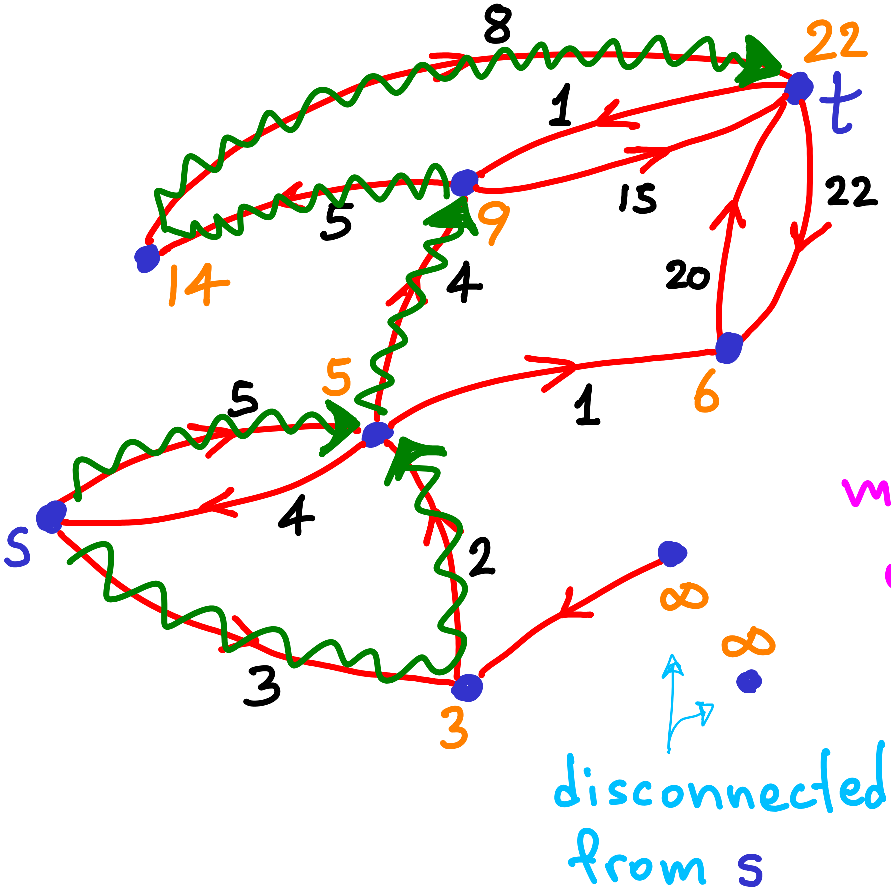
SINGLE $\underbrace{\text{SOURCE}}_{s}$ SHORTEST PATHS



