Problem 6 (4 points):
Consider the algorithm given in Figure 1 of Assignment 3. Show that for any flow problem with demands \( d_i = 1 \) for every \( i \) that has a feasible flow solution with paths of length at most \( L \) when using demands \( d_i' = (1 + \epsilon) \), it holds that the algorithm with sufficiently large queues never has to delete any flow.

Proof. Consider some fixed edge \( e \). The original Awerbuch-Leighton (AL) algorithm maximizes \( \sum_j f_j (\Delta_j(e) - f_j) \), which represents exactly the amount by which the potential at the queues at \( e \) drops when moving a normalized flow of \( f_j \) for each commodity \( j \). When using instead the rule in Figure ??, we have to distinguish between two cases for commodity \( i \) taken by the discrete AL-algorithm, where \( i \) is the value with maximum \( \Delta_i(e) \):

- \( \Delta_i(e) > 2 \): In this case, we can send a flow of 1, i.e. we can fully utilize the edge \( e \) with flow from commodity \( i \). So the potential drop is
  \[
  (\Delta_i(e) - 1) \geq (\sum_j f_j (\Delta_j(e) - f_j)) - 1
  \]
  where the \( f_j \) are chosen as in the original AL-Algorithm. The last inequality holds because \( i \) maximizes \( \Delta_i(e) \) and in whatever way the \( f_j \) are chosen,
  \[
  \sum_j f_j (\Delta_j(e) - f_j) \leq \Delta_i(e)
  \]
  because the \( f_j \) must fulfill \( \sum_j f_j \leq 1 \).

- \( \Delta_i(e)/d_i \leq 2 \): In this case, we can conclude from inequality (1) that also \( \sum_j f_j (\Delta_j(e) - f_j) \leq 2 \). Since the discrete AL-algorithm never increases the potential at an edge, this means that its potential drop is by at most 2 worse than the drop by the original AL-algorithm.

Combining the two cases, it follows that the discrete Awerbuch-Leighton Algorithm achieves a potential drop that is at most an additive 2 worse at any edge than the potential drop achieved by the original Awerbuch-Leighton Algorithm. Taking this into account, it follows from the proof in the lecture that the total potential drop due to the movement of flow in steps 2 and 3 of the AL-algorithm is at least

\[
\left( \sum_i (1 + \epsilon)q_i \right) - (1 + \epsilon)^2 L \cdot K - 2 \cdot 2m
\]
where $m$ is the number of edges. On the other hand, the potential increase caused by injecting new flow at step 1 of the AL-algorithm is at most

$$\sum_i \bar{q}_i.$$

Step 4 of the AL-algorithm can only decrease the potential. Hence, the overall potential increase in one round of the AL-algorithm is at most

$$-\epsilon \sum_i \bar{q}_i + (1 + \epsilon)^2 L \cdot K + 4m.$$

This value is guaranteed to be negative (i.e., the potential decreases) if

$$\sum_i \bar{q}_i > ((1 + \epsilon)^2 L \cdot K + 4m)/\epsilon.$$

Since flow is only sent downwards, it must hold that $\bar{q}_i$ is the maximum queue size for commodity $i$ in any queue of the system. Because there are $2m$ queues of each commodity in the system and according to (2), $\Phi$ increases by at most $(1 + \epsilon)^2 L \cdot K + 4m$ in any step, the potential is limited to

$$\Phi \leq 2m \cdot ((1 + \epsilon)^2 L \cdot K + 4m)/\epsilon)^2/2 + (1 + \epsilon)^2 L \cdot K + 4m.$$

In the worst case, all of this potential may be concentrated in a single queue. Hence, the maximum value a $\bar{q}_i(\epsilon)$ can attain is bounded by $2\sqrt{m} \cdot ((1 + \epsilon)^2 L \cdot K + 4m)/\epsilon$. 

Notice that the discrete AL-algorithm can also be used for arbitrary integral demands $d_i$, because we can simply pretend as if a commodity of demand $d_i$ represents $d_i$ commodities of demand 1, and then run (or better, simulate) the AL-algorithm as given in Figure 1 for this situation.