Kapitel 2: Minimale aufspannende Bäume

2.1 Problemstellung

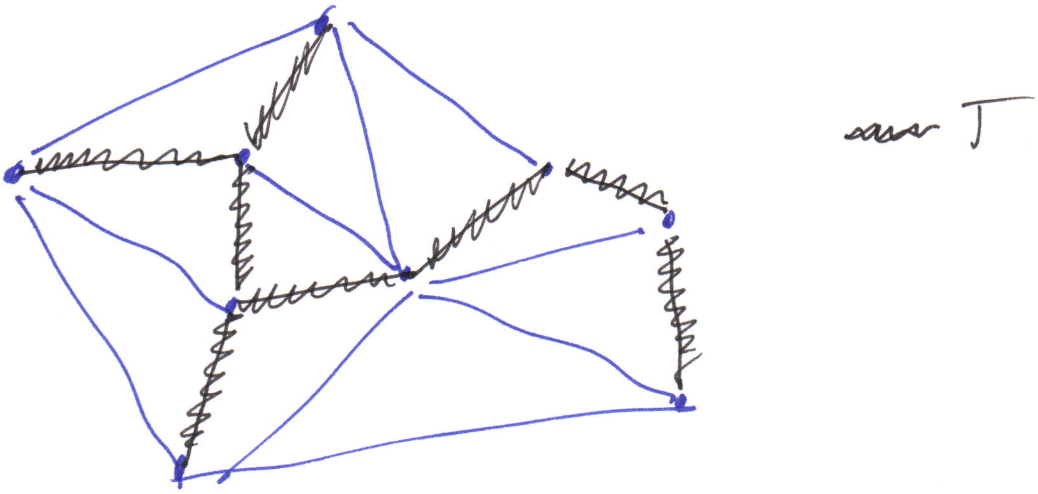
In diesem Kapitel:

Problem 2.1 (kürzestes zusammenhängendes Netzwerk)

Gegeben: Ein Graph $G=(V,E)$
mit Gewichtsfunktion $c: E \to \mathbb{R}_+$, $e \mapsto c_e$

Gesucht: Eine Kantenmenge $F \subseteq E$ möglichst geringen Gesamtgewichts $\sum_{e \in T} c_e$,
so dass in $T=(V,E)$ alle Knoten verbunden sind.

Beispiel:



~~Wir werden in der~~

Fragen:
(2) Wie findet man T systematisch?
(1) Welche besonderen Eigenschaften haben Lösungen?
(3) Gibt es algorithmische Grundideen, die auch in anderen Situationen funktionieren?
(4) Wie setzt man das alles möglichst effizient um?

Wir werden sehen, dass es hier sehr klare Strukturen gibt, die einen sehr einfachen algorithmischen Grundprinzip anwendbar machen!

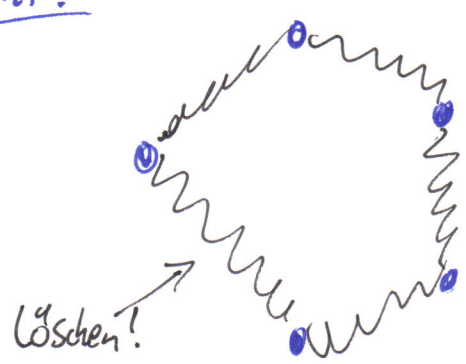
Zunächst einmal beobachten wir, dass man eine Lösung
- konstruktiv (Kanten hinzufügen)
- destruktiv (Kanten wegnehmen)

bauen kann. (Gedanklich ist beides nützlich, aber am Ende ist eines effizienter.

Beobachtung 2.2

Eine optimale Lösung kann keinen Kreis enthalten.

Argument:



Entfernt man aus einem Kreis eine Kante, so bleibt der Rest immer noch verbunden!
Da die Kantengewichte positiv sind, erhält man eine bessere Lösung.

Also:

Satz 1.3

Eine optimale Lösung für Problem 2.1 ist notwendigerweise
- zusammenhängend
- kreisfrei

– also ein Baum!

Beweis: Klar

~~2.2 Eigenschaften von Bäumen~~

Algorithmische Ideen:

Destruktiv: (1) Finde einen Kreis
(2) Entferne die teuerste Kante

Konstruktiv (A) (1) ~~Füge~~ Versuche ~~Kanten~~ in aufsteigendem Gewicht Kanten einzufügen

(2) Füge Kante nur ein, wenn etwas verbunden wird, was noch nicht verbunden war.

Konstruktiv (B) (1) Füge Kante ein, um etwas zu verbinden, was noch nicht verbunden war

(2) Verwende günstigste Kante dafür.

