Annals of Pure and Applied Mathematics
Vol. 7, No.1, 2014, 13-18
ISSN: 2279-087X (P), 2279-0888(online)
Published on 9 September 2014
www.researchmathsci.org

Annals of
Pure and Applied Mathematics

Algorithms to Find Wiener Index of Some Graphs

Kalyani Das

Department of Mathematics
Ramnagar College, Purba Medinipur, West Bengal, India

Received 5 July 2014; accepted 20 August 2014

Abstract. The Wiener index is one of the oldest graph parameter which is used to study molecular-graph-based structure. This parameter was first proposed by Harold Wiener in 1947 to determining the boiling point of paraffin. The Wiener index of a molecular graph measures the compactness of the underlying molecule. This parameter is wide studied area for molecular chemistry. It is used to study the physio-chemical properties of the underlying organic compounds. The Wiener index of a connected graph is denoted by $W(G)$ and is defined as $W(G) = \frac{1}{2}\sum_{u,v} d(u,v)$, that is $W(G)$ is the sum of distances between all pairs (ordered) of vertices of G. In this paper, we give the algorithmic idea to find the Wiener index of some graphs, like cactus graphs and intersection graphs, viz. interval, circular-arc, permutation, trapezoid graphs.

Keywords: Wiener index, cactus graphs, intersection graphs, interval graphs, circular-arc graphs, permutation graphs, trapezoid graphs.

AMS Mathematics Subject Classification (2010): 05C12, 05C85

1. Introduction

Molecular descriptor is a final result of a logic and mathematical procedure which transforms chemical information encoded with in a symbolic representation of a molecule into a useful number or the result of some standardized experiment. The Wiener index ***W*(*G*)** is a distance-based topological invariant is also a molecular descriptor, it much used in the study of the structure-property and the structure-activity relationships of various classes of biochemically interesting compounds introduced by Harold Wiener in 1947 for predicting boiling points (b,p) of alkanes based on the formula $b.p = \alpha W + \beta w(3) + \gamma$, where α, β, γ are empirical constants, and $w(3)$ is called path number. It is defined as the half sum of the distances between all pairs of vertices of G[1,12,14]

$$W(G) = \frac{1}{2}\sum_{u,v} d(u,v)$$

2. Algorithms to find Wiener index

2.1. Cactus graphs

The class of cactus graph is an important subclass of general planar graphs. Let $G=(V,E)$ be a finite, connected, undirected simple graph of n vertices m edges, where V is the set of vertices and E is the set of edges.

A vertex u is called a *cutvertex* if removal of u and all edges incident on u disconnect the graph. A connected graph without a cutvertex is called a *non-separable* graph. A *block* of a graph is a maximal non-separable subgraph. A *cycle* is a connected graph (or subgraph) in which every vertex is of degree two. A block which is a cycle is called a *cyclic block*. A *cactus graph* is a connected graph in which every block is either an edge or a cycle. A *weighted* graph G is a graph in which every edge is associates with a weight. Without loss of generality we assume that all weights are positive. A *weighted cactus graph* is a weighted, connected graph in which every block containing two vertices is an edge and three or more vertices is a cycle. A *path* of a graph G is an alternating sequence of distinct vertices and edges which begins and ends with vertices in G. The *length* of a path is the sum of the weights of the edges in the path. a path from vertex u to v is a *shortest path* if there is no other path from u to v with lower length. The *distance* $d(u,v)$ between vertices u and v is the length of shortest path between u and v in G.

Theorem 1. [13] *The shortest distances from a specified vertex to all other vertices of a weighted cactus graph can be computed in $O(n)$ time and the all pair shortest distance of a weighted cactus graph can be computed in $O(n^2)$ time, where n represents the total number of vertices of the graph.*

Theorem 2. *The Winner index of the cactus graphs can be computed in $O(n^2)$ time, where n represents the total number of vertices of the graph.*

Definition of Intersection graphs

A graph $G=(V,E)$ is called an *intersection graph* for a finite family F of a non-empty set if there is a one-to-one correspondence between F and V such that two sets in F have non-empty intersection if and only if their corresponding vertices in V are adjacent. We call F an intersection model of G. For an intersection model F, we use $G(F)$ to denote the intersection graph for F.

