

Experimentelle Ergebnisse zum Network-Simplex-Algorithmus

Max Kanold

30. August 2018

Inhaltsverzeichnis

1	Einführung	3
2	Network-Simplex-Algorithmus	4
2.1	Min-Cost-Flow-Problem	4
2.2	Algorithmus	5
2.3	Umsetzung	5
2.3.1	Spezielle Konstrukte	5
3	Experimentelle Ergebnisse	6
4	Ausblick	7

Kapitel 1

Einführung

Bla. Zum Beispiel in Kapitel 2.3 habe ich programmiert.

Kapitel 2

Network-Simplex-Algorithmus

Das Simplex-Verfahren, zu welchem eine Einführung in [1] gefunden werden kann, löst Lineare Programme in der Praxis sehr schnell, obwohl die Worst-Case-Laufzeit nicht polynomiell ist. Jedes Netzwerkproblem lässt sich als Lineares Programm darstellen und somit durch das Simplex-Verfahren lösen, durch die konkrete Struktur solcher Probleme genügt jedoch der vereinfachte Network-Simplex-Algorithmus. Auch für diesen gibt es exponentielle Instanzen (siehe [2]), in der Praxis wird er trotzdem vielfach verwendet.

2.1 Min-Cost-Flow-Problem

Definition 1. Ein **Netzwerk** ist ein Tupel (G, b, c, u) , wobei $G = (V, E)$ ein gerichteter Graph, $b : V \rightarrow \mathbb{R}$ eine b-Wert-Funktion, $c : E \rightarrow \mathbb{R}$ eine Kostenfunktion und $u : E \rightarrow \mathbb{R}_{\geq 0}$ eine Kapazitätsfunktion seien.

Anmerkung. Knoten mit positiven b-Wert werden als Quellen, solche mit negativen als Senken bezeichnet.

Ein ungerichteter Graph kann durch das Verwandeln jeder Kante $\{v, w\}$ in zwei Kanten (v, w) und (w, v) zu einem gerichteten modifiziert werden.

Definition 2. Ein **maximaler Fluss** auf einem Netzwerk $(G = (V, E), b, c, u)$ ist eine Abbildung $f : E \rightarrow \mathbb{R}_{\geq 0}$, die folgende Eigenschaften erfüllt:

$$(i) \quad \forall e \in E : f(e) \leq u(e)$$

$$(ii) \quad \forall v \in V : \sum_{(w,v) \in E} f((w,v)) - \sum_{(v,w) \in E} f((v,w)) + b(v) = 0$$

Der **Wert** von f ist $v(f) = \frac{1}{2} \cdot \sum_{v \in V} |b(v)|$,
die **Kosten** von f sind $c(f) = \sum_{e \in E} f(e) \cdot c(e)$.

Beim *Min-Cost-Flow-Problem* wird unter allen maximalen Flüssen einer mit minimalen Kosten gesucht. Sind die Kapazitäten unbeschränkt, so wird es als *Transportproblem* bezeichnet.

Für diese Bachelorarbeit wurde angenommen, dass u und c auf \mathbb{N} sowie b auf \mathbb{Z} abbildet, um Gleitkommazahlungenauigkeit zu vermeiden. Durch eine

entsprechende Skalierung des Problems können die Funktionen nach $\mathbb{R}_{\geq 0}$ bzw. \mathbb{R} hinreichend genau angenähert werden. Zusätzlich wird davon ausgegangen, dass $\sum_{v \in V(G)} b(v) = 0$ ist, Angebot und Nachfrage also ausgeglichen sind. Des Weiteren ist in der konkreten Implementierung E keine Multimenge; es sind keine parallelen Kanten vorgesehen.

2.2 Algorithmus

[3, Dantzig, 1951] und [4, Orden, 1956] vereinfachten das Simplex-Verfahren zum Netzwerk-Simplex-Algorithmus; die folgende Beschreibung orientiert sich zuerst an [1, S. 291 ff.] zur Lösung des Transportproblems, danach wird der Algorithmus anhand von TODO auf den allgemeinen, durch Kapazitäten beschränkten Fall erweitert.

Definition 3. Ein **Baum** T ist ein ungerichteter, zusammenhängender und kreisfreier Graph.

Ein Teilgraph $T = (V', E')$ eines ungerichteten Graphen $G = (V, E)$ heißt **aufspannender Baum**, wenn T ein Baum und $V' = V$ ist.

Anmerkung. Sprechen wir bei einem gerichteten Graphen G über einen aufspannenden Baum, so bezieht sich das stets auf einen aufspannenden Baum des G zugrundeliegenden ungerichteten Graphen.

2.3 Umsetzung

Hier beginnt mein schönes Werk ...

2.3.1 Spezielle Konstrukte

... und hier endet es.

Die Klasse Circle

Kreise halt.[1]

Der Rest halt

Kleinkram.

Kapitel 3

Experimentelle Ergebnisse

Alle scheiße.

Kapitel 4

Ausblick

La la la.

Literaturverzeichnis

- [1] V. Chvátal, *Linear Programming*, pp. 291 ff. Series of books in the mathematical sciences, W. H. Freeman, 1983.
- [2] N. Zadeh, “A bad network problem for the simplex method and other minimum cost flow algorithms,” *Mathematical Programming*, vol. 5, no. 1, pp. 255–266, 1973.
- [3] G. B. Dantzig, “Application of the simplex method to a transportation problem,” in *Activity Analysis of Production and Allocation* (T. C. Koopmans, ed.), ch. XXIII, pp. 359–373, New York: Wiley, 1951.
- [4] A. Orden, “The transshipment problem,” *Management Science*, vol. 2, no. 3, pp. 276–285, 1956.