

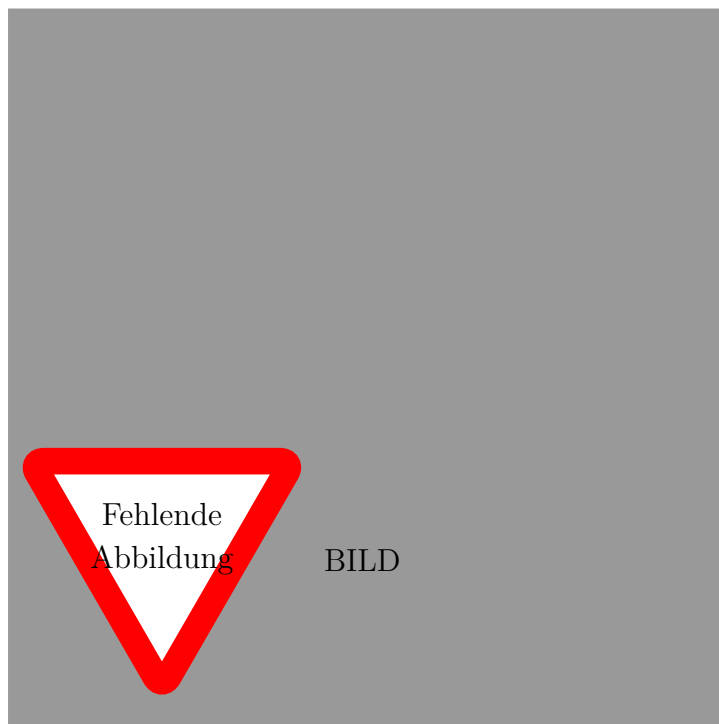
UNIVERSITÄT AUGSBURG

INSTITUT FÜR MATHEMATIK

Ausarbeitung

zum Programmierprojekt

...



von:
Lukas GRAF

Betreut von:
Prof. Dr. Tobias HARKS

„Abstract“

Zusammenfassung/Überblick der Arbeit

1 Capacitated Location Routing (CLR)

1.1 Problemdefinition

Eine Instanz des **Capacitated Location Routing Problems (CLR)** ist gegeben durch:

- einen ungerichteten, zusammenhängenden Graphen $G = (V, E)$,
- einer Partition der Knoten in Klienten \mathcal{C} und Depots \mathcal{F} ,
- einer metrischen Kostenfunktion auf den Kanten $c : E \rightarrow \mathbb{R}_{\geq 0}$,
- Eröffnungskosten für die Fabriken $\phi : \mathcal{F} \rightarrow \mathbb{R}_{\geq 0}$,
- Bedarfen der Klienten $d : \mathcal{C} \rightarrow \mathbb{R}_{\geq 0}$
- und einer einheitlichen Kapazität $u > 0$ für die Fahrzeuge.

Zulässige Lösungen bestehen aus

- einer Teilmenge $F \subseteq \mathcal{F}$ von eröffneten Fabriken
- und einer Menge von Touren $\mathcal{T} = \{T_1, \dots, T_k\}$,

sodass gilt:

- Zu jeder Tour gibt es ein eröffnetes Fabriken $f \in F$, an dem diese startet und endet.
- Alle Touren zusammen erfüllen alle Bedarfe der Klienten.
- Keine der Touren übersteigt die Kapazität u .

Das Optimierungsziel ist es die Gesamtkosten für das Eröffnen der Fabriken und die gefahrenen Touren zu minimieren, also die Minimierung der Kostenfunktion

