HOMEWORK 3 – Q5

MINGLANG XIE

z5228006

5. You have to produce and deliver N chemicals. You need to deliver W_i kilograms of chemical C_i. Production of each chemical takes one day, and your factory can produce only one chemical at any time. However, all of the chemicals evaporate, and at the end of each day you loose p percent of the amount you had at the end of the previous day, so you need to produce more than what you need to deliver. Schedule the production of the chemicals so that the total extra weight of all chemicals needed to produce to compensate for the evaporation loss is as small as possible.

Solution:

First, for each chemical C_i we want to have W_i kilograms to be delivered after N days. We want to determine how much of a chemical is left after k many days for every chemical and record it. We get a list of sequence of $W_{(i,k)}$ for every chemical (e.g. $\begin{bmatrix} \langle W_{(0,0)}, W_{(0,1)}, \dots, W_{(0,N)} \rangle, \langle W_{(1,0)}, W_{(1,1)}, \dots, W_{(1,N)} \rangle, \\ \dots, \langle W_{(N,0)}, W_{(N,1)}, \dots, W_{(N,N)} \rangle \end{bmatrix}$),

where $W_{(i,k)} = \frac{W_i}{p^k}$ means the amount of weights for C_i left after k days (e.g. We have to produce and deliver N chemicals, and Production of each chemical takes one day. At day k, we are producing C_i, the amount we want to produce is $\frac{W_i}{p^{N-k}}$, where W_i is the number of amount we want to delivery, $N - k$ is the day left before we reach day N).

Then compute how much of C_i has been lost after k days by using $W_{(i,k)} = \sum_{k=0}^{N-k} W_{(i,k)} * p^k$, and we have: $\begin{bmatrix} \langle W_{(0,0)}, W_{(0,1)}, \dots, W_{(0,N)} \rangle, \\ \langle W_{(1,0)}, W_{(1,1)}, \dots, W_{(1,N)} \rangle, \\ \dots, \langle W_{(N,0)}, W_{(N,1)}, \dots, W_{(N,N)} \rangle \end{bmatrix}$ where $W_{(i,k)}$ means the amount of weights for C_i lost after k days.

Finally, we can apply greedy algorithm, which for each day, go through the list of sequence. For example, at day k, we go through the list, and we choose a chemical that has the least amount of weights lost after $N - k$ days. Therefore, we produce minimum amount of C_i at day k. (e.g. At day 0, we go through the list, and compare each chemical's weight after $N - 0$ days, which means we compare the last element of each list, and choose the chemical which has the least amount of weights lost after N days)

To prove optimality, we note that the optimal solution must make the total extra weight of all chemicals needed to produce to compensate for the evaporation loss is as small as possible, which means we want to have less loss evaporation for each produce. Assume there is a solution is better that this, than at day k, we need to choose other chemical as C_j except the chemical as C_i that have least amount of wights lost after $N-k$ days. For example, they choice to produce C_j at day k, but $W_{(j,k)}$ is not the smallest after $N-k$ days. Then we have $W_j - W_{(j,k)}$ amount of product left when we deliver the chemical, also we have to choose another day to produce C_i, which is also given a higher amount of wights. Therefore, this is violent the greedy algorithm, because we always want to choose the least amount of weight loss after $N-k$ days, in order to minimum the extra amount of wight that we need to produce.