

Projekt 2: Fiber-To-The-x

Moritz Hefner und Lucia Ortjohann

12. Dezember 2018

Inhaltsverzeichnis

1	Problemstellung	1
2	Vorüberlegung	2
3	Point-to-Point	2
3.1	Point-to-Point mit Glasfaser	2
3.2	Point-to-Point mit Glasfaser und Kupfer	3
3.3	Erweiterung des P2PGK	6
4	Point-to-multipoint communication	6
4.1	Point-to-multipoint mit Glasfaser	7
4.2	Point-to-multipoint mit Glasfaser und Kupfer	7
4.3	Erweiterung des P2MPGK	7
	Anhang	9

1 Problemstellung

TODO hübschere Einleitung + HÄSSLICH!!!

In diesem Projekt wollen wir Glasfaserkabel und Kupferkabel in einem vorhandenen Straßennetz verlegen, sodass jeder Kunde an eine ausgewählte Leitstelle angebunden ist. Dabei muss die Nachfrage von jedem Kunden gedeckt sein und die Lösung soll möglichst kostengünstig sein. Dafür gibt es verschiedene Problemstellungen. Beim Point-to-Point Problem, soll jeder Kunde (bis auf die letzte Meile) eine eigene Glasfaserleitung bekommen. Beim Point-to-Multipoint Problem können sogenannte Splitter installiert werden und Glasfaserkabel zusammengefasst werden.

Um diese Probleme zu lösen sind folgende Daten gegeben. Gegeben ist ein (gerichteter) Graph $G = (V, A)$. Die Menge der Knoten besteht aus 4 disjunkten Mengen $L, K, F, S \subseteq V$, wobei L die Menge der auszuwählenden Leitstellen ist. K ist die Menge der zu versorgenden Kunden. Die Menge F ist eine Menge von so genannten Facility Knoten. In einer Facility können Kunden mit Glasfaser oder Kupfer angeschlossen werden. Die Menge $F_1 \subseteq F$ ist die Menge der Facility Knoten von Typ 1. Hier können Kunden über Glasfaser angeschlossen werden. Die Facility Knoten, von denen die Kunden mit Kupfer angeschlossen werden, sind in der Menge F_2 . Die Mengen F_1 und F_2 sind disjunkt. S ist die Menge der Knoten, die nur für die Verbindung genutzt werden (Steinerknoten).

Die Menge der Kanten V besteht aus den inneren Kanten I und den Anschlusskanten, wobei unterschieden wird, ob diese Verbindung mit Kupfer A_2 oder Glasfaser A_1 möglich ist. Hier gilt, dass A_1 und A_2 disjunkt sind. Eine Anschlusskante geht immer von einem Knoten der Menge F zu einem Kunden $k \in K$. Der Graph bildet das Grundgerüst für unsere Probleme. Zusätzlich gibt es noch einige Bedingungen. Grundsätzlich wollen wir die Kosten des Netzwerks minimieren. Dazu ist die Kostenfunktion $c : V \cup A \rightarrow \mathbb{Q}$ gegeben. Für die Verlegung von Glasfaser auf den inneren Kanten I fallen Kosten in Höhe von $c(e)$ an für alle $e \in I$. Für die Verlegung von Kupfer auf den Anschlusskanten von Typ 2 fallen auch Kosten in Höhe von $c(e)$ an für alle $e \in A_2$. Die Anschlusskanten von Typ 1 haben die Länge 0 und somit keine Kosten. Der Aufbau einer Leitstelle $l \in L$ kostet $c(l)$. In den Facility Knoten kann man entweder einen DSL-Zugangsmultiplexer (DSLAM) installieren (Facility Knoten von Typ 2) oder einen Kunden an Glasfaser anschließen (Facility Knoten von Typ 1). Dafür entstehen Kosten in Höhe von $c(f)$ für alle $f \in F$. Zusätzlich hat jeder Kunde noch eine Nachfrage an Bandbreite von $d(k)$ für alle $k \in K$ und für den Anschluss eines Kunden kann mit Profit von $p_1(k)$ für Anschluss mit Glasfaser und $p_2(k)$ für den Anschluss mit Kupfer gerechnet werden. Außerdem entstehen bei P2MP noch Kosten für die Installation eines Splitters von $c_s(i)$.

