- **Decision Problem**

  The corresponding decision problem to the original optimization problem is, if given a directed graph representing the drainage system, is there a way of allocation to make the flow no less than k.

- **The problem is NP.**

  A certificate of this problem is a remaining graph G' after deleting some edges. For every vertex v in G' we check whether the total weight of edges to v equals that of edges from v. We also add up all flows from the starting vertex to obtain the maximum flow and compare it with k. The process takes polynomial time of roughly O($\mid V \mid\mid E \mid$).

- **The problem is NP-hard.**

  We find SUBSET-SUM known to be NP-complete.

  - **Reduction**

    Given a SUBSET-SUM instance of a set of numbers N and the goal k, we are able to derive a corresponding decision problem concerning a directed graph G representing the drainage system.

    The graph is made of three vertexes, v_1, v_2 and v_3. There is an edge between v_1 and v_2 that has capacity of k. We add $\mid N \mid$ edges between v_2 and v_3, with capacity of n_1, n_2, n_3, and so on, respectively. Let v_1 be the starting point while v_3 the ending point.

    We only consider SUBSET-SUM instances where non-negative integers are involved. If N contains some negative integers, we can transform it to N' by finding the smallest integer n_{min} and adding the absolute value of it to every number in N so that the smallest integer in N' is now 0 with $k' = k + \mid n_{min} \mid$.

    The reduction function runs in polynomial time O($\mid N \mid$).

  - **Claim 1: If (N, k) $\in$ SUBSET-SUM, then (G, k) $\in$ the problem.**

    If there exists some numbers that sum up to k in N, then we simply keep the corresponding edges and set the rest to have zero capacity.

    The solution is satisfactory because v_2 receives a flow of k while it gives off k as well. There is also k going out from the start vertex and k going to the end vertex. The instance is satisfactory since having exactly k satisfies the condition of no less than k.

  - **Claim 2: If (G, k) $\in$ the problem, then (N, k) $\in$ SUBSET-SUM.**

    If G after reduction is satisfactory, then the overall flow must be no less than k.

    In fact, the flow is exactly k. The only edge between v_1 and v_2 that each branch of water must run through has a flow of either 0 or k. The overall flow for any reduced graph G is thus either 0 or k. If G is satisfactory, then it must achieve a flow exactly k.

    The sum of capacities of edges between v_2 and v_3 equals the capacity of the edge between v_1 and v_2. Thus the original N must contains some numbers that sum up to k.

  Thus this problem is NP-hard.

- **The problem is NP-Complete.**

  Because the decision problem is both NP and NP-hard, it is NP-complete. The original optimization problem is no easier than the decision problem, so it is NP-complete as well.