Wir befinden uns im Knoten 9.

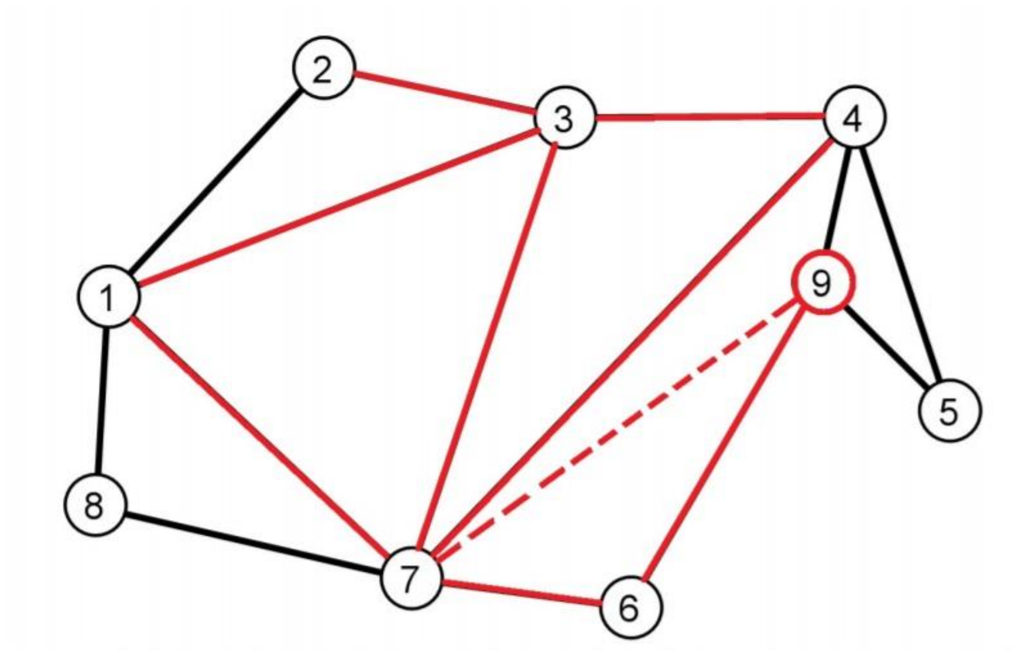


(d) Wahl der gestrichelten Linie verboten, da sonst der Restgraph in zwei unzusammenhängende Graphen zerfällt

Algorithmus von Fleury



(c) Graph nachdem acht Kanten gewählt wurden. Wir befinden uns im Knoten 9.



(d) Wahl der gestrichelten Linie verboten, da sonst der Restgraph in zwei unzusammenhängende Graphen zerfällt

3.3 Verbinden von ungeraden Knoten

Satz. In jedem Graphen ist die Anzahl der Knoten mit ungeradem Grad gerade.

- **Idee:** Knoten mit ungeraden Grad paarweise zu verbinden
- Wollen aber die Summe der Kantengewichte minimal halten
- Verbindung soll also minimal sein
- Verbindung durch Knoten mit geraden Grad erlaubt
- Anmerkung: Wir bleiben optimal!

3.3 Verbinden von ungeraden Knoten

Satz. In jedem Graphen ist die Anzahl der Knoten mit ungeradem Grad gerade.

- **Idee:** Knoten mit ungeraden Grad paarweise zu verbinden
- Wollen aber die Summe der Kantengewichte minimal halten
- Verbindung soll also minimal sein
- **Lösung:** Minimales Matching
- Verbindung durch Knoten mit geraden Grad erlaubt
- Anmerkung: Wir bleiben optimal!

Lösung: Finden einer optimaler Route bei der Mullabfuhr

1. Wandel das Straßennetz in einen ungerichteten zusammenhängenden Graphen um, wobei Kreuzungen die Knoten und Straßen die Kanten darstellen sollen und Sackgassen sollen erstmal ignoriert werden.

2. Falls der Graph Knoten mit ungeraden Grad hat, dann betrachte diese Knoten und finde eine Verbindung mit der kleinsten Summe der Kantengewichte, um unsere Route optimal zu halten.

