## DETAILS OF ALGORITHM 2

### A. Explaination for Function $f$ and $g$

For tree $T(u)$, its optimal partitions $\tilde{T}(u)$ and $\hat{T}(u)$ should contain optimal partitions for all subtrees, i.e. optimal $\tilde{T}(v)$ or optimal $\hat{T}(v)$ for all $v \in H(u)$, where $H(u)$ represents the set of children of node $u$ in $T(\texttt{rt})$. We briefly discuss the kind of partitions children nodes can have based on whether the root is a city or a depot. If root $u$ is a city and is partitioned with $\tilde{T}$, then there exists a child $v$ whose subtree contains a depot connected to $u$. Thus, $u$ and $v$ are connected and have the same partition type $\tilde{T}$. In this case, node $v$ contributes $f(v)+c(u,v)$ to value function $f(u)$. Other children cannot connect to both $u$ as well as any depot in their subtrees at the same time. That is, each child either connects to $u$ with partition type $\hat{T}$ or does not connect to $u$ and has partition type $\tilde{T}$. Thus each child $v' \neq v$ contributes $\min\{f(v'), g(v')+c(u,v')\}$ to the value of $f(u)$. Next, in the case when the root $u$ is either a depot partitioned by $\tilde{T}$ or a city partitioned by $\hat{T}$, every child $v \in H(u)$ cannot connect to both $u$ and any depot in their subtrees at the same time, which contributes $\min\{f(v), g(v)+c(u,v)\}$ to the value of $f(u)$ or $g(u)$. Finally, value $g(u) = +\infty$ for every depot $u$ as we mentioned above.

### B. Proof for Theorem 1

*Lemma 1:* Given the input as in Theorem 1, the assignment of partition type for each node based on the output of Algorithm 2 is consistent with the resulting partition.

*Proof:* We prove the lemma by contradiction. Assume $u$ is the node with an inconsistent partition type and the minimum subtree.

If $u$ is a leaf node in $T(\texttt{rt})$, then $f(u) = +\infty$ for $u \in C$ and $g(u) = +\infty$ for $u \in D$. The former results in type $\hat{T}$ and the latter results in type $\tilde{T}$, matching the resulting partition in both cases. So $u$ cannot be a leaf node, and by minimum assumption, all its children should have consistent types.

If $u$ is assigned type $\tilde{T}$ but does not connect to any depot in the subtree, then $u \in C$. By Eq. (1), there exists a child $v$ of $u$ connect to $u$ and assigned type $\tilde{T}$, which means $v$ also has an inconsistent type, contradicting the minimum assumption.

If $u$ is assigned type $\tilde{T}$ but connects to more than one depot in the subtree, then $u$ can be either a city or a depot. If $u$ is a city, then by Eq. (1), the reconstruction only assigns one such $v$ to connect to $u$ and has type $\tilde{T}$. So the other child connects to $u$ and a depot in its subtree has the wrong type $\hat{T}$, contradicting the minimum assumption. If $u$ is a depot, then all children cannot both connect to $u$ and be assigned $\tilde{T}$, which means the child connects to it and a depot in the subtree has the wrong type $\hat{T}$, contradicting the minimum assumption.

If $u$ is assigned type $\hat{T}$, then $u \in C$ by the assignment rule. If $u$ connects to any depot in the subtree, then by Eq. (2), every child either connects to $u$ or is assigned type $\tilde{T}$. The child connecting to $v$ and a depot in its subtree has the wrong type $\hat{T}$, contradicting the minimum assumption. ■

*Lemma 2:* Given the input as in Theorem 1, the reconstruction based on the value function assignment in Eq. (1)-(2) produce the minimum $\tilde{T}(\texttt{rt})$ and $\hat{T}(\texttt{rt})$.

*Proof:* We prove the lemma by induction. Let $N = m+n$, where $m$ is the number of depots and $n$ is the number of cities. When $N = 1$, either there is no city to visit or there is no depot for a salesman to start from and so the optimality holds trivially. Suppose the algorithm gives an optimal solution for all $N \in \{1, 2, \cdots, m+n-1\}$. Now, for the case when $N = m+n$, assume that there exists a partition with smaller total weights. We prove that this assumption leads to a contradiction and hence the statement holds for the case when $N = m+n$. Let $v$ be the root node of the smallest subtree that $v$ has the same partition type in both the optimal partition and our partition, but the optimal partition gives a smaller total weight of the remaining edges. Such node $v$ exists because the root is given the same partition type and, by assumption, does not minimize the weights. Let $W_e$ denote the total weights of remaining edges in the subtree rooted at $v$ under our partition and $W_o$ of that under the optimal partition. By assumption $W_o < W_e$. Now we construct a partition for the subtree rooted at $v$ with total weights no larger than $W_o$ and no smaller than $W_e$ to get the contradiction. First, assign all child nodes of $v$ the same partition type as in the optimal partition and apply our reconstruction rule to get partitions for their subtrees. Then connect $v$ and its children in the same way as the optimal partition. The new partition is correct because both the connections between $v$ and its children and its children's partition types are the same as the optimal partition. Let $W$ denote the total weights of the new partition. By the induction hypothesis, all subtrees rooted at nodes in $C(v)$ are optimal, which means the total weight of remaining edges in the subtree rooted at $v$ is no larger than $W_o$, i.e. $W \leq W_o$. On the other hand, by Eq. (1)-(2), our partition selects the optimal way of connecting $v$ and its children, which means $W_e \leq W$. So, we have $W_e \leq W \leq W_o < W_e$, a contradiction. Hence, the statement holds for $N = m+n$, and by induction, it holds for all $N$. ■

*Proof:* [Proof for Thm. 1] By lemma 1, all node has consistent partition type. If the partition is incorrect, then there exists a connected component with either more than one depot or zero depots. In both cases, the root of the component has an inconsistent partition type, contradicting lemma 1. So the partition is correct. By lemma 2, $\tilde{T}(\texttt{rt})$ is the optimal partition. ■