paths from s to t

$3 + 2 + 1 + 20$

$3 + 2 + 4 + 15$

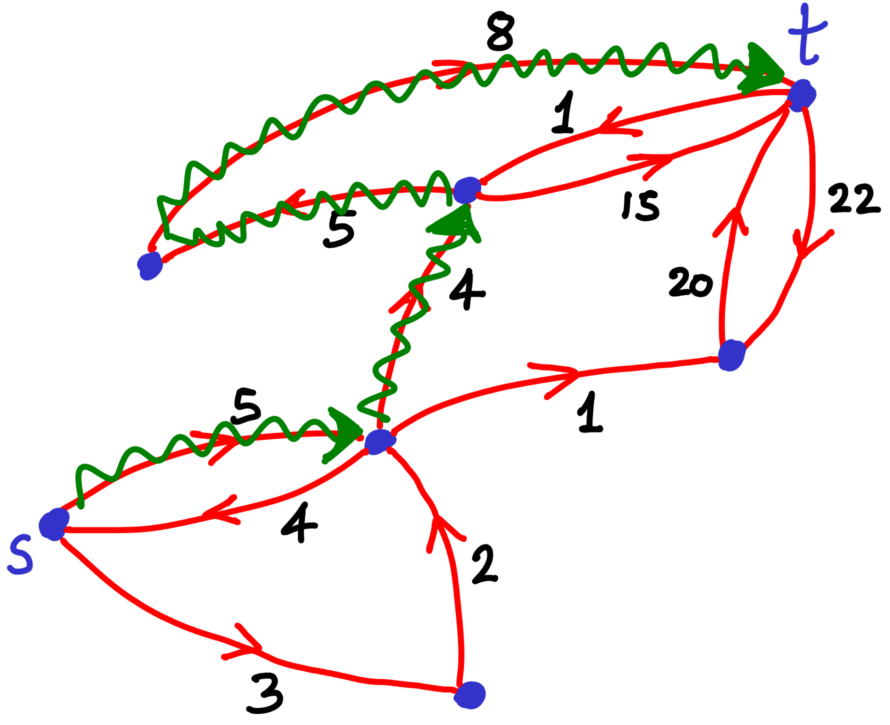
$\underline{3 + 2} + 4 + 5 + 8 = 22$

multiple options $\left\{ \begin{array}{l} \underline{5} + 1 + \cdots \\ \quad\; + 4 + \cdots = 22 \end{array} \right.$

not greedy or BFS

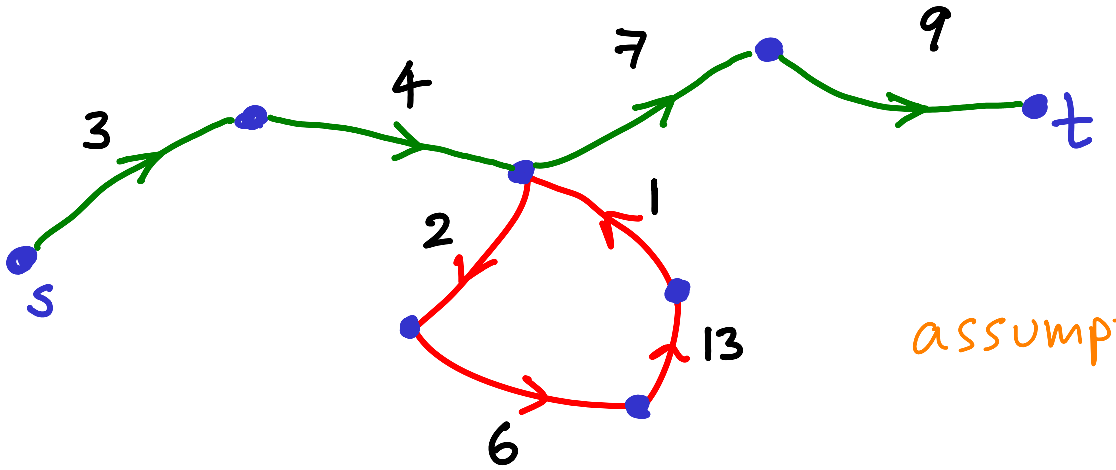
Generally assume a directed graph (can make undirected → directed easily)

