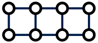
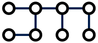
Vorlesung Diskrete Strukturen
Graphen: Zwei Greedy-Algorithmen für Gerüste

Bernhard Ganter

Institut für Algebra
TU Dresden
D-01062 Dresden
bernhard.ganter@tu-dresden.de

WS 2009/10

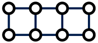
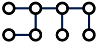
Ein **Gerüst** (engl.: **Spanning Tree**, deshalb gelegentlich auch deutsch **Spannbaum** oder ähnlich) eines Graphen (V, E) ist ein Baum (V, F) mit $F \subseteq E$.

Beispiel: Graph   ein Gerüst

Ein Gerüst ist also ein Baum, der die gleiche Eckenmenge und eine Teilmenge der Kantenmenge des gegebenen Graphen hat.

Gerüste

Ein **Gerüst** (engl.: **Spanning Tree**, deshalb gelegentlich auch deutsch **Spannbaum** oder ähnlich) eines Graphen (V, E) ist ein Baum (V, F) mit $F \subseteq E$.

Beispiel: Graph   ein Gerüst

Ein Gerüst ist also ein Baum, der die gleiche Eckenmenge und eine Teilmenge der Kantenmenge des gegebenen Graphen hat.

Der folgende Hilfssatz ist für endliche Graphen offensichtlich:

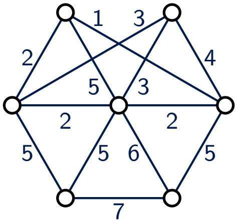
Hilfssatz

Ein Graph hat genau dann ein Gerüst, wenn er zusammenhängend ist.

Gewichtete Graphen

Ein **gewichteter Graph** (V, E, wt) hat zusätzlich zur Eckenmenge V und zur Kantenmenge $E \subseteq \binom{V}{2}$ noch eine **Gewichtsfunktion**

$$\mathrm{wt} : E \to \mathbb{R}.$$



Eine Gewichtsfunktion kann bei kleinen Graphen im Diagramm notiert werden.

Mögliche Interpretationen einer Gewichtsfunktion sind:

- Kosten
- Wahrscheinlichkeiten
- Kapazitäten
- Distanzen
- u.v.m.

Konstruktion eines gewichtsminimalen Gerüstes

Das Problem

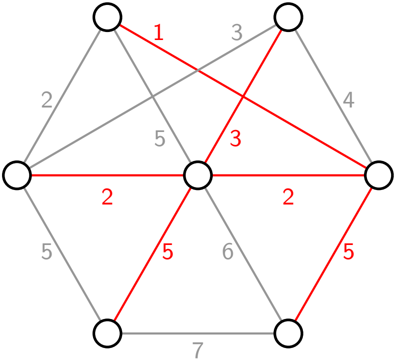
GEGEBEN: Ein endlicher zusammenhängender gewichteter Graph (V, E, wt).
GESUCHT: Ein **gewichtsminimales Gerüst**, d.h. ein Gerüst (V, T) des Graphen, für das $\sum_{\{a,b\}\in T} \mathrm{wt}(\{a, b\})$ minimal ist.

wird gelöst durch den **Greedy-Algorithmus** von Kruskal:

$T \leftarrow \emptyset$;
while (V, T) ist kein Gerüst **do**
$\{$ Wähle unter allen Kanten $e \in E \setminus T$, für die $T \cup \{e\}$ kreisfrei ist, eine minimalen Gewichts, e_{min}, und setze $T \leftarrow T \cup \{e_{min}\}$. $\}$

Statt „gewichtsminimal" sagt man auch gern **kostenminimal**.

Beispiel



Die roten Kanten bilden ein Gerüst vom Gewicht 18 im Beispielgraphen. Es gibt noch 11 weitere Gerüste dieses Gewichts, alle anderen haben größeres Gewicht.

Korrektheitsbeweis für den Greedy-Algorithmus

Der Abstand zweier Ecken

Gegeben sei ein gewichteter Graph (V, E, wt) und zwei Ecken $u, v \in V$. Das Gewicht eines Weges $W_{u,v}$ von u nach v sei die Summe der Gewichte der Kanten in diesem Weg.

Wenn es unter allen Wegen von u nach v in (V, E, wt) einen minimalen Gewichts gibt, dann nennen wir dieses Gewicht die **Distanz** $\mathrm{dist}(u, v)$ von u und v in dem gewichteten Graphen.

