FIND THE DISTANCE OF SHORTEST PATH ALLOCATION BY USING GRAPH THEROY NETWORKS

M.JOICE MABEL Msc. MPhil. B.Ed.,SET.,

[1].Assistant Professor Annai Vailankanni Arts & Science College-Thanjavur

Abstract:

The aim of the paper is to the importance of graph distance theoretical concepts and the applications of distance - 2 domination in graphs to various real life situations in the areas of science and engineering. A set D is a distance - 2 dominating set if for every vertex $u\varepsilon V-D$, d(u, D) ≤2and is denoted by $\gamma_{\leq 2}(G)$. In this paper, we get many limits and accurate values for some standard graphs. Also this paper explores mainly on the applications of distance - 2 dominating sets in networks. Nordhaus – Gaddum type results are obtained for this parameter. Distance - 2 dominating sets are identified in Routing, Radio Stations, Communication Networks and Mobile adhoc Networks and shortest path selection process.

KEYWORD: *graph, tree, network, distance vectors*

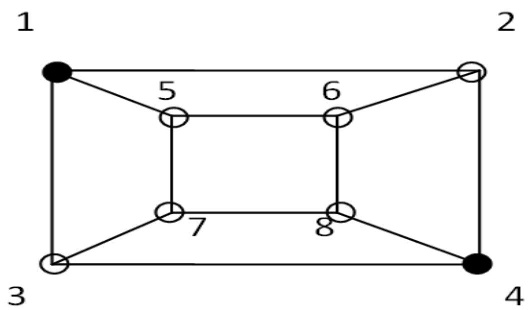
Introduction:

Graph Theory" is an important branch of Mathematics. It has grown rapidly in recent times with a lot of research activities. In the last few decades, at the international level, one third of the Mathematics research papers are from Graph Theory and combinatory. Applications of graph theory in computer and communication, social networks, Molecular physics and chemistry, Biological sciences, Engineering and in other numerous areas In graph theory, one of the extensively researched branches is domination in graphs. This is largely due to a variety of new parameters that can be developed from the basic definition of domination. The NP- completeness in basic domination problems and its close relationship to other NP- completeness problems have contributed to the enormous growth of research activity in Domination theory. All graphs considered here are simple, finite and undirected. The greatest (least) integer less (greater) than or equal to x is $\lfloor x\rfloor(\lceil x\rceil)$. In this paper, the terms and notations used may be found in A set D⊆ Vof vertices in a graph G = (V,E) is called a dominating set if every vertex $v\varepsilon V$ is either an element of D or it is adjacent to an element of D. The domination number of G is the minimum cardinality of a dominating set and it is denoted by γ (G). A recent survey of γ (G) can be found in defines a connected dominating set D to be a dominating set D whose induced subgraph $\langle D\rangle$ is connected. The connected domination number $\gamma_c(G)$ of a connected graph G is the minimum cardinality of a connected dominating set of G.

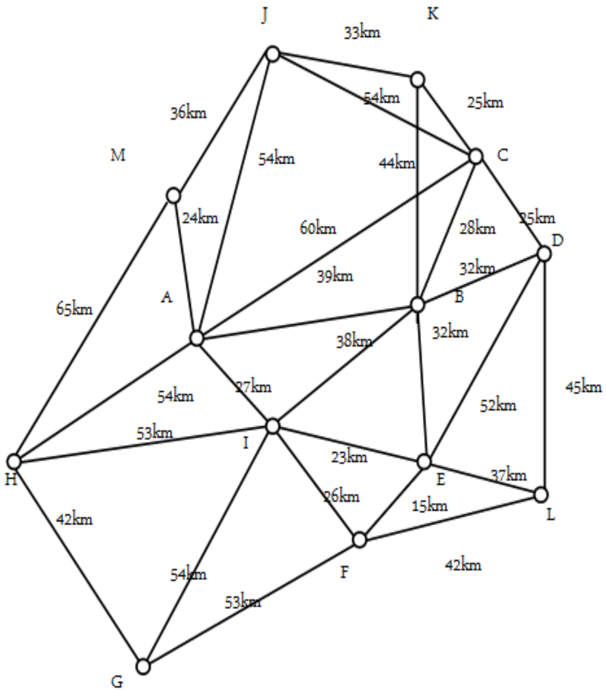
Graph.