SINGLE $\underbrace{\text{SOURCE}}_{s}$ SHORTEST PATHS



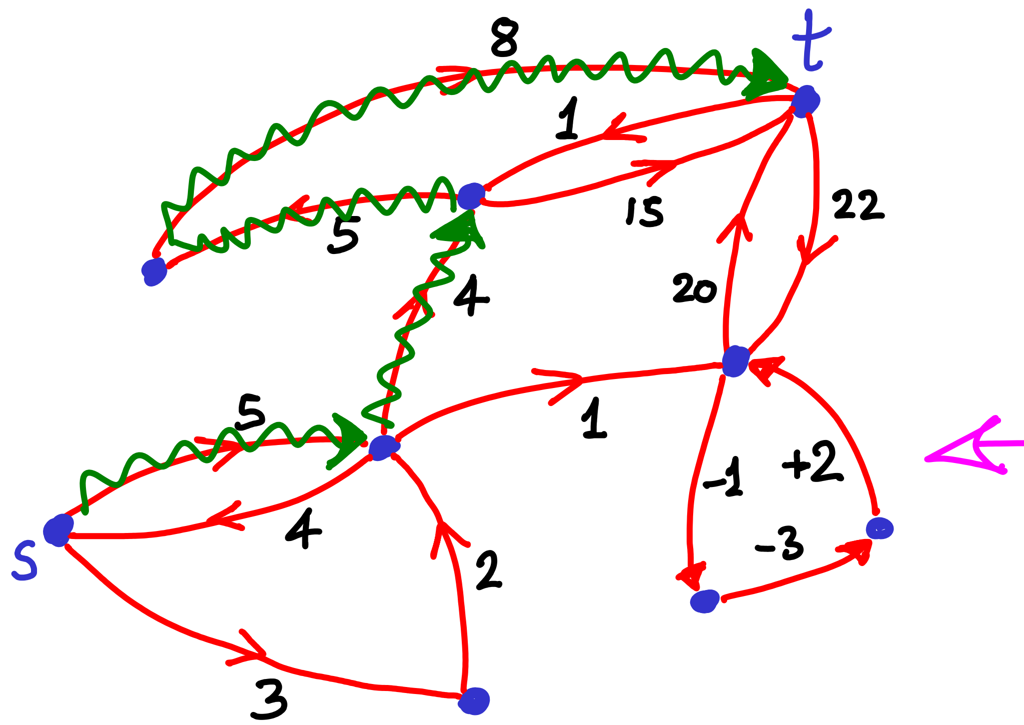
Observations

- No cycles in $s \to t$ (shortest path)



assumption?

SINGLE $\underbrace{\text{SOURCE}}_{s}$ SHORTEST PATHS

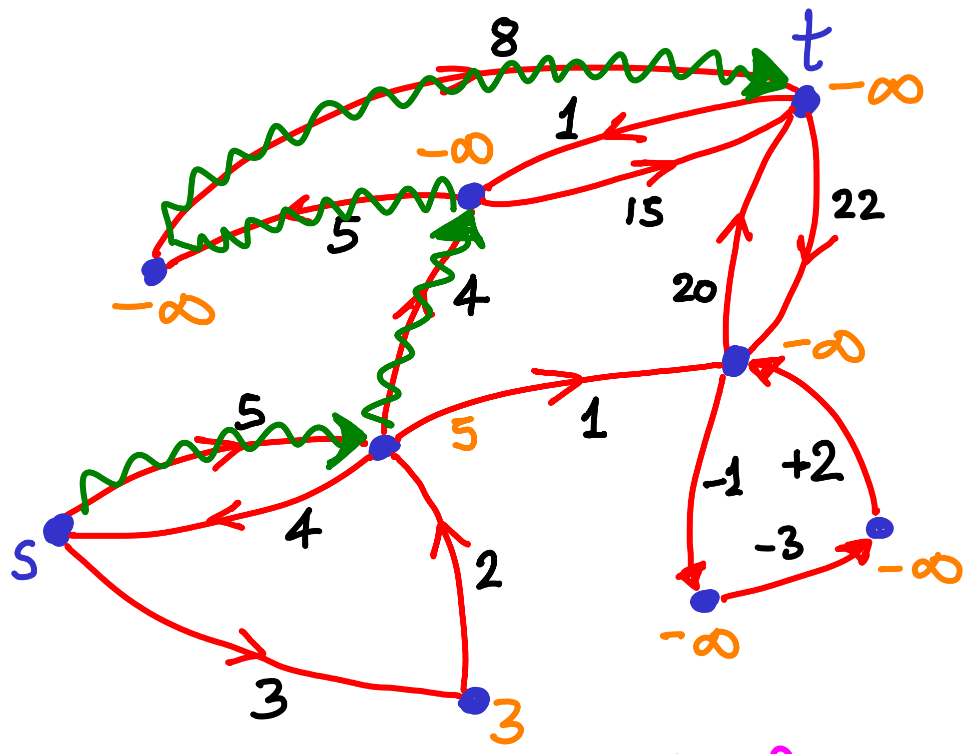


Observations

- No cycles in $s \rightarrow t$
- Negative weights ~OK, unless they form a negative cycle in G