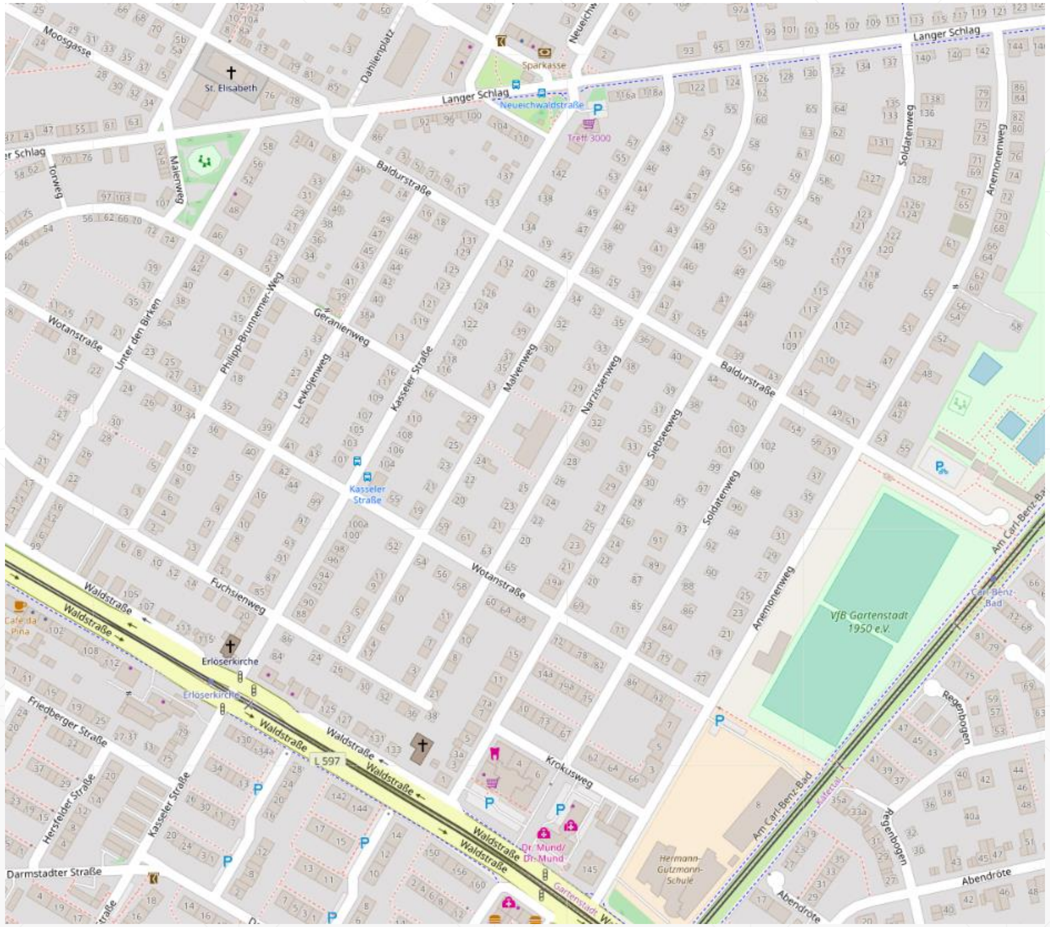
3. Fuge die Verbindung in Form von Kantenzügen in den Graphen hinzu

