Volume: 04 Issue: 06 Pages:1809-1811 (2013) ISSN : 0975-0290

Algorithms to find vertex-to-clique Center in a Graph using BC-representation

A. Ashok Kumar
Department of Computer Science, Alagappa Government Arts College, Karaikudi − 630003, India.
Email:ashokjuno@rediffmail.com
S. Athisayanathan
Research Department of Mathematics, St. Xavier's College (Autonomous), Palayamkottai − 627002, India.
Email:athisayanathan@yahoo.co.in and
A. Antonysamy
Ananda College, Devakottai, India
Email: antonysamy.fr@gmail.com

---ABSTRACT---
In this paper, we introduce algorithms to find the vertex-to-clique (or (V, ζ))-distance $d(v, C)$ between a vertex v and a clique C in a graph G, (V, ζ)-eccentricity $e_1(v)$ of a vertex v, and (V, ζ)-center $Z_1(G)$ of a graph G using BC-representation. Moreover, the algorithms are proved for their correctness and analyzed for their time complexity.

Keywords - **clique, distance, eccentricity, radius, center, binary count.**

Date of Submission: March 10, 2013 Date of Acceptance: June 05, 2013

1 Introduction

By a graph $G = (V,E)$ we mean a finite undirected connected simple graph. $|V|$ and $|E|$ denote the order and size of a graph G respectively. A clique of a graph G is a maximal complete subgraph of G. For other basic definitions not mentioned in this paper, we refer [2, 3].

For vertices u and v in a graph G, the distance $d(u, v)$ between u and v is the length of a shortest $u - v$ path. For subsets A and B of the vertex set V of G, the distance between A and B is defined as $d(A, B) = min\{d(x, y) : x \in A, y \in B\}$. For any vertex v of G, the *eccentricity* of v is $e(v) = max\{d(v, u) : u \in V\}$. The *radius* of G is $r = min\{e(v) : v \in V\}$. The *center* of G is $Z(G) = \{v \in V : e(v) = r\}$. A vertex in $Z(G)$ is called a *central vertex*. The distance matrix $D(G) = [d_{ij}]$ of G is a $n \times n$ matrix, where n is the order of G, and $d_{ij} = d(v_i, v_j)$ the distance between vertext v_i and the vertex v_j in G $(1 \leq i \leq n, 1 \leq j \leq n)$.

In [4] Santhakumaran and Arumugam introduced and studied the following central structures: Let G be a connected graph and $\zeta = \{C : C$ is a clique in $G\}$. For a vertex v and a clique C in G, the *vertex-to-clique* (or (V, ζ))-distance $d(v, C)$ between the vertex v and the clique C in G is defined as $d(v, C) = min\{d(v, u) : u \in C\}$. For a vertex v of G, (V, ζ)-*eccentricity* $e_1(v)$ of v is $e_1(v) = max\{d(v, C) : C \in \zeta\}$. The (V, ζ)-*radius* r_1 of G is $r_1 = min\{e_1(v) : v \in V\}$. The (V, ζ)-center of G is $Z_1(G) = \{v \in V : e_1(v) = r_1\}$. A vertex in $Z_1(G)$ is called a (V, ζ)-*central vertex*.

In [1] Ashok, Athisayanathan and Antonysamy introduced a method to represent a subset of a set which is called *binary count* (or BC) representation. That is, if $X = \{1, 2, 3, 4\}$ is a set, then the binary count (or BC) representation of the subsets $\{\Phi\}$, $\{1\}$, $\{2\}$, $\{3\}$, $\{4\}$, $\{1, 2\}$, $\{1, 3\}$, $\{1, 4\}$, $\{2, 3\}$, $\{2, 4\}$, $\{3, 4\}$, $\{1, 2, 3\}$, $\{1, 2, 4\}$, $\{1, 3, 4\}$, $\{2, 3, 4\}$, $\{1, 2, 3, 4\}$ of X are (0000), (1000), (0100), (0010), (0001), (1100), (1010), (1001), (0110), (0101), (0011), (1110), (1101), (1011), (0111), (1111) respectively. Using this BC representation, given a graph G with the vertex set $V = \{1, 2, 3, \ldots, n\}$ and a subset A of V, they introduced an algorithm to verify whether the subgraph $< A >$ induced by the set A in G is a clique or not. Moreover, a general algorithm is introduced to generate all cliques in G and proved the correctness of these algorithms and analyzed their time complexities.

