Beobachtungen

Lemma. *Sei $G = (V, E)$ ein gerichteter Graph, dessen Kanten alle dieselbe Länge $\ell \in [0, \infty)$ besitzen. Dann wird im Dijkstra* `decreaseKey` *nie aufgerufen, falls die Länge eines kürzesten Weges von einem Startknoten aus gesucht wird.*

Insbesondere erhalten wir hieraus:

Korollar. *Bei der Bestimmung der Länge maximaler Teilsequenzen von rRNA-Sequenzen – mit Hilfe des Dijkstra – wird* `decreaseKey` *nicht aufgerufen, d.h.: sobald wir den Zielknoten gefunden haben, haben wir schon einen kürzesten Weg.*

Wir bezeichnen im Folgenden mit $d(v, T)$ den minimalen zum Zeitpunkt T bekannten Abstand eines Knotens $v \in V$ zum Startknoten.

Beweis (des Lemmas). Angenommen, `decreaseKey` wird aufgerufen, d.h. es existiert ein Knoten $w \in V$, auf den es das erste Mal angewendet wird (dabei ist w insbesondere nicht der Startknoten).

Also gibt es $v, \widetilde{v} \in V$ mit $(v, w), (\widetilde{v}, w) \in E$, so dass:

- `extractMin` liefert zu einem Zeitpunkt T den Knoten v mit Abstand $d(v, T)$ und es ist $d(w, T) = d(v, T) + \ell$ (etwa wenn w in die priority queue eingefügt wird).

- Zu einem späteren Zeitpunkt $\widetilde{T}$ liefert `extractMin` den Knoten $\widetilde{v}$ mit Abstand $d(\widetilde{v}, \widetilde{T})$, was zum ersten Aufruf von `decreaseKey` führt. Also ist

$$d(\widetilde{v}, \widetilde{T}) + \ell = d(w, \widetilde{T}) < d(w, T) = d(v, T) + \ell,$$

  d.h. $d(\widetilde{v}, \widetilde{T}) < d(v, T)$.

War $\widetilde{v}$ zum Zeitpunkt T schon in der priority queue? Nein, denn wegen der Minimalität von w wäre dann $d(\widetilde{v}, T) \leq d(\widetilde{v}, \widetilde{T}) < d(v, T)$, d.h. `extractMin` hätte zu T nicht den Knoten v zurückliefern können.

Also wurde $\widetilde{v}$ erst später eingefügt. Dann ist aber (wegen ⑤, 10.)

$$d(\widetilde{v}, \widetilde{T}) \geq d(v, T) + \ell,$$

im Widerspruch zu $d(\widetilde{v}, \widetilde{T}) < d(v, T)$. □

Lemma. *Sei* $G = (V, E)$ *ein gerichteter Graph, dessen Kantenlängen kleiner als* $m \in (0, \infty)$ *sind. Weiter sei der Graph* $\bmod m$ *konsistent, d.h. die Längen von Wegen zwischen zwei Knoten unterscheiden sich höchstens um ein ganzzahliges Vielfaches von* m*. Wird nun – mit dem Algorithmus von Dijkstra – die Länge eines kürzesten Weges von einem Startknoten aus gesucht, so wird dabei* `decreaseKey` *nie aufgerufen.*

Insbesondere folgt:

Korollar. *Auch bei der Suche von kürzesten Verbindungen in Fahrplangraphen wird* `decreaseKey` *nicht aufgerufen.*

Beweis (des Lemmas). Angenommen, `decreaseKey` wird aufgerufen. Etwa das erste Mal zum Zeitpunkt $\widetilde{T}$ auf $w \in V$ (insbesondere ist w also nicht der Startknoten). Dann existieren $v, \widetilde{v} \in V$ mit $(v, w), (\widetilde{v}, w) \in E$, so dass:

- Zum Zeitpunkt $T < \widetilde{T}$ liefert `extractMin` den Knoten v mit Abstand $d(v, T)$ und w wird in die queue eingefügt, d.h.

$$d(w, T) = d(v, T) + \ell(v, w).$$

- Zu $\widetilde{T}$ liefert `extractMin` den Knoten $\widetilde{v}$, und es kommt zu einem Aufruf von `decreaseKey`, d.h. es existiert ein $k \in \{1, 2, \ldots\}$ mit

$$d(\widetilde{v}, \widetilde{T}) + \ell(\widetilde{v}, w) = d(w, \widetilde{T}) = d(w, T) - k \cdot m.$$

Also ist

$$\begin{aligned} d(\widetilde{v}, \widetilde{T}) &\leq d(\widetilde{v}, \widetilde{T}) + \ell(\widetilde{v}, w) \\ &= d(w, \widetilde{T}) \\ &= d(w, T) - k \cdot m \\ &= d(v, T) + \ell(v, w) - k \cdot m \\ &< d(v, T) + (1 - k) \cdot m \\ &\leq d(v, T), \end{aligned}$$

und damit $d(\widetilde{v}, \widetilde{T}) < d(v, T)$.

Analog zum Lemma über Graphen mit gleichlangen Kanten erhalten wir hieraus einen Widerspruch. □