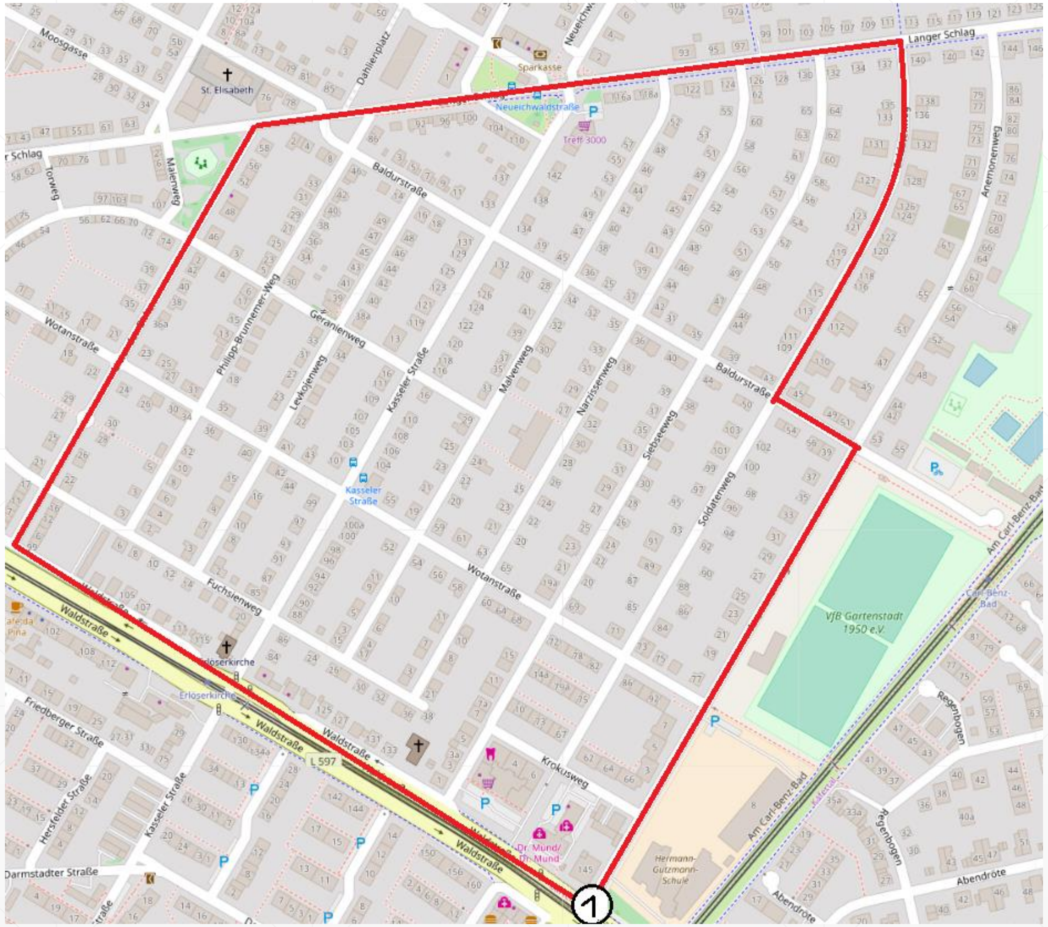
4. Bei Sackgassen müssen wir hinein- und hinausfahren. Also werden für Sackgassen doppelte Kanten eingezeichnet.

5. Wir haben nun einen Eulergraph und jede Eulertour ist optimal. Wir k¨onnen unsere Algorithmen von Hierholzer und Fleury anwenden um eine Eulertour zu finden. Nun haben wir eine optimale Route gefunden und wir sind fertig!