$\Sigma(\text{cycle}) < 0$

SINGLE $\underbrace{\text{SOURCE}}_{s}$ SHORTEST PATHS

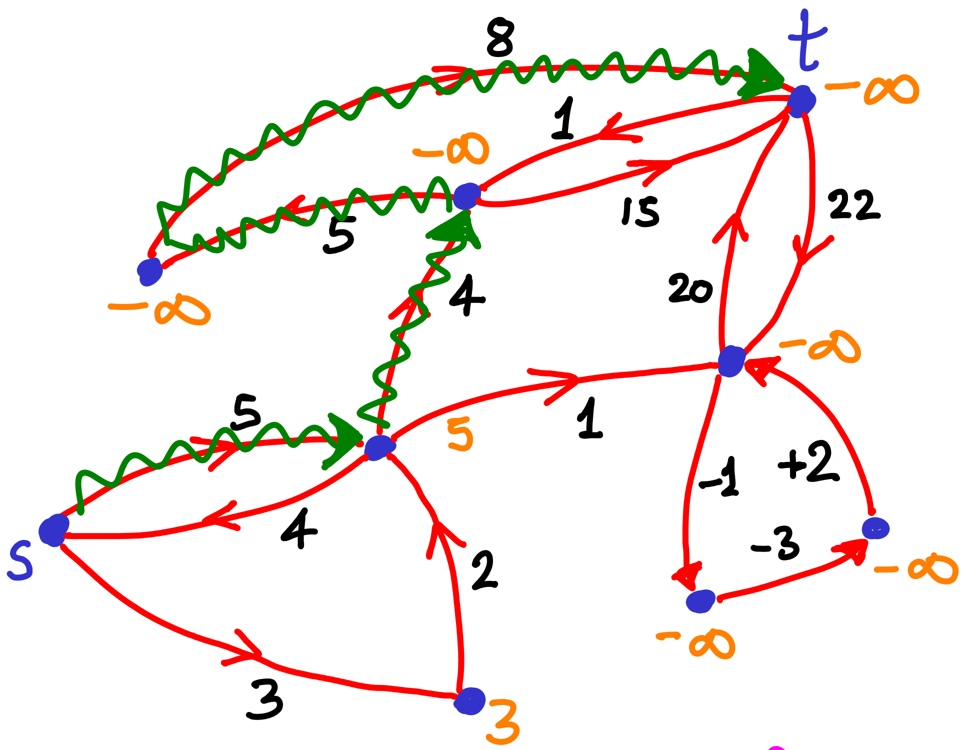


Observations

- No cycles in $s \rightarrow t$
- Negative weights ~OK, unless they form a negative cycle in G

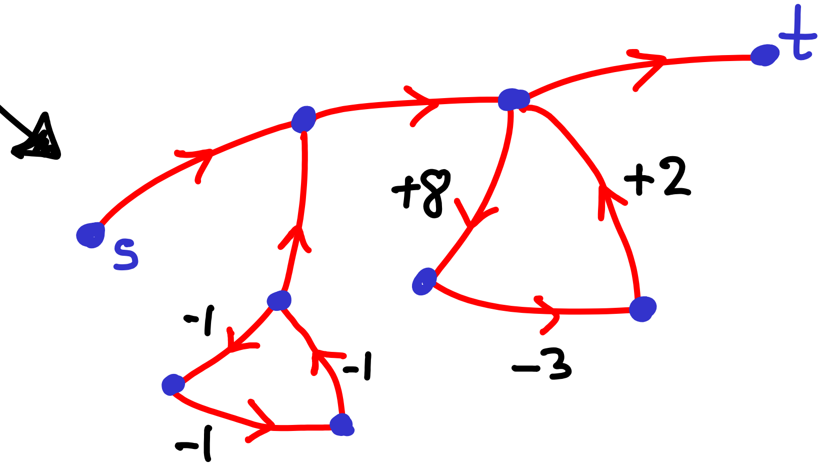
Any vertex reachable from a negative cycle gets a score of $-\infty$ } assuming cycle can be reached from s