Weighted graphs occur frequently in applications of graph theory. In the friendship·,·· graph, for. ex'amp'le, weights might indicate intensity of friendship; in the com,munications graph, they could represent the· construction or maintenance costs of the various communication links.If H is a subgraph of a weighted. gr~ph, the *weight* w(H) of H is the sum of the weights eE~H) *w(e)* on its edges. Many optimisation problems amount to finding, in. a weighted graph, a su.bgraph of a certain type with minimum (or maximum) weight. One such is t~: *sh9rtest path problem:* given a. railway net-work connecting various towns, determine a shortest route 'between two specified towns in the network. . _ . Here. One must find, in a weighted graph, a path of minimum weight c'onnecting two specified vertices u~ an·d *Vo;* the weights'represent distances b'y ·rail between d·irectly-linked towns, and are therefore nOil-negative. The path indicated in the. graph of figure' 1.11 is il (uo, vo)-path of minimum weight (exercise l .8.1). " We now present an algorithm for solving the shortest path problem. For· clarity of exposition, we shall refer to the we.ight of a path' in a. weighted graph as its *leng·th;* similarly th-e minimum weight.' of a *(u,* v)-path will be called the *distance* between *u* and *v'* and, den.oted by *d(u,. v).* These definitions coincide with the usual notions 'of length and distance, as defined in section 1.6, when all the weights are equal toone. It clearly s'l:lffices to deal with the shortest path problem for simple graphs; so we shall assume here that Gis simple. We shall also assume. that all the weights are positive. This, -again, is not a serious restriction because, if the w~ight .of an ,edge is .zero, then· its ends ca·n· be id·entified. W·e adopt the convention that *w(uv).* , cx) if uv~ E.

Graphs and Sub graphs:

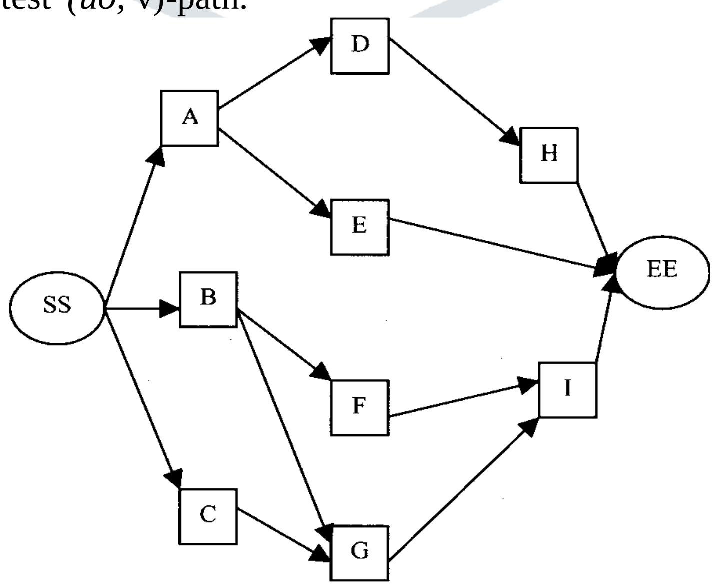


The algorithm to be described was discQveredby Dijkstra (1959) and, independently, by Whiting and Hillier (1960). It finds not only a shortest *(uo,* vo)-path, but shortest paths from *Uo* to all other vertices of G·. The basic idea is as .f,ollows. - Suppose that S is a proper subset of *'V* such that *Uo* E S, and let S denote *V\S.* If P =*Uo* · · · *iii)* is a shortest· path from *Uo* to 5 then clearly ii E Sand the *(uo,* u)-section of P must be a shortest *(uo,* u)-path. Therefore *d(uo,* i3) = *d(uo, u)* +*w(uv)* and the distance from *Uo* to 5 is given by the formula *d(uo,* S) = min{d(uo, *u)* + *w(uv)}* ueS veS This formula is the basis of Dijkstra's algorithm. Starting with the set So = *{uo},* an increasing sequence So, 51, ... ,5 11 - 1 of subsets of *V* is constructed, in such a way that, at the end of stage i, shortest paths from *Uo* to all vertices in Si are known. The first step is to determine a vertex nearest to uo. This is achieved by computing *d(uo,* So) and selecting a vertex *Ul* ESo such that *d(uo, Ul)*= *d(uo,* So); by (1.1) *d(uo,* So) = min{d(uo, *u)* + *w(uv)}* = *min{w(uov)}* ueSo veSo ~ESo and so *d(uo,* 50) is easily computed. We now set SI = *ruG, Ul}* and let PI



