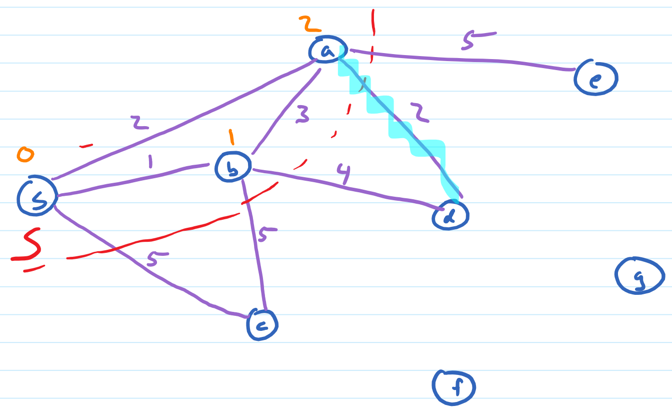
Shortest Paths

min total weight



$d(v)$

0   s is a shortest overall path (tot weight 0)

1   sb is shortest path $s \rightsquigarrow b$ (all other paths start $s \to a \rightsquigarrow b$ tot weight ≥ 2
$s \to b \rightsquigarrow b$ ≥ 1
$s \to c \rightsquigarrow b$ ≥ 5

2   sa is shortest path $s \rightsquigarrow a$

$v \in S$, $\exists u \in S$ s.t. $(u,v) \in E$

next shortest path is to some vert v adj to S (paths $s \rightsquigarrow v$, $v \notin S$: $\underbrace{s \rightsquigarrow x \to y}_{c_1}$ $\rightsquigarrow v$) (c_2: $s \rightsquigarrow v$; $x \in S$, $y \notin S$)

$$\min_{\substack{u \in S \\ (u,v) \in E \\ v \in V - S}} d(s) + \ell(u,v)$$

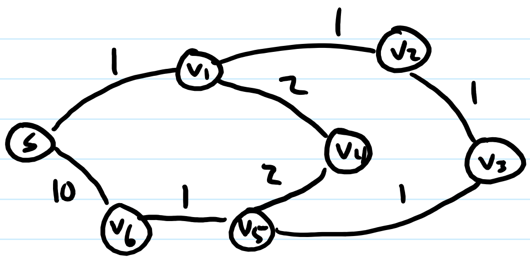
$c_1 \leq c_2$
so consider y (adj to S) instead of v

Dijkstra(G, ℓ)
$S \leftarrow \{s\}$
$d(s) \leftarrow 0$
while $S \neq V$
choose $v \notin S$ to minimize $d'(v) = \min_{\substack{u \in S \\ (u,v) \in E}} d(u) + \ell(u,v)$

(and next-to-last vertex on shortest path $s \rightsquigarrow v$ is u that yielded min)

Shortest Paths

Given weighted G (directed or undirected), and a source vertex s, find min-weight path $s \rightsquigarrow v$ for all vertices v.



Dijkstra(G, ℓ)

$S \leftarrow \{s\}$

$d(s) \leftarrow 0$

while $S \neq V$

choose $v \notin S$ to minimize $d'(v) = \min_{\substack{u \in S \\ (u,v) \in E}} d(u) + \ell(u,v)$

$S \leftarrow S \cup \{v\}$

$d(v) = d'(v)$

$Q \leftarrow \emptyset$

$S \leftarrow \{s\}$ solved verts

$d[s] \leftarrow 0$ cost of min-cost path from s

$\pi[s] \leftarrow$ NIL   next-to-last vert on min-cost path

for $v \in V, v \neq S$

if $(s,v) \in E$

$d'[v] \leftarrow \ell(s,v)$ best cost so far

$\pi[v] \leftarrow s$ next-to-last

else

$d'[v] \leftarrow \infty$

$\pi[v] \leftarrow$ NIL

Q.enqueue$(v, d'[v])$

n iterations   while $Q \neq \emptyset$ Q = unsolved verts

1 / iteration v = Q.extractMin()

$d[v] \leftarrow d'[v]$

$S \leftarrow S \cup \{v\}$

m iterations total   for $(v,w) \in E$ where $w \in Q$

if $d[v] + \ell(v,w) < d'[w]$ $\longleftarrow$ is path $s \rightsquigarrow v \rightarrow w$ better than $s \rightsquigarrow \pi[w] \rightarrow w$?

$d'[w] = d[v] + \ell(v,w)$

$\pi[w] = v$

1/inner iteration   Q.decreasePriority$(w, d'[w])$

Dijkstra's Running Time

Priority Queue Implementation

	binary heap	array	Fib heap
1 extractMin	$O(\log n)$	$O(n)$	$O(\log n)$ amortized
n extractMins	$O(n \log n)$	$O(n^2)$	$O(n \log n)$
1 decreasePriority	$O(\log n)$	$O(1)$	$O(1)$ amortized
m decreasePriorities	$O(m \log n)$	$O(m)$	$O(m)$
TOTAL (assuming adj list)	$O(m \log n)$ (assuming $m \geq n-1$)	$O(n^2)$ (since $m \leq n^2$)	$O(m + n \log n)$
sparse if m is $\Theta(n)$	$O(n \log n)$	$O(n^2)$	$O(n \log n)$
dense m is $\Theta(n^2)$	$O(n^2 \log n)$	$O(n^2)$	$O(n^2)$