

Zusammenfassung Platzeffiziente Alg.

© Tim Baumann, <http://timbaumann.info/uni-spicker>

Ziel. Algorithmen entwerfen, die wenig Speicherplatz und Speicherzugriffe benötigen, aber trotzdem schnell sind.

Problem (Erreichbarkeit). Gegeben sei ein gerichteter oder ungerichteter Graph, ein Startknoten und ein Zielknoten darin. Frage: Ist der Zielknoten vom Startknoten erreichbar?

Algorithmus. Algorithmen, mit denen man Problem lösen kann, sind Breiten- und Tiefensuche.

Lem. Es sei ein Graph mit n Knoten und m Kanten gegeben. Tiefensuche benötigt $\Theta(n + m)$ Zeit und $\Theta(n \log n)$ Speicherplatz.

Algorithmus (Savitch).

```
1: function SREACHABLE( $u, v, k$ )
2:   if  $u = v$  then return true
3:   if  $k = 0$  then return false
4:   if  $(u, v) \in E$  then return true
5:   if  $k = 1$  then return false
```

```
6:   for  $x \in V$  do
7:     if REACHABLE( $u, x, \lfloor \frac{n}{2} \rfloor$ )  $\wedge$  REACHABLE( $x, v, \lceil \frac{n}{2} \rceil$ ) then
8:       return true
9:   return false
```

```
10: return reachable(s,t,n-1)
```

Lem. Savitch's Algorithmus löst das Erreichbarkeits-Problem in $\mathcal{O}((\log n)^2)$ Zeit.

Bem. Die Laufzeit von Savitch's Algorithmus ist allerdings sehr schlecht.