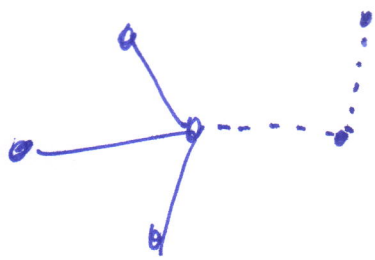
Wir werden sehen, dass alle diese einfachen Ideen funktionieren (wenn auch unterschiedlich gut). Allerdings müssen wir ein wenig nachdenken, um sicherstellen zu können, dass und wie alles funktioniert!

2.2 Eigenschaften von Bäumen

Wie viele Kanten hat ein Baum mit n Knoten?

Beobachtung 2.4

Mit jeder eingefügten Kante nimmt die Zahl der Zshgskomp. um 1 ab – mit jeder gelöschten um 1 zu.



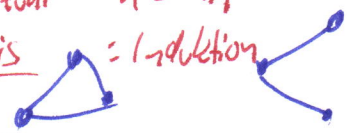
Also: Ein Baum hat $n-1$ Kanten!

Ersetzt während VL:

Lemma 2.5

Ein w/q mit n Knoten, m Kanten, p Zshgs.komp. erfüllt $n = m + p$

Beweis: Induktion



Beobachtung 2.5

Nicht jede Menge von $(n-1)$ Kanten bildet einen Baum!

Wichtig ist also das Zusammenspiel von

- Kreisfreiheit → nicht zu viele Kanten
- Zusammenhang → nicht zu wenige Kanten

Diese Eigenschaften kann man in verschiedener Form zusammenstellen:

Satz 2.6 (Eigenschaften von Bäumen)

Sei ~~G = (V, E)~~ ein Graph mit n Knoten. $T = (V, F)$

Dann sind äquivalent:

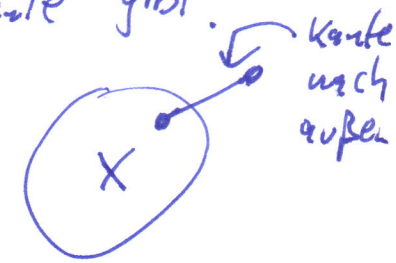
(a) T ist ein Baum (d.h. zshgd. und kreisfrei)

(b) T hat $(n-1)$ Kanten und ist kreisfrei.

(c) T hat $(n-1)$ Kanten und ist zshgd.

(d) T ist ein minimaler zusammenhängender Graph.
(D.h. keine Kante kann entfernt werden, ohne dass der Zusammenhang verloren geht.)

(e) T ist ein minimaler Graph, für den es für jede Knotenmenge $\emptyset \neq X \subsetneq V$ eine nach außen verbindende Kante gibt.
(Man schreibt $\delta(X) \neq \emptyset$.)



(f) T ist ein maximaler kreisfreier Graph.
(D.h. Hinzufügen einer Kante erzeugt immer einen Kreis.)

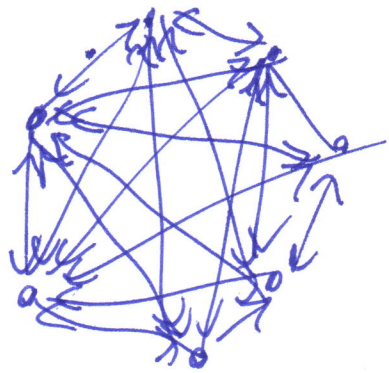
(g) T enthält einen eindeutigen Pfad zwischen je zwei Knoten.

Beweis:

Wir machen uns klar:

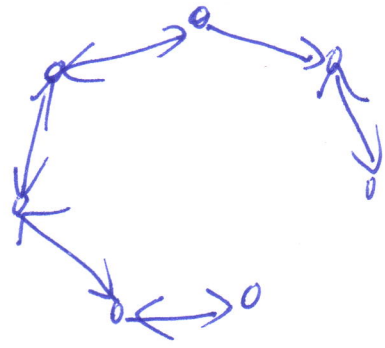
$$(a) \Rightarrow (g) \Rightarrow (e) \Rightarrow (d) \Rightarrow (d) \Rightarrow (f) \Rightarrow (b) \Rightarrow (c) \Rightarrow (a)$$

(Das nennt man „Ringschluss“; praktisch, weil man nicht 21 Äquivalenzen (42 Schlüsse!!



und auch nicht

~~21~~ Äquivalenzen (14 Schlüsse!)



braucht, sondern nur 8!