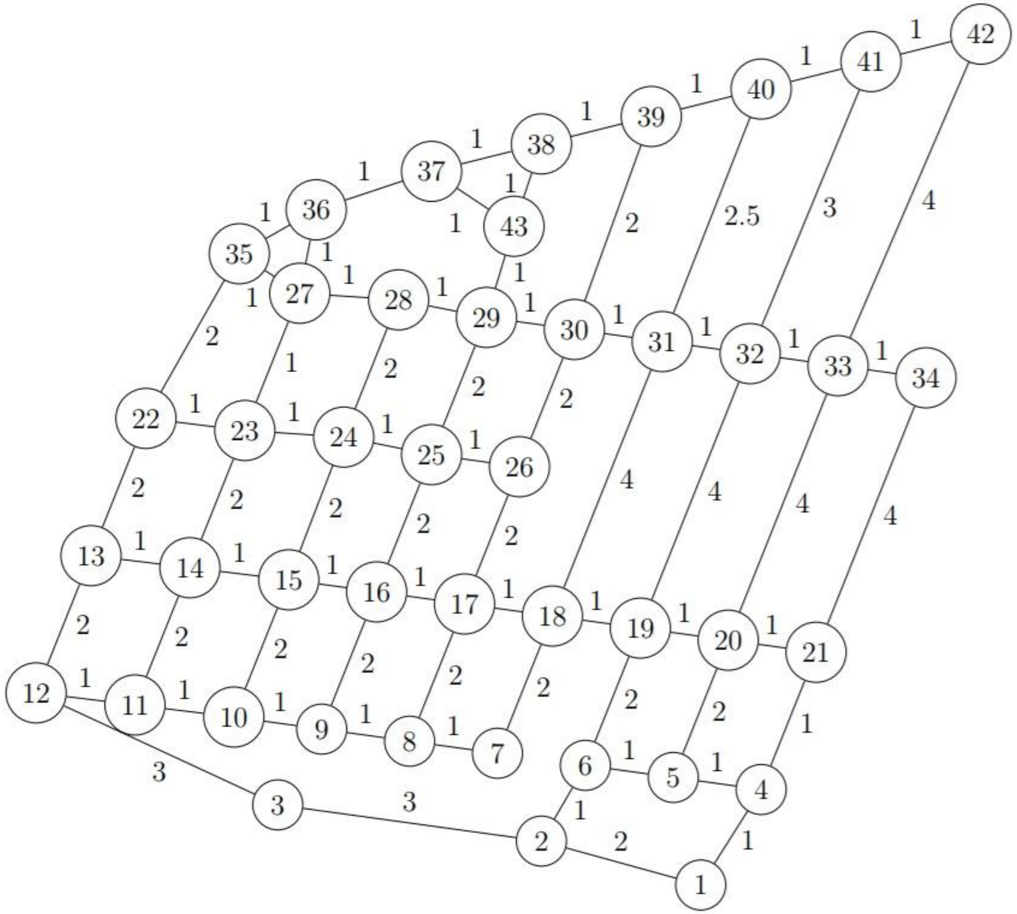
4. Numerische Umsetzung und Implementierung