$$\sum_{T \in \mathcal{T}} c(T) + \sum_{f \in F} \phi(f)^1$$

1.2 Der Algorithmus

Intuition zum Algorithmus

¹Überladung der Funktion c

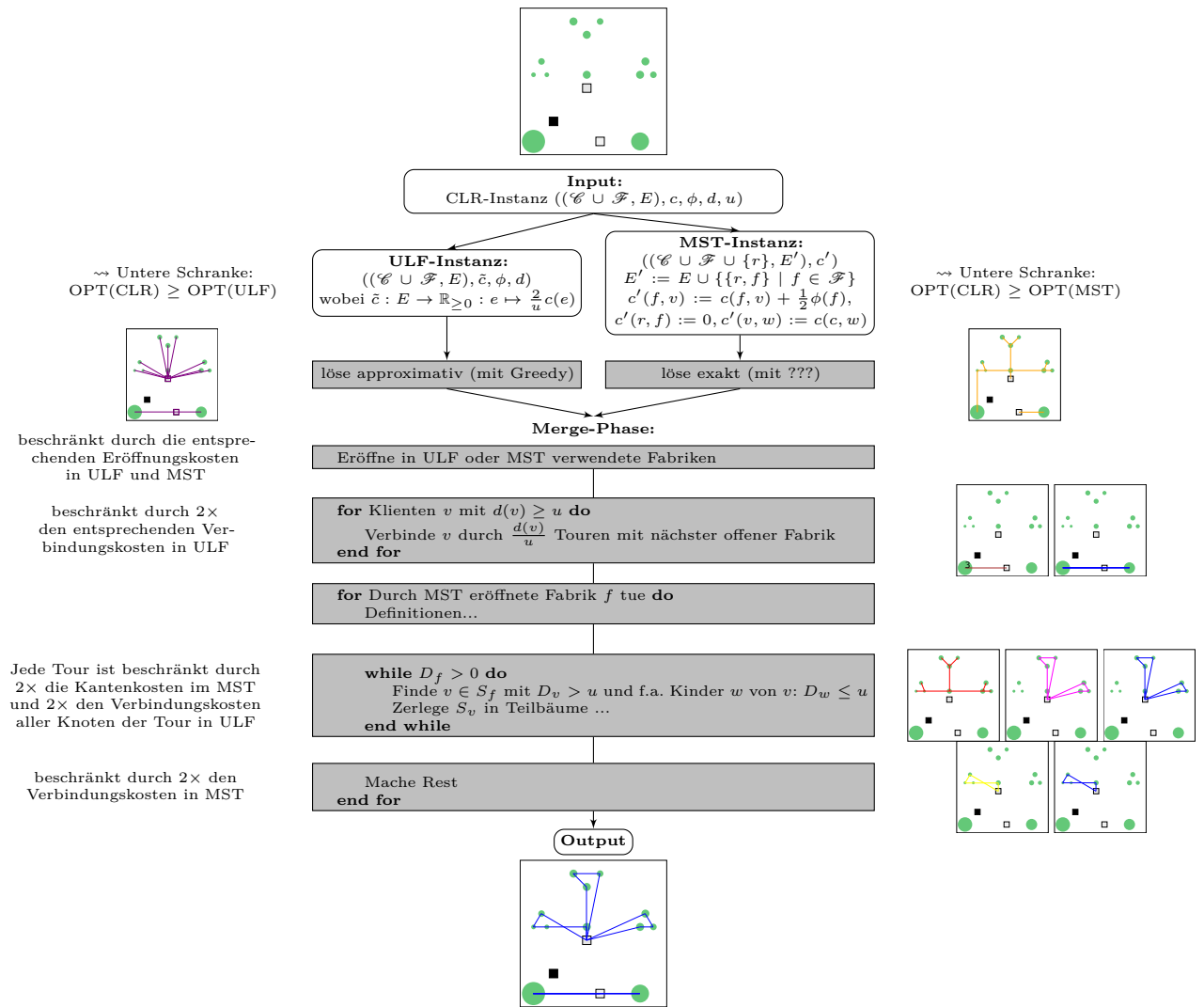


Abbildung 1: Schematische Darstellung des Algorithmus für CLR

1.3 Visualisierung

Beschreibung der Klasse zur Visualisierung

2 CLR with Hard Facility Capacities (CLRHFC)

2.1 Problemdefinition

Eine Instanz von **Capacitated Location Routing with Hard Facility Capacities (CLRHFC)** ist gegeben durch:

- eine Instanz $(G = (\mathcal{C} \cup \mathcal{F}, E), c, \phi, d, u)$ von CLR
- und zusätzlich Kapazitäten der Fabriken $l : \mathcal{F} \rightarrow \mathbb{R}_{\geq 0}$.

Zulässige Lösungen sind Lösungen der zugrunde liegenden CLR-Instanz, die zudem die Kapazitätsschranken der Fabriken einhalten.

Das Optimierungsziel weiterhin die Minimierung der Kostenfunktion der CLR-Instanz.

Bemerkung 2.1.

Im Unterschied zu harten Kapazitäten gibt es auch die Variante mit weichen Kapazitäten (zumindest für ULF) - vgl. zum Beispiel ...

2.2 Lösungsansätze

Zunächst einmal ist klar, dass es nicht genügen wird, nur die Merge-Phase des Algorithmus für CLR anzupassen. Denn die bereits in der ULF- und der MST-Phase gefällte Entscheidung welche und wie viele Fabriken eröffnet werden, muss offenkundig

Ideen und Probleme für Anpassungen

2.3 Algorithmus

Beschreibung des angepassten Algorithmus

2.3.1 Algorithmus 1

2.3.2 Algorithmus 2

2.4 Implementierungen

2.5 Analyse der Algorithmen

2.5.1 Theoretische Betrachtungen

Untere Schranken

Schlechte Beispiele

2.5.2 Heuristische Beurteilung

2.6 Ausblick

Was könnte man verbessern? Welche Probleme gibt es dabei?

Liste der noch zu erledigenden Punkte

Abbildung: BILD	1
Zusammenfassung/Überblick der Arbeit	2
Intuition zum Algorithmus	2
Beschreibung der Klasse zur Visualisierung	3
Im Unterschied zu harten Kapazitäten gibt es auch die Variante mit weichen Kapazitäten (zumindest für ULF) - vgl. zum Beispiel	4
Ideen und Probleme für Anpassungen	4
Beschreibung des angepassten Algorithmus	4
Untere Schranken	4
Schlechte Beispiele	4
Was könnte man verbessern? Welche Probleme gibt es dabei?	5

Literatur

- [HKM13] Tobias Harks, Felix G. König und Jannik Matuschke. „Approximation Algorithms for Capacitated Location Routing“. In: *Transportation Science* 47.1 (2013), S. 3–22. DOI: <http://dx.doi.org/10.1287/trsc.1120.0423>. URL: <http://researchers-sbe.unimaas.nl/tobiasharks/wp-content/uploads/sites/29/2014/02/HKM-TS-2013.pdf>.
- [Tur10] Mark Turney. *simple-svg*. Google Code Archive. simple-svg ist eine header-only C++ Library, mit deren Hilfe einfache svg-Graphiken erstellt werden können. 2010. URL: <https://code.google.com/archive/p/simple-svg/>.