Solution to Problem 3 from Homework 1

Problem Statement

Show that the neighborhood defined in Example 1.5 for the MST problem is exact.

Comments Before Solution:

Before you even start, you need to know what the MST problem is (See Example 1.2 on page 5). Given an integer \( n > 0 \) (also denoted as \( n \in \mathbb{Z}^+ \) in your text), assume we have an \( n \times n \) symmetric distance matrix \( d = \{d_{ij}\} \), \( d_{ij} \in \mathbb{Z}^+ \) (for all entries \( d_{ij} \) of the symmetric distance matrix). The problem is to find a spanning tree (a connected acyclic undirected subgraph of a graph \( G \) with the same vertex set as \( G \)): one where all edges are in \( \mathbb{Z}^+ \), on the \( n \) vertices of a graph \( G \) that has a minimal total length of its edges. So our feasible set \( F \) for a graph \( G = (V, E) \) is the set of all spanning trees \( F \) where \( V(G) = n \) and the cost function is \( c : (V, E) \rightarrow \sum_{i,j \in E(G)} d_{ij} \).

Expanded Problem Statement

Show that the neighborhood defined in Example 1.5 for the MST is exact; e.g. if \( f \) is a spanning tree of a graph \( G \) with \( n \) vertices that is a local optimum, we can define the neighborhood \( N(f) \) as the set

\[
\{g : g \in F \text{ and } g \text{ can be obtained from } f \text{ as follows}\}:
\]

add an edge \( e \) to the tree \( f \), producing a cycle,

and then delete any edge on the cycle.

You can’t do this problem unless you know what exact means and what \( N(f) \) is—this was a big problem in a lot of your solutions.

Solution: Assume for contradiction that we have a spanning tree \( f \) of a graph \( G \) with satisfying \( V(f) = V(G) \), that is a local optimum in the neighborhood \( N(f) \), \( f \in F \), but not a global optimum. Let \( T \) be a globally optimal spanning tree with the maximum possible edges in common with \( f \) (we may assume this as we are given in the problem statement that \( G \) is a finite graph, as it has \( n \) vertices). Let \( e \in E(G) \) be an edge with minimum weight from set of edges that are in \( T \) but not in \( f \) (again, we can assume such an edge exists as \( G \) is a finite graph). Removing \( e \) from \( T \) splits \( T \) into two smaller trees \( T_1 \) and \( T_2 \).

We can under our assumptions construct a new neighbor \( T' \) of \( f \) by (1) adding an edge \( e \) to \( f \) resulting in a cycle \( c \). This cycle must contain an edge \( e' \neq e \) connecting a vertex in \( T_1 \) with a vertex in \( T_2 \), where \( T_1 \) and \( T_2 \) are two subtrees of \( T \). We finish making \( T' \) by (2) connecting \( T_1 \) and \( T_2 \) with the edge \( e' \).

As \( f \) is a local optimum, we know that the cost of \( T' \) is at least the cost of \( f \). Hence in the symmetric distance matrix, \( d(e) \geq d(e') \). Yet we can connect \( T_1 \) and \( T_2 \) by adding the edge \( e' \) to create a spanning tree that is of at least as minimal as \( T \) and has at least one more edge in common with \( f \). This is a contradiction, because all the entries of the matrix \( d \) are positive integers, in other words in \( \mathbb{Z}^+ \). Hence \( N(f) \) is exact.