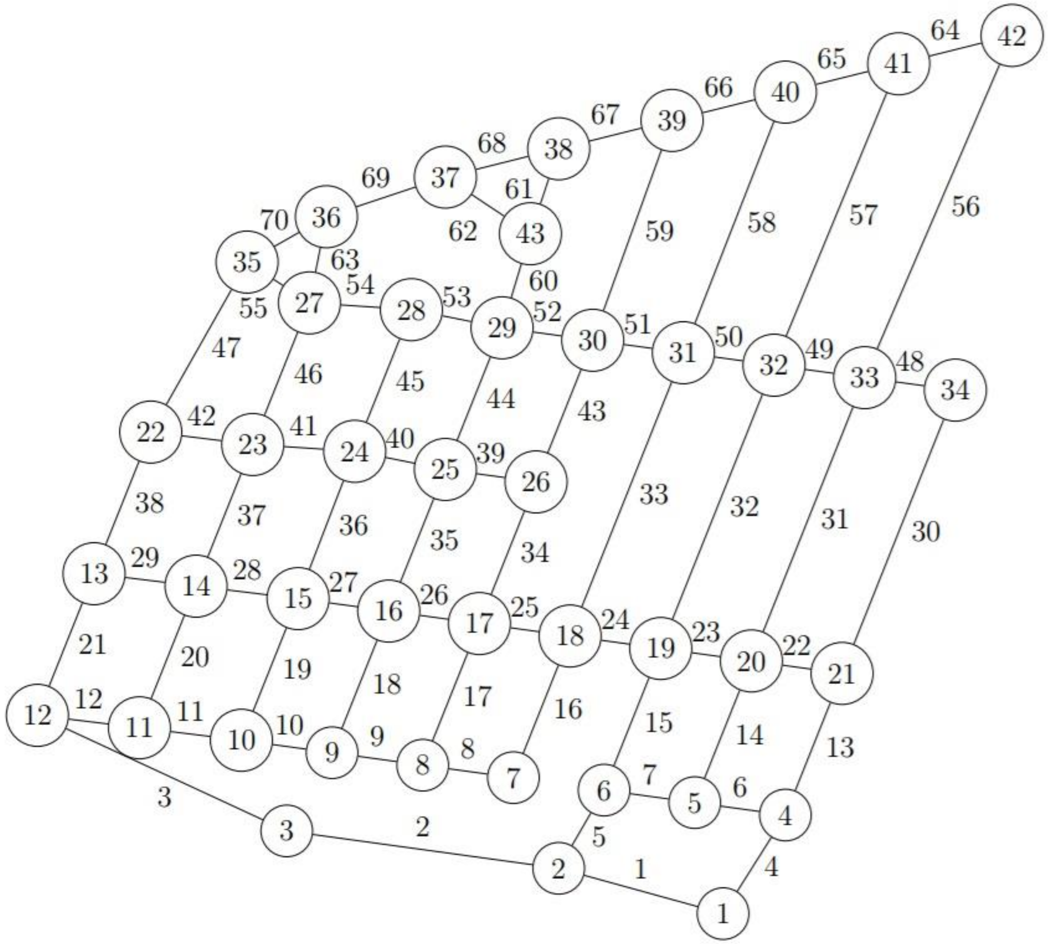
Stadtteil „Gartenstadt“



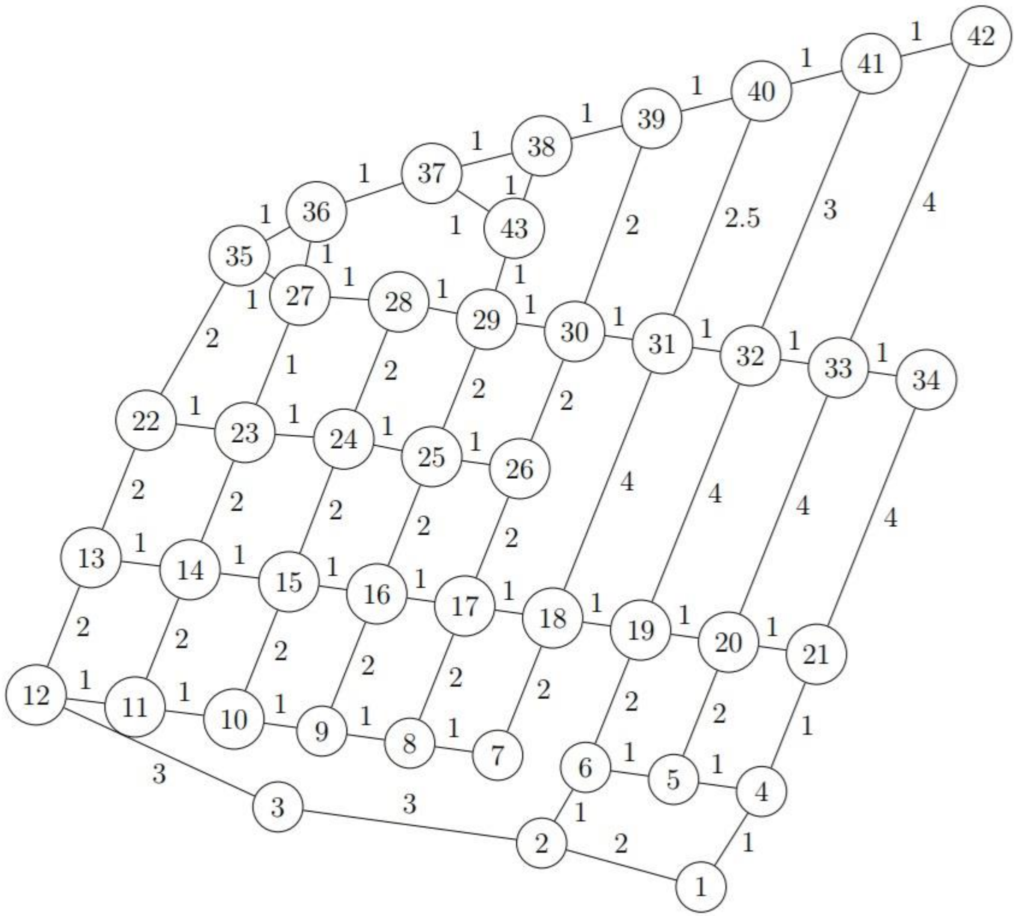


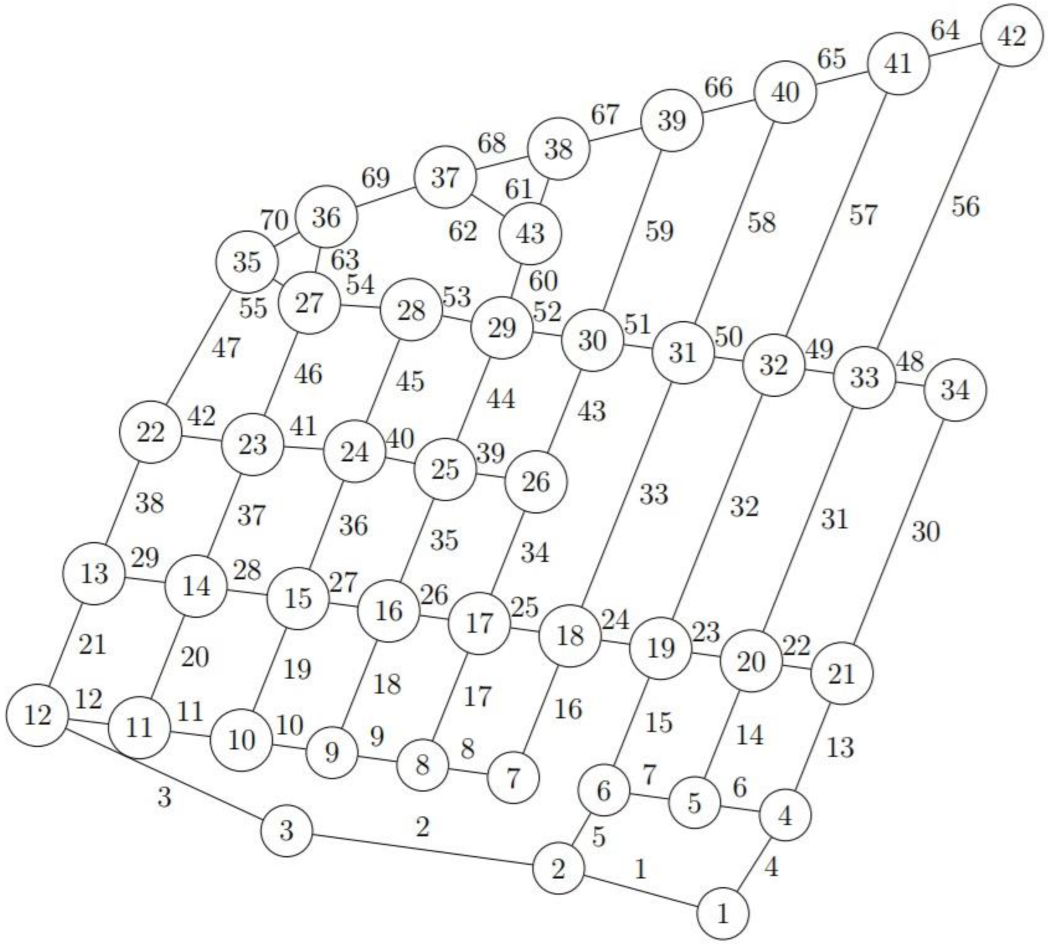
