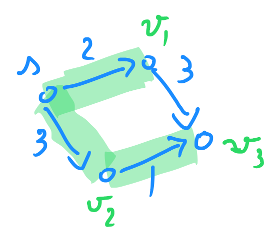
DIJKSTRA'S ALGORITHM ('60s)

WE WILL ASSUME THAT THE GRAPH IS "STRONGLY CONNECTED", THAT IS, ONE CAN GO FROM ANY NODE TO ANY OTHER NODE IN SOME NUMBER OF STEPS.



DIJKSTRA($G(V,E)$, ℓ, s):

$d(s) \leftarrow 0$ // $d(v)$ WILL CONTAIN THE LENGTH OF A SHORTEST PATH FROM s TO v, $\forall v$ (THE DISTANCE OF s TO v)

$P_s \leftarrow [s]$ // P_v WILL CONTAIN A SHORTEST PATH FROM s TO v, $\forall v$

$S \leftarrow \{s\}$ // S IS THE SET OF NODES VISITED SO FAR

WHILE $S \neq V$:

$T \leftarrow \{ w \mid w \in V-S$ AND THERE EXISTS $v \in S$ S.T. $(v,w) \in E \}$ // THE "FRONTIER" NODES

$\forall w \in T$, LET $d'(w) = \min_{\substack{(u,w)\in E \\ u \in S}} \left(d(u) + \ell(u,w) \right)$

LET $v \in T$ BE A NODE OF MINIMUM $d'(v)$

// v IS THE NODE THAT WE WILL VISIT IN THIS ITER.

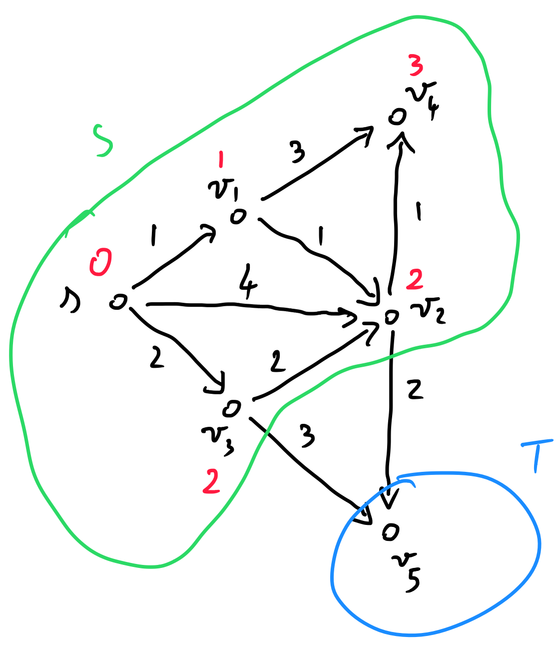
LET $u \in S$ BE SUCH THAT $d'(v) = d(u) + \ell(u,v)$.

$d(v) \leftarrow d'(v)$

$P_v \leftarrow P_u + [v]$

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

RETURN d (AND, IF ONE WANTS, $P_{v_1}, P_{v_2}, \ldots, P_{v_n}$)



$d(s) = 0$

$P_s = [s]$

$S = \{s\}$

$d(v_1) = d(s) + 1 = 1$

$P_{v_1} = P_s + [v_1] = [s, v_1]$

$S = \{s, v_1\}$

$d(v_3) = d(s) + 2 = 2$

$P_{v_3} = P_s + [v_3] = [s, v_3]$

$S = \{s, v_1, v_3\}$

$d(v_2) = d(v_1) + 1 = 2$

$P_{v_2} = P_{v_1} + [v_2] = [s, v_1, v_2]$

$S = \{s, v_1, v_2, v_3\}$

$d(v_4) = d(v_2) + 1 = 3$

$P_{v_4} = P_{v_2} + [v_4] = [s, v_1, v_2, v_4]$

$S = \{s, v_1, v_2, v_3, v_4\}$

$d(v_5) = d(v_2) + 2 = 4$

$P_{v_5} = P_{v_2} + [v_5] = [s, v_1, v_5]$

$S = \{s, v_1, v_2, v_3, v_4, v_5\}$

(RECALL THAT WE ASSUME THAT THERE EXISTS A PATH FROM EACH NODE TO EACH OTHER NODE)

L: AT ANY POINT DURING THE EXECUTION OF THE ALGORITHM, IF $u \in S$ THEN P_u IS A SHORTEST PATH FROM s TO u, AND THE LENGTH OF P_u IS $d(u)$.