Die Ergebnisse der einzelnen Problem auf drei vorgegeben Instanzen werden jeweils am Ende des Abschnittes beschrieben. Um die Laufzeiten der

Algorithmen besser vergleichen zu können, geben wir nun die Technischen Daten der genutzten Computer an:

Computer	Prozessor	Installierter RAM
PC 1	Intel(R) Core(TM) i7-4500 CPU @ 1.80GHz x 4	8,00 GB
PC 2	Intel(R) Core(TM) i5-2430 CPU @ 2.40GHz	8,00 GB
PC 3	4 x Intel(R) Xeon(R) CPU @ 2.93GHz	12,30 GB

Tabelle 1: Technische Daten

2 Vorüberlegung

TODO Einleitung (Noch mehr Preprocessing??)

Um jeden Kunden angemessen zu versorgen, muss die Nachfrage jedes Kunden gedeckt sein. Diese Bedingung kann man direkt mit Preprocessing lösen.

Da die Kapazität der Glasfaserleitungen in unserem Modell unendlich ist, wird die Nachfrage eines Kunden auf jeden Fall gedeckt, wenn dieser mit Glasfaser angebunden ist. Außerdem wird auf den inneren Kanten auch nur Glasfaser verlegt. Das heißt, die einzigen Kanten, die dafür sorgen könnten, dass die Nachfrage eines Kunden nicht gedeckt ist, sind die Anschlusskanten von Typ 2. Die Anschlusskanten von Typ 2, gehen von einer Facility zu einem Kunden. Falls diese Kante nicht die Kapazität hat, die die Nachfrage des Kunden deckt, können wir die Kante in unserem Netzwerk nicht benutzen. Deswegen löschen wir, bevor wir die verschiedenen Probleme lösen, diese Kanten aus dem oben genannten Graphen. Das heißt, jedes Netzwerk in dem neuen Graphen, in dem alle diese Kanten gelöscht sind, erfüllt somit die Nachfrage der Kunden. Diesen neuen Graphen bezeichnen wir im folgenden wieder mit $G = (V, A)$ und lösen die folgenden Probleme auf diesem Graphen. Da die gelöschten Kanten in keiner der Lösungen benutzt werden könnten, verändert dies die Lösungen nicht.

3 Point-to-Point

TODO P2P beschreiben?

3.1 Point-to-Point mit Glasfaser

Beim P2P mit Glasfaser Problem (P2PG) muss jeder Kunde mit einer eigenen Glasfaserleitung angeschlossen werden. Das heißt, unsere Problem besteht darin, eine Leitstelle aus L auszusuchen und dann ein Glasfaserkabel von dieser Leitstelle zu einem Facility Knoten von Typ 1 zu verlegen, um dann mit der Anschlusskante von der Facility zum Kunden den Kunden

anzubinden. Dabei sollen die Kosten minimiert werden. Es gibt genau eine Anschlusskante und eine Facility von der ein Kunde mit Glasfaser angebunden werden kann und die Länge dieser Kante ist 0. Damit kann man das Problem einen Kunden anzuschließen, auch als kürzestes Wege Problem von einer Leitstelle zu dem Kunden gesehen werden. Wir lösen also für jede Leitstelle in L einmal das kürzeste Wege Problem zu jeder Facility von Typ 1.