Example 1.1 Consider the graph G given in Figure 1.1 with the vertex set $V = \{1, 2, 3, 4, 5\}$. Then the distance matrix $D(G)$ of G is

$$D(G) = \begin{pmatrix} 0 & 1 & 1 & 2 & 3 \\ 1 & 0 & 2 & 1 & 2 \\ 1 & 2 & 0 & 1 & 2 \\ 2 & 1 & 1 & 0 & 1 \\ 3 & 2 & 2 & 1 & 0 \end{pmatrix}$$

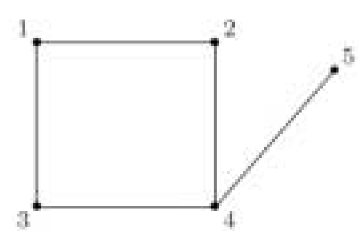

Figure 1.1: G

Moreover, the set of all cliques in graph G is $\zeta = \{\{1, 2\}, \{1, 3\}, \{2, 4\}, \{3, 4\}, \{4, 5\}\}$. Now using the algorithms discussed in [1], it is easy to verify that the set ζ of all cliques in G in BC representation is $\zeta = \{(11000), (10100), (01010), (00110), (00011)\}$. Note that if C is the clique $\{3, 4\}$, then the BC representation of C is $BC(C) = (00110)$, and further $BC(C(1)) = BC(C(2)) = BC(C(5)) = 0$, and $BC(C(3)) = BC(C(4)) = 1$. That is, $BC(C(i))$ $(1 \leq i \leq n)$ denotes the integer (1 or 0) in the i^{th} place in the BC representation of the clique C in the graph G.

Volume: 04 Issue: 06 Pages:1809-1811 (2013) ISSN : 0975-0290

In this paper we introduce algorithms to find (V, ζ)-distance, (V, ζ)-eccentricity and (V, ζ)-center in a connected graph G of order $n(> 1)$ using BC representation.

2 Vertex-to-Clique Center Algorithms

First, we introduce an algorithm to find the (V, ζ)-distance $d(i, C)$ between a vertex i and a clique C in a graph G using BC representation.

Algorithm 2.1 Let G be a graph with $V = \{1, 2, 3, . . . , n\}$ and $\zeta = \{C : C$ is a clique in BC representation in $G\}$.

1. Let $D(G) = [d_{ij}]$ be the distance matrix of graph G.
2. Let $i \in V$ and $C \in \zeta$.
3. if $BC(C(i)) = 1$ then $d(i, C) = 0$, goto step 9
4. for $j = 1$ to n
5. $d(i, j) = n$
6. if $BC(C(j)) = 1$ then $d(i, j) = d_{ij}$
7. next j
8. Find $d(i, C) = min\{d(i, j) : 1 \le j \le n\}$
9. return $d(i, C)$
10. stop

Theorem 2.2 For any vertex i and a clique C in a graph G, the Algorithm 2.1 finds the (V, ζ)-distance $d(i, C)$ from the vertex i to the clique C.

Proof. Let G be a graph with $V = \{1, 2, 3, . . . , n\}$, $\zeta = \{C : C$ is a clique in BC representation in $G\}$, and $D(G)$ the distance matrix of G. Let $i \in V$ and $C \in \zeta$. If the vertex i is a vertex of the clique C, then $BC(C(i)) = 1$ so that the (V, ζ)-distance $d(i, C) = 0$. If the vertex i is not a vertex of the clique C, then $BC(C(i)) = 0$, then the steps 4 to 6 of the Algorithm 2.1 find the distance $d(i, j)$ from the vertex i to the vertices $j(1 \le j \le n)$ of G as follows: If j is a vertex of the clique C then $BC(C(j)) = 1$ otherwise $BC(C(j)) = 0$. Hence $d(i, j) = n$ if $BC(C(j)) = 0$ and $d(i, j) = d_{ij}$ if $BC(C(j)) = 1 (1 \le j \le n)$. Then the step 8 of Algorithm 2.1 finds the (V, ζ)-distance $d(i, C) = min\{d(i, j) : 1 \le j \le n\}$ from the vertex i to the clique C.

Theorem 2.3 The distance between vertex i and a clique C in a graph G can be found in $O(n)$ time using Algorithm 2.1.

Proof. It follows from the fact that the step 3 is executed in $O(1)$ time, the steps 4 to 7 are executed in $O(n)$ time and step 8 is executed in $O(n)$ time in the Algorithm 2.1.