Route allocation

denote the path $u_o.u_t$; this is clearly a shortest (uo,.u1)-path. In general, if the set Sk = {uo, Ul, · • ., Uk} and corresponding shortest paths PI, P_2, ..., Pk have ,already been detennined, we compute *d(uo,* Sk) using (1.1) and select a vertex Uk+l E 5k such that *d(uo,* Uk+l}= *d(uo,* Sk)' By (1.1), *d(uo,* Uk+l) = d(uo,Uj)+W(UjUk+l) for some *j<k;* we get a shortest (uo,uk+l)-path by adjoining the edge UjUk+1 to the path Pj. We illustrate this procedure by considering the weighted graph depicted in figure 1.12a. Shortest paths from *Uo* to the remaining vertices are determined in seven stages. At each stage, the vertices to which shortest paths have been found are indicated by solid dots, and each is labelled by its distance from *uo;* initially Uo is labelled O. THe actual shortest paths are indicated by solid lines. Notice that, at each stag'e, these sh.9rtest paths together form a COftnected graph without cycles; SllCh a graph is called a *tree*, and we can think of the :algorthm as a 'tree-growing' procedure. The final tree, in figure *1.12h*, has the property that, for each vertex *v*, the path conIlecting Uo and *v* is a shortest' *(uo,* v)-path.



Converted activity-on-node critical-path method network for example problem

Implementation process:

In this section, we discuss the implementation details of measuring similarity between graphs based on the DFS code mentioned in Section 2. Note that, we view the graph which is used to match against the graph database as the source graph. According to the real requirements of different application scenarios, we can divide our implementation into two main phases which are preprocessing and matching. Preprocessing phase is performed offline while matching is online. Usually, people pay much attention to matching phase since it has direct influences on user experiences driven by efficient performance. Firstly, we present the pseudo code of preprocessing.

Radio stations

Suppose that we have a collection of small villages in a remote part of the world. We would like to locate radio stations in some of these villages so that messages can be broadcasted to all the villages in the region. But since the installations of radio stations are costly, we want to locate as few as possible which can cover all other villages. Let each village be represented by a vertex. An edge between two villages is labeled with the distance, say in kilometers. The distance between the two villages is shown in fig.3. Let us assume that a radio station has a broadcast range of hundred kilometers. In this case we seek a distance - 2 dominating set among all the vertices within the distance of 100 kilometers. Clearly this fig.4 gives the distance - 2 dominating set.

CONCLUSION

In this paper, we have discussed the use of distance – 2 dominating sets in graph communication networks. In future propose to extend this work with an algorithm to identify a distance - 2 dominating set in any communication network with different transmission radius. We need to verify the effectiveness of our updated strategies when the topology of the underlying network changes and that graph process first search to allocation of find out the best path and choose to graph theory action that event process to send information process on graph allocation action and implementation of future work of few distance allocation and The connected domination number $\gamma_c(G)$ of a connected graph G is the minimum cardinality of a connected

REFERENCES

[1]. E.J.Cockayne, S.T.Hedetniemi, "Towards a Theory of Domination in Graphs",Networks,7:247-261.

[2]. J.Cyman, M.Lemanska and J.Raczek: "Lower bound on the distance *k*-domination number of a tree", *Math. Slovaca* 56 (2006), no. 2, 235-243.

[3]. O.Favaron and D.Kratsch: Ratios of domination parameters, "Advances in graph theory, viswa", Gulbarga (1991), 173-182.

[4]. P. Fraisse, "A note on distance dominating cycles". Discrete Math. 71 (1988), 89-92.

[5]. A. Hansberg, D. Meierling, and L. Volkmann, "Distance domination and distance irredundance in graphs". Electronic J.Combin. 14 (2007), #R35.