Analysis of Dijkstra's Algorithm

We provide additional details on the analysis of Dijkstra's algorithm given in the textbook.

At the end of each "while" loop, we call the nodes that have been removed from the queue *black*, and the nodes still in the queue *white*. We call a path from s to v an *all-black* path if all the vertices in the path, except possibly v, are black.

We will assume that all edge lengths are non-negative.

We first make the following simple observations:

- At all times, and for every vertex v, there is an all-black path from s to v of length $dist[v]$ (This is true at the beginning, and it is an invariant that is maintained every time we update $dist[.]$ of a vertex.)
- For every vertex v, the value $dist[v]$ can only be decreased or remain the same from step to step, and it is never increased. (This can be verified by inspecting the code.)
- If, at some point, for a certain vertex v, the value $dist[v]$ equals the length of the shortest path from s to v, then the value of $dist[v]$ is never changed in subsequent steps. (This is an immediate consequence of the previous two observations.)

We prove that the following invariant holds.

Lemma 1 *At the end of each iteration of the "while" loop:*

1. *All the black nodes v have a value of $dist[v]$ equal to the length of the shortest path from s to v, or ∞ if no such path exists.*
2. *All the nodes v have a value of $dist[v]$ equal to the length of the shortest all-black path from s to v.*

At the end of the execution of the algorithm, all nodes are black, and so either of the two invariants implies that the values of $dist[v]$ equal the length of the shortest path from s to v, for all v.

Base Case of Induction. Consider what happens after the first iteration of the while loop: the node s is taken out of the queue and is the only black node. We have $dist[s] = 0$ and indeed the distance of s to itself is 0 (because the "empty path" is a valid path of length 0, and there cannot be negative-length paths). This verifies the first condition. For the second condition, $dist[s] = 0$ is also the length of the shortest all-black path from s to itself; for vertices $v \neq s$, an all black path from s to v must be a one-edge path $s \to v$. If the edge (s, v) exists, then at the end of the first iteration we indeed have $dist[v] = \ell(s, v)$, which is the length of the all-black shortest path from s to v. If the edge (s, v) does not exists, then we have $dist[v] = \infty$ which is again correct because no all-black path exists from s to v.

Inductive Step Part (1). Now we argue that if both conditions are true after t iterations, then the first condition must be true after $t + 1$ iterations. Let v be the vertex that is removed from the queue during the $(t + 1)$-th iteration of the "while" loop. We just need to argue that $dist[v]$ is equal to the length of the shortest path from s to v.

The operations that we perform during this iteration do not modify the value of $dist[v]$, which remains what it was after the t-th iteration. We want to argue that, at time t, the two properties that we assume true after t iterations mean that v has a value $dist[v]$ equal to the actual shortest path length from s to v. Suppose, toward a contradiction, that there is a path from s to v of length $L < dist[v]$. Because of the second property, such path must use some white vertex as an intermediate step. Let u be the first white vertex we encounter along such path. Then the length of this path up to u is $\geq dist[u]$, because it is an all-black path from s to u and the second property is telling us that $dist[u]$ is the minimum length of such paths. Furthermore, the length of this path up to u is $\leq L$, so we conclude that

$$dist[u] \leq L < dist[v]$$

where all the value of $dist[]$ refer at the end of the t-th iteration. But this is a contradiction because v was chosen to be the white vertex of minimum $dist[.]$, for the values of $dist[.]$ that we got at the end of the t-th iteration.

Inductive Step Part (2). Now we argue that if both conditions are true after t iterations, then the second condition must be true after $t + 1$ iterations. As before, let us call v the vertex that is taken out of the queue during iteration $t + 1$. We will distinguish between the value $dist_t[x]$ of a vertex x at the end of the t-th iteration, and the value $dist_{t+1}[x]$ at the end of iteration $t + 1$.

If x is a vertex that was black at time t, then $dist_t[x]$ was the length of the shortest path from s to v, and we must have $dist_{t+1}[x] = dist_t[x]$. Furthermore, by the second property at time t we have that there is an all-black path from s to x of length $dist_t[x]$, so there is an all-black path from s to v of length $dist_{t+1}[x]$ at time $t+1$ (and there cannot be a shorter all-black path).

If we consider v, we argued above that $dist_{t+1}[v] = dist_t[v]$ and that $dist_{t+1}[v]$ is the length of a shortest path from s to v. By property 2 applied at time t, we have that there is an all-black path of length $dist_t[v]$ at time t, and so there is an all-black path from s to v of length $dist_{t+1}[v]$ at time $t+1$ (and there cannot be a shorter all-black path).

Finally consider a vertex y that is white at time $t+1$. Suppose toward a contradiction that there is an all-black path from s to y of length $L < dist_{t+1}[y]$. If the path does not contain v, then the path was already all-black at time t, and it has length $L < dist_{t+1}[y] \leq dist_t[y]$ which contradicts property 2 at step t. If the path contains v as last intermediate vertex, then the path has a $s \to v$ part of length $L - \ell(v,y)$ plus the edge $\ell(v,y)$. Note that we must have $dist_t[v] \leq L - \ell(v,y)$, because $dist_t[v]$ was already the length of a shortest path from s to v. Note, however, that the updates that we do at iteration $t+1$ imply $dist_{t+1}[y] \leq dist_t[v] + \ell(v,y)$, so we have

$$dist_t[v] + \ell(v,y) \leq L < dist_{t+1}[y] \leq dist_t[v] + \ell(v,y)$$

which is again a contradiction. Finally, if the path contains v as intermediate vertex, but not as the last intermediate vertex, let us call x the last intermediate vertex, so the path of length L is a path $s \to v$, followed by a path $v \to x$, followed by the edge (x,y). Let us call $t' \leq t$ the iteration in which x was taken out of the queue. First, we note that $L \geq dist_{t'}[x] + \ell(x,y)$, because the path of length L is made of a path from s to x, which must have length at least $d_{t'}[x]$, plus the edge (x,y). We also see that

$$dist_{t+1}[y] \leq dist_{t'}[y] \leq dist_{t'}[x] + \ell(x,y) \leq L$$

where the first inequality follows from the fact that $t' \leq t$, the second inequality follows from the updates that we do at step t', and the third inequality was observed above. We have now reached the contradiction $L < L$.