Dabei gehen wir wie folgt vor: Für jede Leitstelle $l \in L$ berechnen wir auf dem Hilfsgraphen $H = (\{l\} \cup S \cup F, I, c|_I)$ einen kostenminimalen Weg von der Leitstelle l zu jeder Facility $f \in F_1$. Dazu führen wir für den Dijkstra-Algorithmus einmal mit Startknoten l durch. Der errechnete Weg sei nun $W_{l,f}$ und die Kosten dieses Weges seien $c_{\text{dij}}(\{l, f\})$. Jeder Kunde besitzt nur eine eingehende Anschlusskante von Typ 1, diese hat Verlegungskosten 0. Außerdem besitzt jeder Kunde genau eine Facility aus F_1 von der dieser Kunde mit Glasfaser angeschlossen werden kann. Andersherum besitzt jede Facility aus F_1 nur einen Kunden den dieses anschließen muss. Dh. die Kosten der optimalen Lösung für die ausgewählte Leitstelle l nun $\text{Kosten}_l := \sum_{f \in F_1} c_{\text{dij}}(\{l, f\}) + c(f)$. Also die Kosten die Wege von der Leitstelle l zur jeder Facility aus F_1 plus die Kosten für die Anschlüsse aller Kunden an die Facility aus F_1 . Nun suchen wir die Leitstelle $i \in L$ mit den geringsten Kosten ($i := \arg \min_{l \in L} \text{Kosten}_l$). Das zugehörige Netzwerk stzt sich aus den errechneten kürzesten Wegen $W_{i,f}$ für alle $f \in F_1$ zusammen und allen Anschlusskanten von Typ 2, also ist das Netzwerk die Menge $(\dot{\bigcup}_{f \in F_1} W_{i,f}) \cup A_2$. Die disjunkte Vereinigung bedeutet, dass falls zwei Wege über dieselbe Kante laufen, wir in unserem Netzwerk auch zwei Glasfaserkabel über diese Kante verlegen.

TODOErgebnisse + Bilder als Übersicht:

Instanz	Naunyn	Berlin	Vehlefanz
Kosten	TODO	TODO	TODO
Laufzeit auf PC1	TODO	TODO	TODO

Tabelle 2: Ergebnisse des P2PG

3.2 Point-to-Point mit Glasfaser und Kupfer

Das Point-to-Point mit Glasfaser und Kupfer Problem (P2PGK) ist wie oben schon erklärt eine Erweiterung des P2PG bei dem jeder Kunde nun auf der

letzten Meile mit Kupfer angeschlossen werden kann und diese Kupferkabel können in einem Facility Knoten von Typ 2 zu einem Glasfaserkabel zusammengefasst werden. **TODO**genauere Problembeschreibung...

Für jede Leitstelle $l \in L$ führen wir den folgenden Algorithmus durch. Zuerst berechnen wir für alle Facilitys $f \in F$ den kostenminimalen Weg von der Leitstelle l zu der Facility f , genau wie beim P2PG Problem. Dazu führen wir einmal den Dijkstra-Algorithmus für kürzeste Wege auf dem Hilfsgraphen $H = (\{l\} \cup S \cup F, I, c|_I)$ mit Startknoten l aus. Der errechnete Weg sei nun $W_{l,f}$ und die Kosten dieses Weges seien $c_{\text{dij}}(\{l, f\})$. Für unsere Lösung heißt das, falls wir einen Kunden über die Facility f anschließen, verlegen wir das Glasfaserkabel von l nach f genau auf dem kostenminimalen Weg $W_{l,f}$. Jetzt müssen wir nur noch entscheiden, welchen Kunden wir an welche Facility anschließen. Dazu konstruieren wir einen weiteren Hilfsgraphen. Sei dazu $E = \{\{l, f\} \mid f \in F\}$ die Menge der Kanten von der ausgewählten Leitstelle zu jeder Facility. Diese Kanten $\{l, f\}$ ersetzen den errechneten kürzesten Weg $W_{l,f}$ von l nach f . Der Hilfsgraph $H'' = (V'', A'')$ besteht nun aus den Knoten $V'' = \{l\} \cup F \cup K$ und den Kanten E und den Anschlusskanten von Typ 1 und Typ 2 ($A'' = E \cup A_1 \cup A_2$), wie in Abbildung 1 dargestellt.

