Algorithmik Blatt 7 Teil 1

Mtr.-Nr. 6329857

Universität Hamburg — 6. Dezember 2019

Aufgabe 19

```
1: procedure ListAugmentingPaths(maxFlow)
2:     while |maxFlow| > 0 do
3:         path = FindSourceTargetPath(maxFlow)
4:         minEdge = FindLimitingEdge(path)
5:         output path, minEdge.flow
6:         foreach e in path.edges do
7:             ReduceFlow(maxFlow, e, minEdge.flow)
8:
```

Terminierung

Der Algorithmus Terminiert, weil sowohl die innere als auch die äußere Schleife terminieren:

- Die innere Schleife iteriert über eine endliche Anzahl Elemente, da der Pfad endliche Länge hat
- Die äußere Schleife iteriert, weil in jeder iteration mindestens eine Kante (die mit dem niedrigsten Fluss in dem gefundenen Pfad) aus dem Fluss entfernt wird, der Fluss sich also mit jeder Iteration verringert und nie erhöht.

In einem Fluss > 0 kann auch immer ein Pfad von der Quelle zur Senke gefunden werden. In einem solchen Pfad existiert immer mindestens eine Kante, die den Fluss begrenzt.

Korrektheit

Der Algorithmus ist korrekt, weil der Maximale Fluss in jeder Iteration entlang eines gesamten Pfades von der Quelle zur Senke um einen festen verringert wird. Der daraus entstehende Fluss ist also wieder ein gültiger Fluss. Bezüglich des neuen Flusses ist dieser Pfad ein augmentierender Pfad der wieder die Ausgangsfluss ergeben würde. Da dies für jede Iteration gilt und der Algorithmus mit einem Fluss von 0 terminiert, kann ein Fluss von 0 entgegengesetzter Reihenfolge durch hinzunehmen der entfernten Pfade wieder zum maximalen Fluss schrittweise augmentiert werden.

Anzahl der Pfade

Da in jeder Iteration mindestens eine Kante aus dem Fluss entfernt wird, durchläuft der Algorithmus maximale so viele Iterationen wie der Fluss, also auch der Netzwerk, Kanten hat. Pro Iteration wird nur ein Pfad ausgegeben, also wird eine Folge von Pfaden gefunden, deren Länge durch die Anzahl der Kanten im Netzwerk begrenzt ist.

Aufgabe 20

Wir betrachten möglichen Operationen des Push-Relabel Algorithmus gerennt voneinander:

Relabel Die Anzahl der Operationen ist, wie in der Vorlesung gezeigt, durch $2 \cdot |V|^2$ beschränkt. $2 \cdot |V|^2 < |V| \cdot |E|$, weil wir einen zusammenhängenden Graphen betrachten.

Sättigender Push Ebenfalls wie in der Vorlesung gezeigt, liegt die Anzahl dieser Operationen in $2 \cdot |V| \cdot |E|$

Nicht-Sättigender Push Jeder Push erhöht den Fluss über eine Kante um mindestens 1. Jede Kante kann aber höchstens einen Fluss von $k-1$ bekommen bevor sie gesättigt ist. Eine Kante kann also höchstens $k-1$ mal nicht-sättigend gepusht werden. Das Residuale Netzwerk hat höchstens $2|E|$ Kanten, also können höchstens $2 \cdot (k-1) \cdot |E|$ nicht sättigende Pushes durchgeführt werden — $2 \cdot (k-1) \cdot |E| < k \cdot |V| \cdot |E|$

Da Push-Relabel nur diese drei Operationen durchführt, werden insgesamt höchstens

$$2 \cdot |V|^2 + 2 \cdot |V| \cdot |E| + 2 \cdot (k-1) \cdot |E| = |V| \cdot |E| + 2 \cdot |V| \cdot |E| + 2 \cdot (k-1) \cdot |E|$$

Operationen durchgeführt, die jeweils nur konstante Zeit benötigen. Daraus ergibt sich eine obere Lauzeitschranke von:

$$\mathcal{O}(k \cdot |V| \cdot |E|)$$