AN $O(n^2)$ ALGORITHM FOR FINDING THE COMPACT SETS OF A GRAPH

CHIOU-KUO LIANG
Department of Computer Science
Chung Hua Polytechnic Institute
Hsinchu, Taiwan 30067
Republic of China

Abstract.

A special clustering problem is discussed in this paper, called the compact set problem. The goal of the problem is to find all compact sets in a complete, weighted, undirected graph with $n$ vertices. A subset $C$ of vertices is called a compact set if $1 < |C| < n$ and the maximum weight among all edges in $C$ is smaller than the minimum weight among all edges connecting one vertex in $C$ and the other vertex not in $C$. An algorithm with complexity $O(n^2)$ is given for the problem improving the previous results.

CR category: F.2.2.

Keywords: Graph algorithms, minimum spanning trees, compact sets.

1. Introduction.

In this paper we consider a special clustering problem called the compact set problem [6] defined as follows. Let $G = (V, E)$ be a complete, weighted and undirected graph with vertex set $V = \{1, 2, \ldots, n\}$ and edge set $E$. Each edge in $E$ is associated with a nonnegative weight. Let $(u, v)$ denote an edge in $E$ between vertices $u$ and $v$. The weight of $(u, v)$ is denoted $w(u, v)$. We call a subset $C$ of $V$ compact if it satisfies the following conditions:

1. $1 < |C| < n$, and
2. $\max\{w(u, v) | u, v \in C\} < \min\{w(a, b) | a \in C, b \in V - C\}$.

The problem is to find all compact sets in the graph $G$ (if any). Figure 1 shows an example of a graph and the corresponding compact sets.

* This research was partially supported by the National Science Council of the Republic of China under Grant NSC81-0408-E-216-502.

Figure 1. The compact sets of the graph are \{1, 3\}, \{4, 6\} and \{1, 2, 3, 5\}.

The difficulty of the problem lies in the requirement to generate all existing compact sets since we have to check all proper subsets of vertices to see if they are compact or not. A brute force approach will hence result in an exponentially running time. In [6], the author proposed an $O(n^2 \log n)$ algorithm to find all compact sets. In this paper, we propose an algorithm which solves the problem in $O(n^2)$ time. Our algorithm is based on the properties of the minimum spanning tre of the graph. We also show that the number of compact sets is at most $n - 2$.

2. Preliminaries.

Let $C$ be a subset of $V$ and $\overline{C} = V - C$. For an nonsingleton subset $C$ of $V$, let $\delta(C)$ denote its diameter, i.e.,

$$\delta(C) = \max\{w(u, v) | u \in C, v \in C\}.$$ 

Suppose that $A$ and $B$ are disjoint subsets of $V$. The distance between $A$ and $B$, denoted $d(A, B)$, is defined by

$$d(A, B) = \min\{w(a, b) | a \in A, b \in B\}.$$ 

Referring to the notation above, we say that $C$ is a compact set if $1 < |C| < n$ and $\delta(C) < d(C, \overline{C})$. For convenience, we say that a vertex is a $C$-vertex (non-$C$-vertex) if it is in $C$ (not in $C$).

Let $T$ be a minimum spanning tree of $G$ and $C$ a subset of $V$. We define an auxiliary graph $G_T(C) = (C, E_C)$ on the vertex set $C$ as follows: $(u, v)$ is an edge in $E_C$ if and only if $(u, v)$ is in $T$.

**Theorem 1.** Let $C$ be a subset of $V$. If $C$ is a compact set, then $G_T(C)$ is connected.

**Proof.** Suppose that $G_T(C)$ is not connected. Then there exist at least two