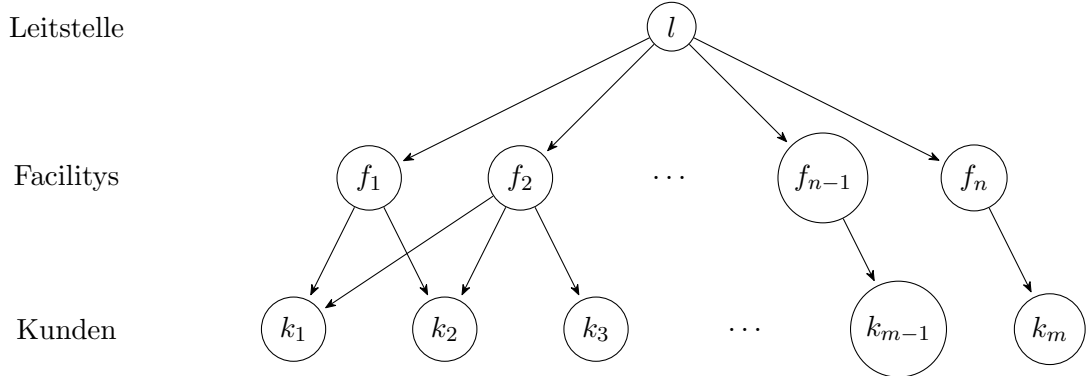


Abbildung 1: Hilfsgraph H''

Außerdem definieren wir eine Kostenfunktion c' auf den Kanten des Hilfsgraphen, wie folgt:

$$c' : A'' \rightarrow \mathbb{Q}, \{i, j\} \mapsto \begin{cases} c_{\text{dij}}(\{l, f\}), & \text{falls } \{i, j\} \in E \text{ und } j \in F_1 \\ c_{\text{dij}}(\{l, f\}) + c(j), & \text{falls } \{i, j\} \in E \text{ und } j \in F_2 \\ c(\{i, j\}) + c(i), & \text{falls } \{i, j\} \in A_1 \\ c(\{i, j\}), & \text{falls } \{i, j\} \in A_2 \end{cases}$$

Für die Kanten $\{l, f\}$ ergeben sich Kosten von $c_{\text{dij}}(l, f)$, für die Verlegung von Glasfaser auf dem Weg von l nach f für alle $f \in F$. Falls $f \in F_2$ eine Facility von Typ 2 ist, addieren wir noch Kosten von $c(f)$ auf die Kante, da falls wir diese Kante später benutzen, installieren wir auch einen DSALM auf

dieser Facility. Für die Kanten $\{i, j\} \in A$ gibt es Kantenkosten von $c(\{i, j\})$. Außerdem addieren wir für $\{i, j\} \in A_1$ noch die Kosten für Glasfaseranschluss $c(i)$ auf diese Kante, da $c(\{i, j\}) = 0$ und wir nur eine Kante von der Facility i haben, ist es egal ob wir diese auf die Kanten auf E' addieren oder auf die Kanten aus A_2 .

Dann lösen wir das Steinerbaum Modell mit der Fluss-Formulierung, wie in der Vorlesung beschrieben. Dabei wählen wir den Graphen $H'' = (V'', A'')$ mit Kantenkosten c' , die Terminals wählen als die Kunden $K \cup \{l\}$ und l ist die Wurzel. Die Lösung des Steinerbaum Modell ist nun ein Baum T_l mit Wurzel l , der jeden Kunden mit der Wurzel verbindet. Also ist jeder Kunde mit der Leitstelle l über 2 Kanten verbunden. Die erste Kante ist eine Kante aus E' und die zweite Kante ist eine Anschlusskante aus A_1 oder A_2 . Für jede Leitstelle werden mit diesem Modell Kosten $c'(T_l)$ berechnet. Addiert man auf diese Kosten noch die Kosten für den Aufbau der Leitstelle l hinzu, erhält man die Kosten der Lösung für Leitstelle l , $\text{Kosten}_l := c(l) + c'(T_l)$.

Um nun zur Lösung des Problems zu kommen, suchen wir die Leitstelle i mit den geringsten Kosten, $i := \arg \min_{l \in L} \text{Kosten}_l$. Das zugehörige Netzwerk setzt sich aus den errechneten kürzesten Wegen $W_{i,f}$ und dem Steinerbaum T_i zusammen. Für jede Kante $\{i, f\} \in T_i$ fügen wir den Weg $W_{i,f}$ zu unserem Netzwerk hinzu. Darüber hinaus fügen wir jede Anschlusskante aus T_i hinzu. Also insgesamt ergibt sich das Netzwerk als die Menge dieser Kanten $(\bigcup_{\{i,f\} \in T_i \cap E'} W_{i,f}) \cup (T_i \cap A)$.

Nochmal zur Zusammenfassung ist hier der Algorithmus aufgeschrieben.

