Mar 4

Recall $f(1) \leq f(2) \cdots \leq f(n)$

Interval Scheduling

Greedy alg:

$O(n)$ $\begin{cases} 0. \quad R = [n] \leftarrow O(n) \\ 1. \quad S = \emptyset \leftarrow O(1) \end{cases}$

2. While $R \neq \emptyset \leftarrow \leq n$

$O(n)$ $\begin{cases} (2.1) \text{ Pick } i \in R \text{ with the smallest index} \leftarrow O(n) \\ (2.2) \text{ Add } i \text{ to } S \leftarrow O(1) \\ (2.3) \text{ Remove all } j \text{ that conflict with } i \text{ from } R \leftarrow O(n) \end{cases}$

3. Return $S^* = S \rightarrow O(n)$

Overall runtime: $O(n) + n \cdot O(n) + O(n) = O(n) + O(n^2) + O(n)$

$= O(n^2)$ ∎

Shortest Path Problem

Input: Directed graph $G = (V, E)$

$s \in V$

"length" $\longrightarrow$ $l_e \geq 0 \quad \forall e \in E$

integer

Output: $\forall\, t \in V$, output a shortest s-t path

w.r.t. the length of the path

$$l(P) = \sum_{e \in P} l_e$$

Simpler version: Only output $d(t) \quad \forall\, t \in V$

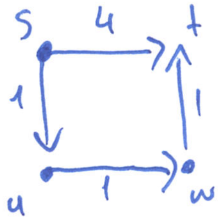
length of the shortest s-t path

Special case: $l_e = 1 \quad \forall e \in E \equiv$ same problem HW 3 Q 3

[also if $l_e = L \quad \forall e \quad (L \geq 1)$] [Run BFS on s & layer of # $t = d(t)$]

<u>General case</u>: $l_e > 0 \quad \forall e \in E$ | Idea: Reduce this to the case of $l_e = 1 \ \forall e$

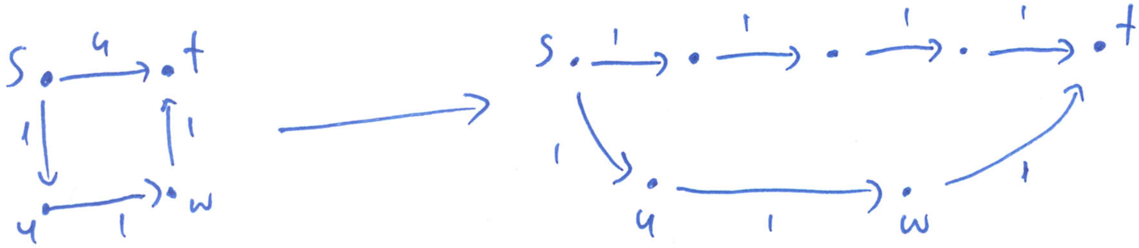
<u>Idea</u>: Ignore l_e (i.e. just assume $l_e = 1 \ \forall e$) & run alg from HW3 Q3

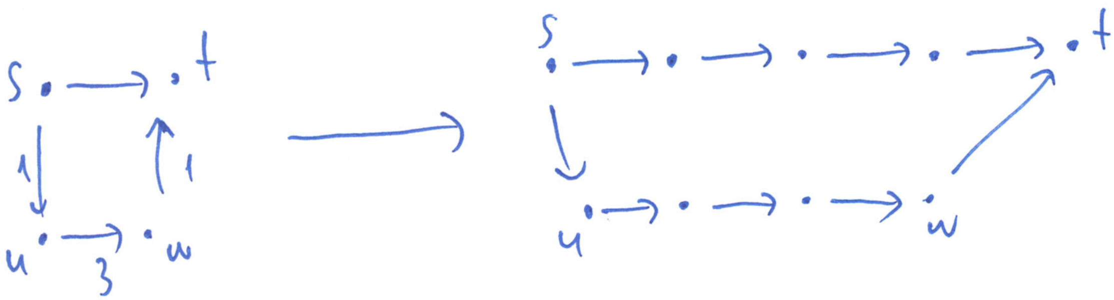


$d(t) = 3$ $\xrightarrow[\text{ignore } l_e]{\text{Run}}$

mistakenly claims
↳ $d(t) = 4$

<u>Idea 2</u>: Replace each edge $e \in E$ by a path of length e





G　　　　　G'

<u>Claim</u>: a shortest path in G $\Leftrightarrow$ equivalent shortest path in G'

$\Rightarrow$ Run HW 3 Q 3 on G'

<u>Correctness</u>: Claim + correctness of HW3 Q3