SINGLE $\underbrace{\text{SOURCE}}_{s}$ SHORTEST PATHS



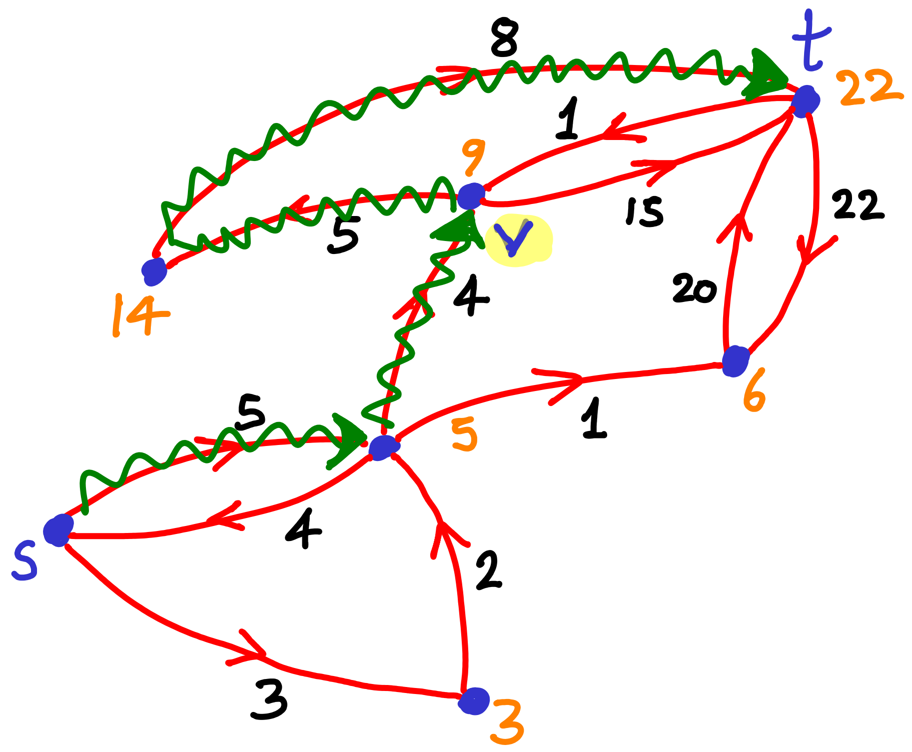
Observations

- No cycles in $s \rightarrow t$
- Negative weights ~OK, unless they form a negative cycle in G



Any vertex reachable from a negative cycle gets a score of $-\infty$

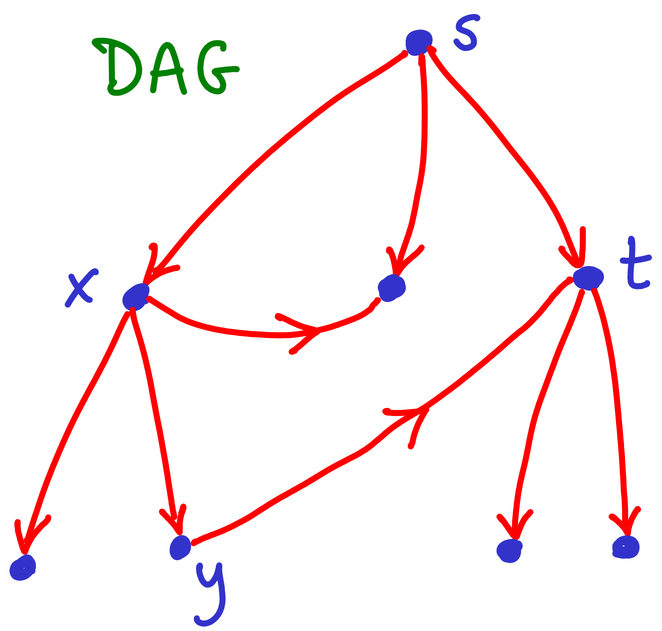
SINGLE $\underbrace{\text{SOURCE}}_{s}$ SHORTEST PATHS



Observations

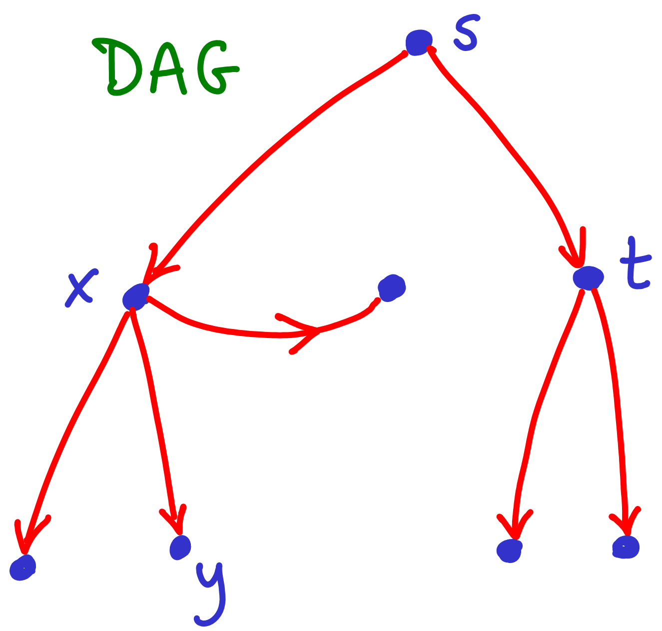
- No cycles in $s \to t$
- Negative weights ~OK, unless they form a negative cycle in G
- shortest path $s \to v \to t$ contains

  shortest path $s \to v$ (9)
  &
  shortest path $v \to t$ (13)