1. Schritt: Straßennetz umwandeln



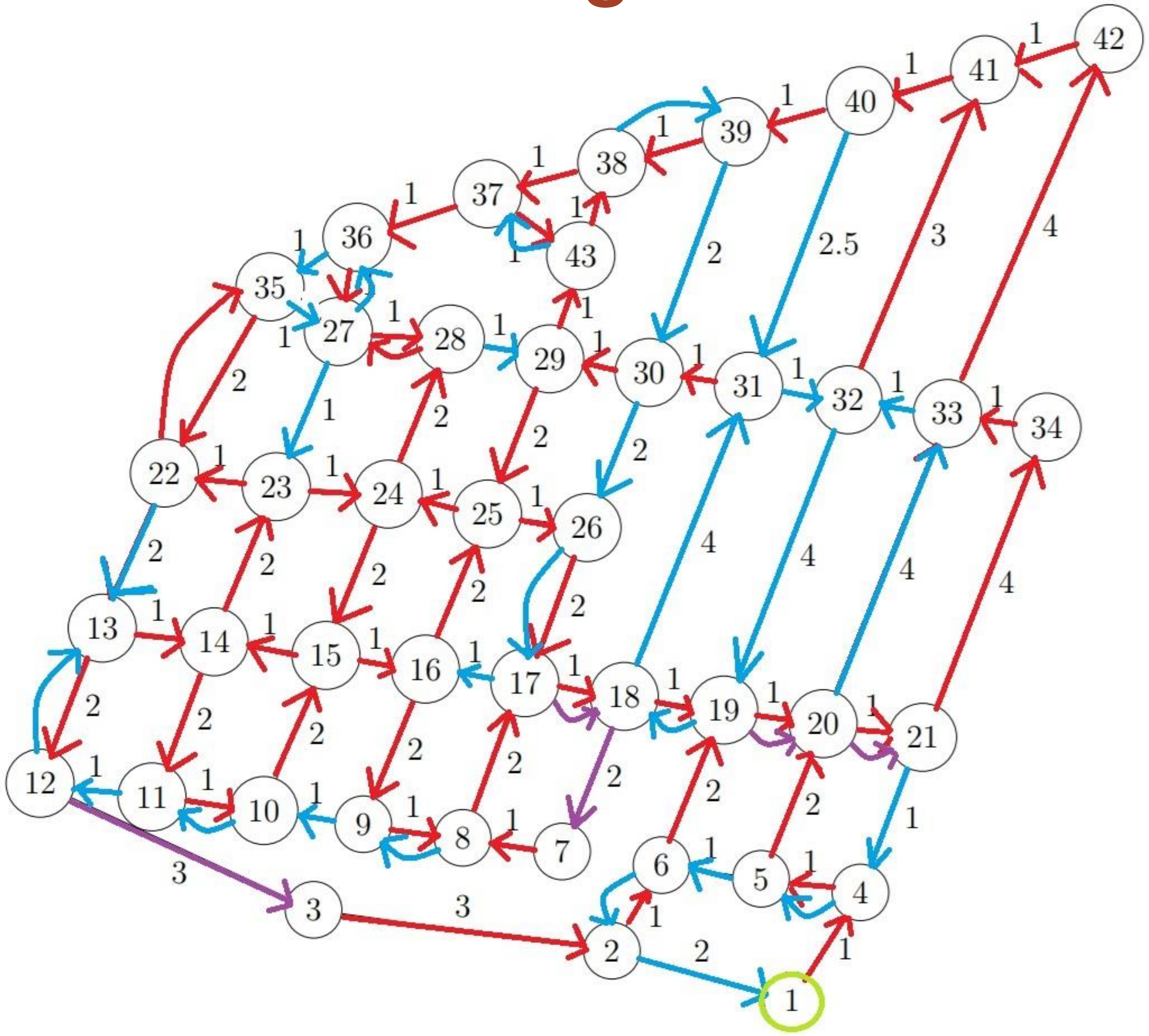


2. Schritt: Minimales Matching





Endergebnis



5. Ausblick

Ausblick

- Einbahnstraßen ins Modell mit einbringen
- Straßen, die man zweimal befahren muss mit einbringen
- Programm, die ein Straßennetz in eine Inzidenzmatrix umwandelt
- Optimierung des Codes