Die Distanz existiert für je zwei Ecken, wenn der betrachtete Graph *endlich, zusammenhängend* und *positiv gewichtet* ist.

Dijkstras Algorithmus

GEGEBEN: Ein endlicher zusammenhängender positiv gewichteter Graph (V, E, wt) und eine Ecke $u \in V$.
GESUCHT: Ein Gerüst mit der Eigenschaft, dass für jede Ecke v das Gewicht des (eindeutig bestimmten) Weges von u nach v minimal ist, also gleich $\mathrm{dist}(u, v)$.

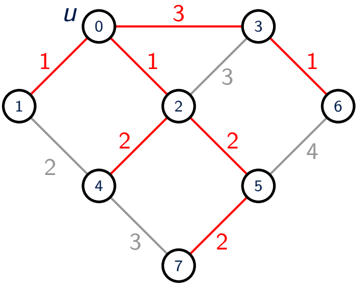
Algorithmus:

(1) Setze $i \leftarrow 0$, $u_0 := u$, $V_0 := \{u_0\}$, $E_0 := \emptyset$, $W(u_0) := 0$.

(2) Sei $V_i = \{u_0, \ldots, u_i\}$ und $E_i = \{e_1, \ldots, e_i\}$ berechnet.
Wenn $i = n - 1$, stopp.
Anderenfalls betrachte für alle Kanten $\{v, w\}$ mit $v \in V_i$, $w \notin V_i$ den Ausdruck $f(\{v, w\}) := W(v) + \mathrm{wt}(\{v, w\})$ und wähle darunter $\{\overline{v}, \overline{w}\}$ mit $f(\{\overline{v}, \overline{w}\})$ minimal und $\overline{v} \in V_i$.
Setze $u_{i+1} := \overline{w}$, $e_{i+1} := \{\overline{v}, \overline{w}\}$, $V_{i+1} := V_i \cup \{u_{i+1}\}$, $E_{i+1} := E_i \cup \{e_{i+1}\}$, $W(u_{i+1}) := f(e_{i+1})$ und $i \leftarrow i + 1$.
Wiederhole (2).

Beispiel

Eine ausführliche Darstellung von Dijkstras Algorithmus findet man z.B. in dem Buch *Diskrete Mathematik* von Martin Aigner. Daraus ist auch das folgende Beispiel entnommen. Es zeigt einen gewichteten Graphen mit ausgezeichneter Ecke u und ein von Dijkstras Algorithmus darin konstruiertes Gerüst (rot).



Die Zahlen in den Ecken geben den Laufindex i wieder.

Behauptung: Der Algorithmus tut, was er soll.

Satz

Für jeden endlichen zusammenhängenden positiv gewichteten Graphen $G := (V, E, wt)$ mit einer festgewählten Ecke u konstruiert der Algorithmus von Dijkstra ein Gerüst T mit der Eigenschaft, dass für alle $v \in V$ der Weg von u nach v in T minimales Gewicht unter allen Wegen von u nach v in T hat.

Um diesen Satz zu beweisen, betrachten wir die Folge $T_i := (V_i, E_i)$, $i := 0, \ldots, n-1$ der vom Algorithmus konstruierten Graphen und behaupten:

Hilfssatz

Für alle i ist T_i ein Baum mit der Eigenschaft, dass $W(v) = dist(u, v)$ für alle $v \in V_i$ gilt.

Beweis des Hilfssatzes mittels Induktion über i

Beweis.

Für $i = 0$ ist $V_i = \{u\}$, $E_i = \emptyset$ und die Behauptung ist offensichtlich richtig.
Sei nun die Behauptung für ein $i < n - 1$ beweisen, T_i also ein Baum, für den die Behauptung stimmt, und sei $\{\overline{v}, \overline{w}\}$ die in Schritt (2) konstruierte Kante.
Weil $\overline{w} \notin V_i$ ist, ist $T_{i+1} := (V_i \cup \{\overline{w}\}, E_i \cup \{\{\overline{v}, \overline{w}\}\})$ ein Baum. Zu zeigen bleibt, dass $W(\overline{w}) = \text{dist}(u, \overline{w})$ gilt.
Betrachte also irgendeinen Weg $(u = a_0, \ldots, a_r = \overline{w})$ von u nach v in G. Er enthält eine Kante $\{a_j, a_{j+1}\}$ die nicht zu E_i gehört.

$\square$