Example 2.4 Consider the graph G of order $n(= 5)$ given in Figure 1.1 and the distance matrix $D(G)$ of G as in the Example 1.1. Now using the Algorithm 2.1, let us find the (V, ζ)-distance between the vertex $i = 1$ and the clique $C = \{1, 2\}$. Clearly $BC(C) = (11000)$. Since $BC(C(i)) = 1$, the Algorithm 2.1 returns (V, ζ)-distance $d(i, C) = 0$. Again using the Algorithm 2.1, let us find the (V, ζ)-distance $d(i, C)$ between the vertex $i = 1$ and the clique $C = \{2, 4\}$. Clearly $BC(C) = (01010)$. Since $BC(C(i)) = 0$, the Algorithm 2.1 finds the (V, ζ)-distance $d(i, C) = min\{d(i, j) : 1 \le j \le n\}$. Since $BC(C(j)) = 0$, $d(i, j) = n$ for $j = 1, 3, 5$ and since $BC(C(j)) = 1$, for $j = 2, 4$, $d(i, 2) = d_{i2} = 1$ and $d(i, 4) = d_{i4} = 2$. Hence the algorithm 2.1 returns (V, ζ)-distance $d(i, C) = min\{d(i, j) : 1 \le j \le n\} = min\{d(1, 1), d(1, 2), d(1, 3), d(1, 4), d(1, 5)\} = min\{5, 1, 5, 2, 5\} = 1$.

Next, we introduce an algorithm to find the (V, ζ)-eccentricity $e_1(i)$ of a vertex i in a graph G of order n using BC representation.

Algorithm 2.5 Let G be a graph with $V = \{1, 2, 3, . . . , n\}$ and $\zeta = \{C : C$ is a clique in BC representation in $G\}$.

1. Let $\zeta = \{C_1, C_2, . . . , C_m\}$.
2. Let $i \in V$
3. for $j = 1$ to m
4. Find $d(i, C_j)$, by calling Algorithm 2.1
5. next j
6. find $e_1(i) = max\{d(i, C_j) : 1 \le j \le m\}$
7. return $e_1(i)$
8. stop

Theorem 2.6 For a vertex i and the set of all cliques ζ in G, the Algorithm 2.5 finds (V, ζ)-eccentricity $e_1(i)$.

Proof. Let G be a graph with $V = \{1, 2, 3, . . . , n\}$ and $\zeta = \{C_1, C_2, . . . , C_m\}$ be the set of all cliques in BC representation in G. Let $i \in V$. Then the step 4 of Algorithm 2.5 finds the (V, ζ)-distance $d(i, C_j)$ between the vertex i and every clique $C_j (1 \le j \le m)$ in G, and the step 6 of Algorithm 2.5 finds (V, ζ)-eccentricity $e_1(i) = max\{d(i, C_j) : 1 \le j \le m\}$. Hence the theorem.

Theorem 2.7 The Algorithm 2.5 finds (V, ζ)-eccentricity $e_1(i)$ of vertex i in a graph G in $O(mn)$ time.

Proof. By Theorem 2.3, the time complexity of the step 4 in the Algorithm 2.5 is $O(n)$, so that the steps 3 to 5 in the Algorithm 2.5 are executed in $O(mn)$ time. The time complexity of the step 6 in the Algorithm 2.5 is $O(m)$. Hence the theorem.

Example 2.8 Consider the graph G given in Figure 1.1 with the vertex set V and the clique set ζ as in the Example 1.1. Clearly the order n of G is 5 and the number of cliques m in G is 5. Let $C_1 = (11000)$, $C_2 = (10100)$, $C_3 = (01010)$, $C_4 = (00110)$, $C_5 = (00011)$, and $i = 1 \in V$. Now we find the (V, ζ)-eccentricity $e_1(i)$. By calling the Algorithm 2.1 m times, the step 4 of Algorithm 2.5 finds the (V, ζ)-distances $d(i, C_1) = 0$, $d(i, C_2) = 0$, $d(i, C_3) = 1$, $d(i, C_4) = 1$ and $d(i, C_5) = 2$. Then the step 6 of Algorithm 2.5 finds the (V, ζ)-eccentricity $e_1(i) = max\{0, 0, 1, 1, 2\} = 2$.

Finally, we introduce an algorithm to find the (V, ζ)-center $Z_1(G)$ of a graph G of order n using BC representation.

Algorithm 2.9 Let G be a graph with $V = \{1, 2, 3, . . . , n\}$ and $\zeta = \{C : C$ is a clique in BC representation in $G\}$.

1. Let $\zeta = \{C_1, C_2, . . . , C_m\}$

Volume: 04 Issue: 06 Pages:1809-1811 (2013) ISSN : 0975-0290

2. Let $Z_1(G) = \Phi$.

3. for $i = 1$ to n

4. Find $e_1(i)$, by calling Algorithm 2.5.

5. next i

6. Find $r_1 = min\{e_1(i) : 1 \leq i \leq n\}$

7. for $i = 1$ to n

8. if $e_1(i) = r_1$ then $Z_1(G) = Z_1(G)$ U $\{i\}$.