P: WE PROVE THE CLAIM BY IND. ON $|S|$.

IF $|S| = 1$, THEN $S = \{s\}$, $d(s) = 0$ AND THE SHORTEST PATH FROM s TO s IS $P_s = [s]$.

NOW, ASSUME THAT THE CLAIM IS TRUE FOR $|S| = K$; WE PROVE IT FOR $K+1$.

CONSIDER THE ITERATION WHICH STARTS WITH $|S| = K$. IN THIS ITERATION THE ALGORITHM WILL SELECT SOME NODE $v \in V-S$; AND IT WILL ADD IT TO S, OBTAINING A NEW SET $S' = S \cup \{v\}$.

GIVEN THAT $v \notin S$, $|S'| = K+1$.

LET $u \in S$ BE THE NODE THAT THE ALGORITHM USED TO GET TO v. THEN, $P_v = P_u + [v]$ AND THE LENGTH OF P_v IS $d(u) + \ell(u,v)$.

BY THE INDUCTIVE HYPOTHESIS, SINCE $u \in S$, P_u IS A SHORTEST PATH FROM s TO u.

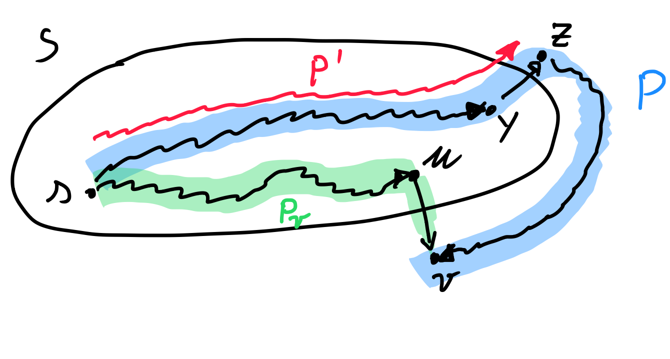
WE WANT TO PROVE THAT P_v IS A SHORTEST PATH FROM s TO v.

CLEARLY, P_v IS A PATH FROM s TO v.

LET US NOW CONSIDER ANY OTHER PATH P FROM s TO v — WE WILL SHOW THAT P IS NOT SHORTER THAN P_v.

SINCE $v \notin S$, AND P REACHES v FROM $s \in S$, P MUST LEAVE S AT SOME POINT.

LET z BE THE FIRST NODE OF P THAT IS NOT IN S, AND LET y BE THE NODE THAT COMES BEFORE z IN P.



P_v: $s \rightsquigarrow u \to v$

P: $s \rightsquigarrow y \to z \rightsquigarrow v$

WE WANT TO PROVE THAT P IS NOT SHORTER THAN P_v. WE WILL ARGUE BY CONSIDERING THE SUBPATH P' OF P THAT GOES FROM s TO z.

WE WILL SHOW THAT P' IS NOT SHORTER THAN P_v, AND THIS WILL CONCLUDE THE PROOF.

IN ITERATION $K+1$, THE ALGORITHM CONSIDERED ADDING z TO S, BUT —INSTEAD— IT ADDED v.

THUS, GIVEN THAT THE ALGORITHM CHOSE v, THERE EXISTS NO PATH FROM s TO z THAT IS SHORTER THAN P_v.

THEN, P' IS NOT SHORTER THAN P_v,
P IS NOT SHORTHER THAN P', THERE ARE NO EDGES OF NEGATIVE LENGTH

IT FOLLOWS THAT P IS NOT SHORTER THAN P_v. □