there may be multiple shortest paths
e.g. $s \to t$ or $s \to x \to y \to t$

All shortest paths from s to V
can be represented in a DAG



there may be multiple shortest paths

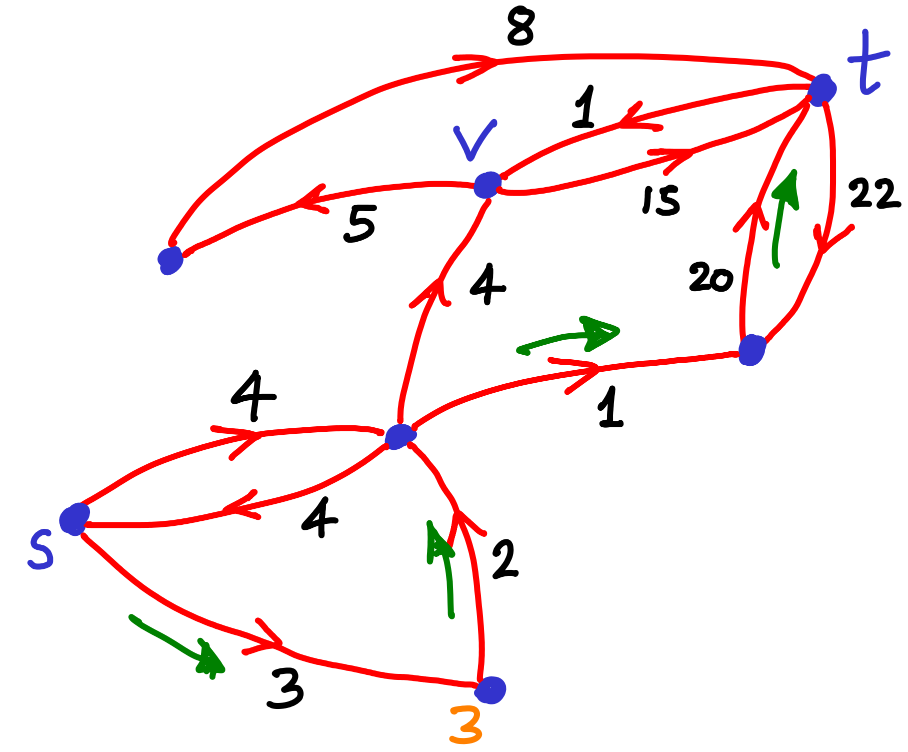
e.g. $s \to t$ or $s \to x \to y \to t$

All shortest paths from s to V can be represented in a DAG

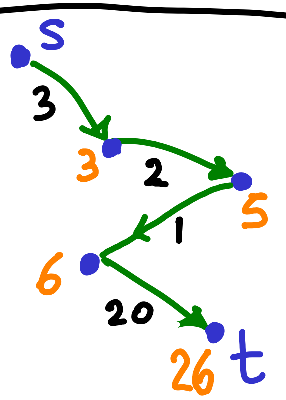
DAG → tree : arbitrarily keep one path to each vertex

"shortest paths tree"

(similar to picking one BFS/DFS search)



shortest paths tree



change tree

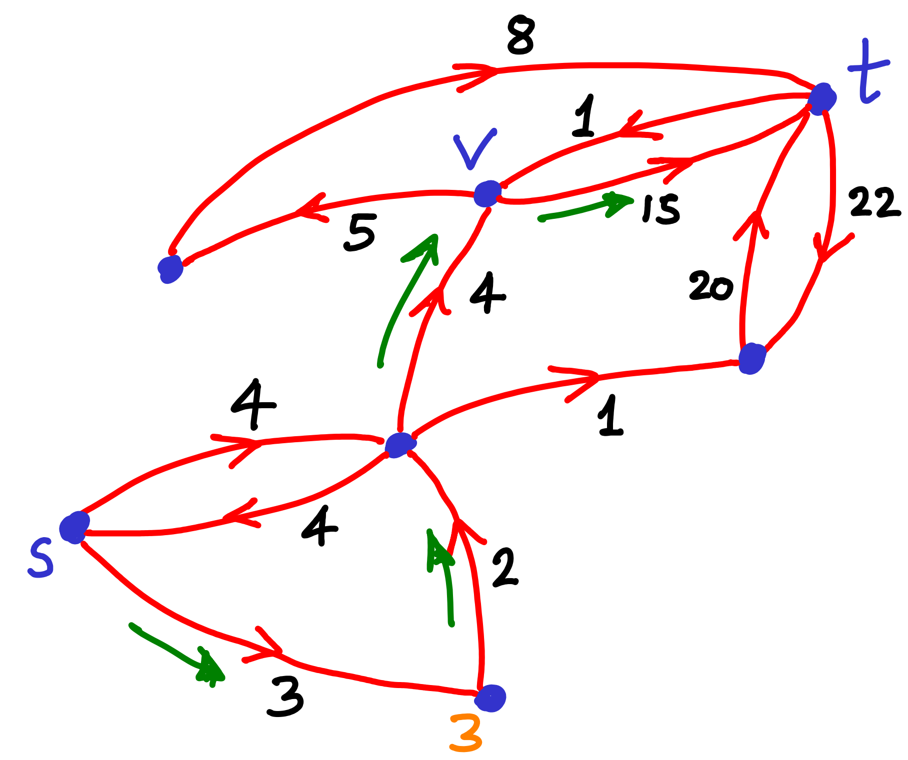
By exploring some path from s to t we get a path score (e.g. 26)

