Steiner Tree NP-completeness Proof

Alessandro Santuari

May 7, 2003

Abstract

This document is an exercise for the Computational Complexity course taken at the University of Trento. We propose an NP-completeness proof for the Steiner Tree problem in graphs.

1 The Steiner tree problem (ST)

The Steiner tree problem in graphs, called for brevity ST, is defined in decisional form as follows:

INSTANCE:

- an undirected graph $G = (V, E)$;
- a subset of the vertices $R \subseteq V$, called *terminal nodes*;
- a number $k \in \mathbb{N}$.

QUESTION:
is there a subtree of G that includes all the vertices of R (i.e. a spanning tree for R) and that contains at most k edges?

This problem has many applications, especially when we have to plan a connecting structure among different terminal points. For example, when we want to find an optimal way to build roads and railways to connect a set of cities, or decide routing policies over the internet for multicast traffic, usually from a source to many destinations.

Unfortunately, this problem has shown to be "intractable", in the sense that there exists no polynomial algorithm to solve it, unless P = NP. However, many approximate algorithms have been developed by researchers, see for example [2, 3, 4].

The goal of this exercise is to propose an NP-completeness proof for the Steiner tree problem by transforming another known NP-complete problem to it. In short, we will obtain our own proof to the following classical result.

Theorem: Steiner tree problem in graphs is NP-complete.

The proof will follow step by step the template advised by Garey and Johnson [1] to show that a problem Π is NP-complete:

1. show than Π is in NP;
2. select a known NP-complete problem Π';
3. construct a transformation f from Π' to Π;
4. prove that f is a polynomial transformation.

2 Steiner Tree is in NP

So firstly we want to be sure that the ST problem is actually in NP. Assume $\langle G, R, k\rangle \in ST$, that is, assume the instance $\langle G, R, k\rangle$ reserves a *yes* answer. In this case, given an hypothetic positive solution $T \subseteq G$, we can check in polynomial time that:

- T is really a tree: it contains no cycles and it is connected;
- the tree T touches all the terminals specified by the set R;
- the number of edges used by the tree is no more than k.

We can now proceed to the next step: select an (appropriate) known NP-complete problem for the reduction.

3 Exact Cover by 3-Sets (X3C)

The Exact Cover by 3-Sets problem seems to serve well for the task, besides X3C is well known and it is mentioned among the basic NP-complete problems in [1] as a generalization of the 3-Dimensional Matching (3DM) problem.

INSTANCE:

- a finite set X with $|X| = 3q$;
- a collection C of 3-element subsets of X, $C = \{C_1, \cdots, C_n\}$, $C_i \subseteq X, \quad |C_i| = 3 \quad 1 \leq i \leq n$;

QUESTION:

does C contain an *exact cover* for X, that is, a subcollection $C' \subseteq C$ such that every element of X occurs in exactly one member of C'?

Note that, where C' is a solution which certificate that $\langle X, C \rangle \in$ X3C then:

- the members of the solution C' form a partition of the set X;
- $|C'| = q$.

Now we are ready to move through the next step: construct a transformation function from X3C to ST.

4 Transform X3C to ST

In this section, we propose a reduction from X3C to ST giving a set of rules to build an instance of ST starting from a generic instance of X3C and we prove that such transformation is executable in polynomial time.

Given an instance of X3C, defined by the set $X = \{x_1, \cdots, x_{3q}\}$ and a collection of 3-element sets $C = \{C_1, \cdots, C_n\}$, we have to build the ST instance specifying the graph $G = (V, E)$, the set of terminals R, and the upper-bound on the spanning tree size k.

- define the set of vertices V as:

$$V(G) = \{v\} \cup \{c_1, \cdots, c_n\} \cup \{x_1, \cdots, x_{3q}\}.$$

basically, we put a new node v, a node for each member of C, and a node for each element of X.

- now define the set of edges:

$$E(G) = \{vc_1, \cdots, vc_n\} \cup \left(\bigcup_{x_j \in C_i} \{c_i x_j\} \right)$$

there is an edge from v to each node c_i, and an edge $c_i x_j$ if the element x_j appears into the set C_i of the X3C instance.

- the terminal nodes set $R \subseteq V$ is:

$$R = \{v, x_1, \cdots, x_{3q}\}$$

- set k equal to $4q$.

It's easy to see that the reduction from X3C to ST can be done in polynomial time. The graph constructed according to these rules is shown in Figure 1. The role of the new node v is to make sure that the graph we

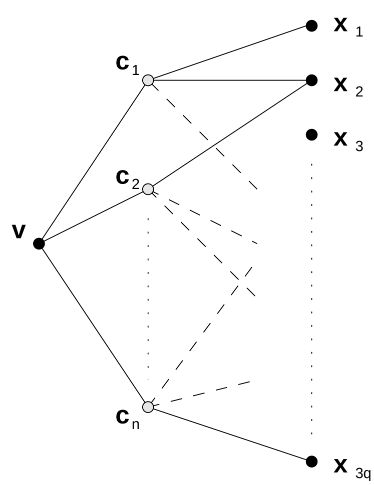

Figure 1: The graph constructed by the transformation process. The black nodes are those in the set R, the terminal nodes.

just generated is connected. Now we are going to prove that there exists a Steiner tree with no more than k edges if and only if there is an exact cover for the X3C instance of the problem.

Lemma: $\langle X, C \rangle$ belongs to X3C if and only if $\langle G, R, k \rangle$ belongs to ST.

Proof: We split the proof into two parts, one for each implication.

- X3C $\Rightarrow$ ST
  Suppose there is an exact 3-cover C' for the X3C problem. Clearly, C' uses exactly q subsets. Without loss of generality suppose they are $C_1, \cdots, C_q$ (if it's not the case, we just have to relabel them). Then the tree consisting of edges:

  - $vc_1, \cdots, vc_q$
  - $c_i x_j$, if $x_j \in C_i$ and $1 \leq i \leq q$

  is a Steiner tree solving the problem with $q + 3q = 4q = k$ edges. So, if there is an exact 3-cover, then there is a Steiner tree using no more than k edges.

- X3C $\Leftarrow$ ST
  Suppose now there exists a Steiner tree T with at most $4q$ edges. Since T is a tree, it has at most $4q + 1$ nodes. According to the definition of Steiner tree, T must also touch the terminal nodes $x_1, \cdots, x_{3q}$ and v, so T contains at most q c-nodes. But the degree of c-nodes (considering only the edges toward x-nodes) is 3, so it is impossible to hit all the $3q$ x-nodes if the tree contains less than $4q + 1$ nodes. We conclude that T has exactly $4q$ edges and contains exactly q c-nodes. Without loss of generality suppose these nodes are $c_1, \cdots, c_q$, then the solution C' of the X3C problem is given by the set

$$C' = \{C_1, \cdots, C_q\}.$$

This concludes the proof. □

References

[1] Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W.H. Freeman and Company, 1979.

[2] Clemens Gröpl, Stefan Hougardy, Till Nierhoff, and Hans Jürgen Prömel. Steiner Trees in Uniformly Quasi-Bipartite Graphs. November 30, 2001.

[3] Romeo Rizzi. On Rajagopalan and Vazirani's $\frac{3}{2}$-Approximation Bound for the Iterated 1-Steiner Heuristic. Accepted by *Information Processing Letters*.

[4] Gabriel Robins and Alexander Zelikovsky. Improved Steiner tree approximation in graphs. In *Symposium on Discrete Algorithms*, pages 770–779, 2000.