Eingabe : $G = (V, A, c)$, wie in Kapitel 1 beschrieben

Ausgabe: Leitstelle, Netzwerk, Kosten

```

1 forall  $l \in L$  do
2   forall  $f \in F$  do
3     Berechne  $W_{l,f}$  und  $c_{\text{dij}}(\{l, f\})$  mit dem Dijkstra-Algorithmus
      auf  $H'$ 
4   end
5   Berechne Steinerbaum  $T_l$  auf  $H''$  mit Wurzel  $l$ , Terminals
       $K \cup \{l\}$  und Kosten  $c'$ 
6    $\text{Kosten}_l := c(l) + c'(T_l)$ 
7 end
8  $\text{Leitstelle} := i := \arg \min_{l \in L} \text{Kosten}_l$ 
9  $\text{Netzwerk} := (\bigcup_{\{i,f\} \in T_i \cap E'} W_{i,f}) \cup (T_i \cap A)$ 
10  $\text{Kosten} := \text{Kosten}_i$ 

```

Algorithm 1: Algorithmus zum Lösen des P2PGK Problems

Jetzt bleibt nur noch zu zeigen, dass der Algorithmus das optimale Ergebnis des P2PGK Problems liefert.

Satz 1: Die Lösung des Algorithmus 1 ist die optimale Lösung für das P2PGK Problem.

Beweis: **TODO**

□

TODOErgebnisse + Bilder als Übersicht:

Instanz	Naunyn	Berlin	Vehlefan
Kosten	TODO	TODO	TODO
Laufzeit auf PC1	TODO	TODO	TODO

Tabelle 3: Ergebnisse des P2PGK

3.3 Erweiterung des P2PGK

TODOEINLEITUNG SCHÖÖN :(((Es gibt noch zwei weitere Eigenschaften die man beim P2PGK betrachten kann. Zum einen kann davon ausgegangen werden, dass die Nachfrage der Kunden nach größeren Bandbreite steigt und die angegebene Nachfrage an Bandbreite in den nächsten Jahren nicht mehr ausreicht. ... ZUm anderen Profit

Um die steigende Nachfrage an Bandbreite der Kunden zu modellieren, haben wir den angegebenen Bedarf mit verschiedenen Werten multipliziert und dann, wie im Abschnitt 2, die Kupfer Anschlusskanten gelöscht, die den Bedarf der Kunden nicht decken können.

TODOErgbniss bilder reihe und kosten reihe

Nun zu der Erweiterung, wo der Profit für den Anschluss eines Kunden betrachtet wird. Der Profit pro Jahr für den Anschluss eines Kunden $k \in K$ beträgt für Glasfaser $p_1(k)$ und für Kupfer $p_2(k)$, ($p : K \times \{1, 2\} \rightarrow \mathbb{Q}, (k, i) \mapsto p_i(k)$). Dazu führen wir die Variable t ein. Für $t = 0$ ist die Lösung die Lösung für das P2PGK, wie im Abschnitt zuvor. $t = 1$ ist das erste Jahr mit Profit, und so weiter. Um diese Erweiterung unseres P2PGK Problems zu lösen, übergeben wir dem Algorithmus 1 hierbei noch zusätzlich t und p und passen die Kostenfunktion c' wie folgt an.

$$c' : A'' \rightarrow \mathbb{Q}, \{i, j\} \mapsto \begin{cases} c_{\text{dij}}(\{l, f\}), & \text{falls } \{i, j\} \in E \text{ und } j \in F_1 \\ c_{\text{dij}}(\{l, f\}) + c(j), & \text{falls } \{i, j\} \in E \text{ und } j \in F_2 \\ c(\{i, j\}) + c(i) - t \cdot p_1(j), & \text{falls } \{i, j\} \in A_1 \\ c(\{i, j\}) - t \cdot p_2(j), & \text{falls } \{i, j\} \in A_2 \end{cases}$$

Es ist leicht zu sehen, dass der Algorithmus 1 auch hierfür die optimale Lösung ausgibt.

TODOvergleich und zahlen zu dieser Lösung

4 Point-to-multipoint communication

TODO

- 4.1 Point-to-multipoint mit Glasfaser
- 4.2 Point-to-multipoint mit Glasfaser und Kupfer
- 4.3 Erweiterung des P2MPGK

Literaturverzeichnis

Anhang