9. next i

10. Stop

Theorem 2.10 For a graph G, the Algorithm 2.9 finds (V, ζ)-center $Z_1(G)$ of G.

Proof. Let G be a graph with $V = \{1, 2, . . . , n\}$ and $\zeta = \{C_1, C_2, . . . , C_m\}$ be the set of all cliques in their BC representation in G. The step 4 of Algorithm 2.9, finds (V, ζ)-eccentricity $e_1(i)$ for all $i \in V$ ($1 \leq i \leq n$). Then the step 6 finds (V, ζ)-radius $r_1 = min\{e_1(i) : i \in V\}$ of G, and the steps 7 to 9 find (V, ζ)-center $Z_1(G) = \{i \in V : e_1(i) = r_1\}$. Thus the Algorithm 2.9 finds (V, ζ)-center $Z_1(G)$ of G.

Theorem 2.11 The (V, ζ)-center $Z_1(G)$ of a graph G can be obtained in $O(mn^2)$ time using Algorithm 2.9.

Proof. By Theorem 2.7, the computing time for step 4 of the Algorithm 2.9 is $O(mn)$ so that time complexity for the steps 3 to 5 of the Algorithm 2.9 is $O(mn^2)$. The step 6 of the Algorithm 2.9 finds r_1 in $O(n)$ time and the steps 7 to 9 of the Algorithm 2.9 finds $Z_1(G)$ of G in $O(n)$ time. Hence the theorem.

Example 2.12 Consider the graph G given in Figure 1.1 as in the Example 1.1. Clearly the vertex set of G is $V = \{1, 2, 3, 4, 5\}$ and the set of all cliques in G is $\zeta = \{(11000), (10100), (01010), (00110), (00011)\}$. Now we find the ($V$, ζ)-center $Z_1(G)$. By calling the Algorithm 2.5 n times, the step 4 of Algorithm 2.9 finds the (V, ζ)-eccentricities $e_1(1) = 2$, $e_1(2) = 1$, $e_1(3) = 1$, $e_1(4) = 1$, and $e_1(5) = 2$. The step 6 of Algorithm 2.9 finds the (V, ζ)-radius $r_1 = min\{e_1(i) : 1 \leq i \leq n\} = 1$. Finally, the step 8 of Algorithm 2.9 finds the (V, ζ)-center $Z_1(G) = \{i \in V : e_1(i) = r_1\} = \{2, 3, 4\}$.

3 Conclusion

In this paper we have developed sequential algorithms to find the (V, ζ)- central structures in a graph G and these algorithms may be used in networking, data mining and cluster analysis.

References

[1]A.Ashok Kumar, S.Athisayanathan, A.Antonysamy, Algorithm to Find All Cliques in a Graph, *International Journal of Advanced Networking and Applications, Vol. 2*, No-2(2010), 597-601.

[2]G. Chartrand and Ortrud R. Oellermann, *Applied and Algorithmic Graph Theory*, McGraw-Hill International editions (1993).

[3]K.R.Parthasarathy, *Basic Graph Theory*, Tata McGraw-Hill Publishing Company Limited, New Delhi(1994).

[4]A.P.Santhakumaran and S.Arumugam, centrality with Respect to Cliques, *International Journal of Management and Systems, 18*, No.3(2002), 275-280

Authors Biobraphy

Dr.A.Ashok Kumar **–** He was awarded Ph.D. in computer Science by Manonmaniam sundaranar university, Tirunelveli, Tamil Nadu, India. He authored three books in computer science. Now he is working as Assistant Professor in computer science at Alagappa Govt Arts College, Karaikudi, Tamilnadu. His area of research interests includes Algorithms, Networks and Distributed systems.

Dr.S.Athisayanathan – He was awarded Ph.D. in Mathematics by Manonmaniam sundaranar university, Tirunelveli, Tamil Nadu, India. He is working as Professor and Head in Research department of mathematics at St.Xavier's College(Autonomous), Palayamkottai, TamilNadu. His area of interest includes Detour concepts in graphs and related Graph theory areas.

.

Rev.Dr.A.Antonysamy **–** He was awarded Ph.D. in computer science by Madurai Kamaraj University, TamilNadu. He has more than 35 years of experience in teaching and research. He is working as Principal, Ananda College, Devakottai, Tamilnadu. His area of interest include Graph theory and Computer Algorithms.