the score of t is 26, which may only decrease as we explore more options.

If we update the score of v : $d(v)$
& $\exists$ edge $v \to t$
then we can possibly improve $d(t)$:

$d(v) + w(v,t) < d(t)$?

(15)

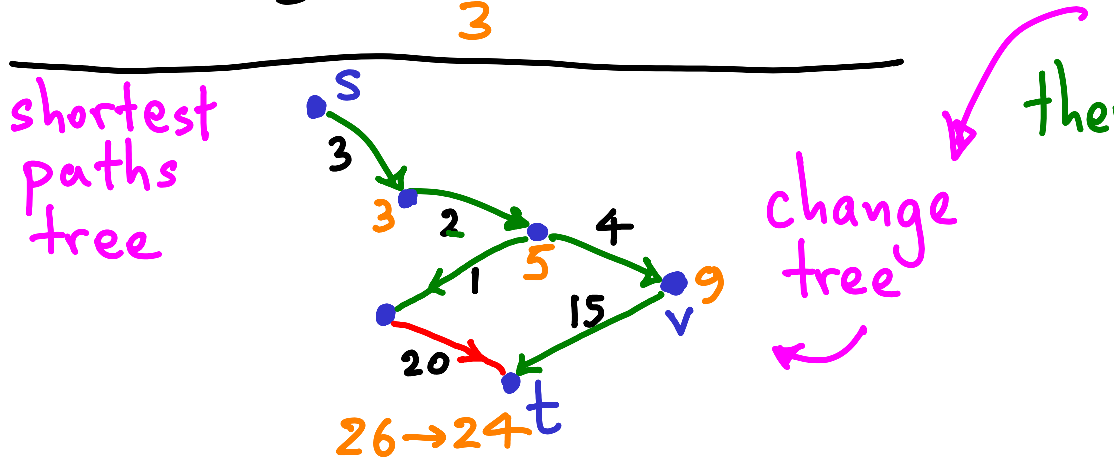


By exploring some path from s to t we get a path score (e.g. 26)

