

CSC373H1 - Assignment 3

Question 2

a) Define $w_i = 1$ if worker i is hired and 0 otherwise. The total cost is then $\sum_{i=1}^n c_i w_i$ and the program is

$$\begin{aligned} & \text{minimize} && \sum_{i=1}^n c_i w_i \\ & \text{s.t.} && \sum_{i: s_i \leq a < t_i} w_i \geq 1 \\ & && \sum_{i: s_i \leq t_j < t_i} w_i \geq 1, \forall j \text{ s.t. } a < t_j < b \\ & && w_i \in \{0, 1\}, \forall i = 1, \dots, n \end{aligned}$$

The first constraint enforces that at least 1 worker who delivers the food from position a is hired. The second constraint enforces that for every worker j who stops at/after a but before reaching b , regardless of whether they are hired or not, we must hire ≥ 1 other worker i who starts at/before t_j and who goes even closer towards b . The third constraint enforces that w_i is binary and non-negative.

On top of constraints 1 and 3, each worker has at most $n - 1$ versions of constraint 2, one for every other worker. This means $\leq n + 1$ constraints in total for each worker and $O(n^2)$ constraints overall, which is polynomial for n workers.

b) We prove a one-to-one correspondence between solutions of the original problem and the linear program. By proving this, we conclude that any minimum value obtained by the program must be the optimal solution for the problem.

Take any (not necessarily optimal) solution to the original problem. By assumption, there is a set of hired workers S s.t. $[a, b] \subseteq \bigcup_{i \in S} [s_i, t_i]$ (since they can transport the food from a to b). Set $w_i = 1$ for $i \in S$ and $w_i = 0$ for $i \notin S$. Note there must be a worker j who delivers the food out of a , i.e. $a \in [s_j, t_j)$, so constraint 1 is met. Now, for any worker j s.t. $t_j \in (a, b)$, hired or not, it must be that $t_j \in [s_i, t_i)$ for another hired worker i . If this were not so, t_j would lie in a subset of $[a, b]$ not covered by the stretch of any hired worker, i.e. $t_j \notin \bigcup_{i \in S} [s_i, t_i]$, which implies $t_j \notin (a, b)$ (a contradiction). Then, constraint 2 is met, so any feasible solution of the original problem can be converted into an instance of the linear program.

Take any instance of the linear program. By constraint 1, we hire a worker j who can deliver the food out of a , i.e. $w_j = 1$ and $a \in [s_j, t_j)$. For every worker j with $t_j \in (a, b)$, we hire ≥ 1 other worker who could pick up the food at t_j and deliver it to a point closer to b . This other worker must exist since constraint 2 guarantees a worker i s.t. $w_i = 1$ and $s_i \leq t_j < t_i$ for such a j . Thus, we are able to hire a subset of workers who can deliver the food from a to b since

- The first hired worker who delivers the food out of a is counted as a j in constraint 2, so another hired worker will pick the food up from them.
- For the last j in constraint 2, i.e. the j with the latest $t_j \in (a, b)$, we hire another worker i who picks the food up from them. Since $s_i \leq t_j < t_i$ and t_j is the latest $t \in (a, b)$, $t_i \notin (a, b)$,

meaning $s_i < b \leq t_i$. Thus, i can deliver the food to b , as required.

- Every hired worker with t in between the previous two workers satisfy constraint 2, so there will always be another worker who can pick the food up from them.

Then, there is a one-to-one correspondence between any instance of the linear program and feasible solution of the original problem. This means that minimizing the linear program yields the optimal solution for the problem.