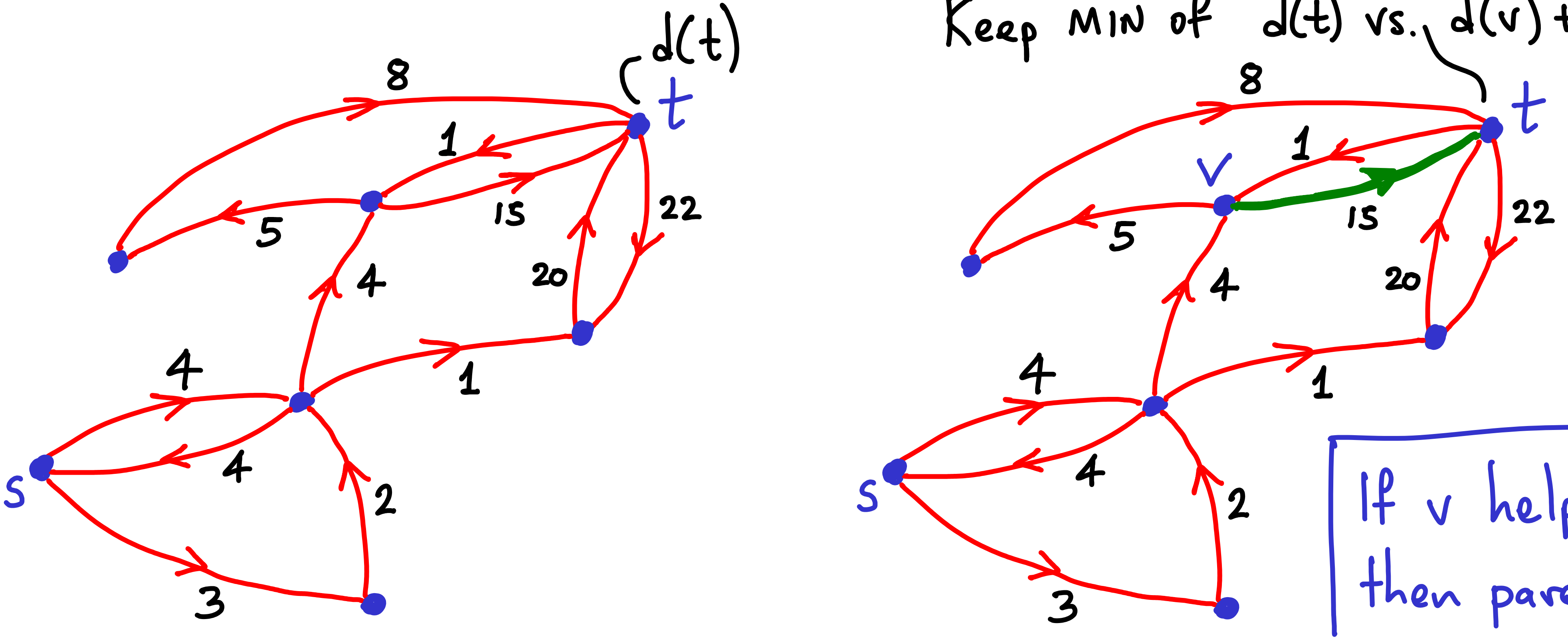
the score of t is 26, which may only decrease as we explore more options.

If we update the score of v: $d(v)$ & $\exists$ edge $v \to t$ then we can possibly improve $d(t)$:

$d(v) + w(v,t) < d(t)$?
(15)



Keep MIN of $d(t)$ vs. $d(v)+w(v,t)$

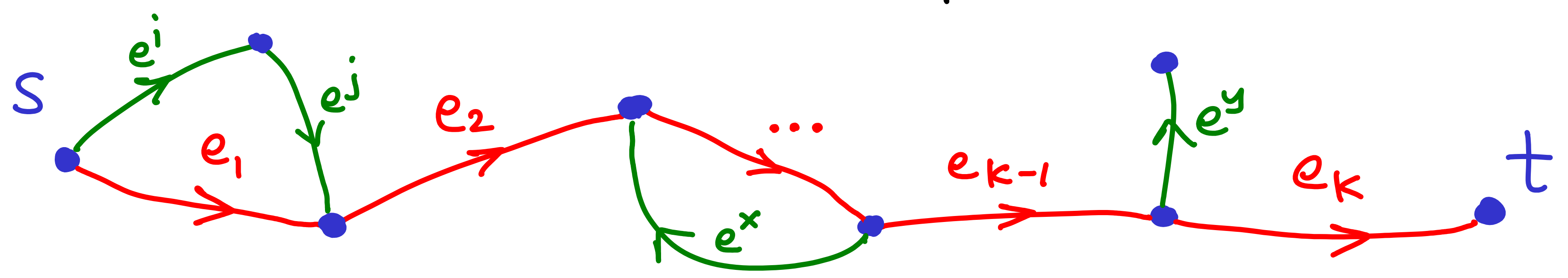


If v helps,
then parent(t) = v

Relax(v,t) : checking if score of t can be improved (lowered) by using $s \to v \to t$

Assume this is a shortest path from s to t

unknown but exists



Suppose we have an algorithm based on relaxing edges.

If we relax e_1 before e_2 before ... before e_{k-1} before e_k

then we will correctly compute $d(t)$ by INDUCTION

Relax sequence : $e^x\ e_1\ e^j\ e^y\ e_2\ e^x e^i\ e_k\ e_{k-1}\ e_1\ e^x\ e_k\ e^y$: OK

(don't care if we relax other edges or the same ones repeatedly)