Depending on the nature or geometric configuration of the sets $S_1, S_2, \ldots$ different types of intersection graphs are generated. The most useful intersection graphs are

- Interval graphs (S is the set of intervals on a real line)

- Tolerance graphs
- Circular-arc graphs (S is the set of arcs on a circle)
- Permutation graphs (S is the set of line segments between two line segments)
- Trapezoid graphs (S is the set of trapeziums between two line segments)
- Disk graphs (S is the set of circles on a plane)
- Circle graphs (S is the set of chords within a circle)
- Chordal graphs (S is the set of connected subgraphs of a tree)
- String graphs (S is the set of curves in a plane)
- Graphs with boxicity k (S is the set of boxes of dimension k)
- Line graphs (S is the set of edges of a graph).

2.2. Interval graphs

An undirected graph $G = (V, E)$ is said to be an *interval graph* if the vertex set V can be put into one-to-one correspondence with a set I of intervals on the real line such that two vertices are adjacent in G if and only if their corresponding intervals have non-empty intersection. That is, there is a bijective mapping $f : V \rightarrow I$.

The set I is called an interval representation of G and G is referred to as the interval graph of I [19]. A large number of work on intersection graphs and cactus graphs have been done in [20-37].

Theorem 3. *[7] The time complexity for finding the distances between all pair of vertices on interval graphs is* $O(n^2)$.

Theorem 4. *The Winner index of the interval graphs can be computed in* $O(n^2)$ *time, where n represents the total number of vertices of the graph.*

2.3. Circular-arc graphs

A graph is a *circular-arc* graph if there exists a family A of arcs around a circle and a one-to-one correspondence between vertices of G and arcs in A, such that two distinct vertices are adjacent in G if and only if the corresponding arcs intersect in A. Such a family of arcs is called an *arc representation* for G.

A graph G is a *proper circular-arc (PCA)* graph if there exists an arc representation for G such that no arc is properly included in another.

A graph G is a *unit circular-arc (UCA)* graph if there exists an arc representation for G such that all the arcs are of the same length.

Theorem 5. *[18] The all-pair shortest paths problem on circular-arc graph is computed in* $O(n^2)$ *time.*

Theorem 6. *The Winner index of the circular-arc graph is computed in* $O(n^2)$ *time.*

2.4. Permutation graphs

An undirected graph $G=(V,E)$ with vertices $V=\{1,2,\ldots,n\}$ is called a permutation graph if there exists a permutation π on $N=\{1,2,\ldots,n\}$ such that for all $i,j\in N$,

$$(i-j)(\pi^{-1}(i)-\pi^{-1}(j))<0$$

if and only if i and j are joined by an edge in G [19]. Geometrically, the integers $1,2,\ldots,n$ are drawn in order on a real line called as *upper line* and $\pi(1),\pi(2),\ldots,\pi(n)$ on a line parallel to this line called as *lower line* such that for each $i\in N$, i is directly below $\pi(i)$. Next, for each $i\in V$, a line segment is drawn from i on the lower line to i on the upper line and it is denoted by $l(i)$. Then from definition it follows that there is an edge (i,j) in G if and only if the line segment $l(i)$ for i intersects the line segment $l(j)$ for j.

Theorem 7. *[17] The all-pair shortest paths problem on permutation graphs in* $O(n^2)$ *time.*

Theorem 8. *The Winner index of the permutation graphs can be computed in* $O(n^2)$ *time.*

2.5. Trapezoid graphs

A trapezoid T_i is defined by four corner points $[a_i,b_i,c_i,d_i]$, where $a_i<b_i$ and $c_i<d_i$ with a_i,b_i lying on top line and c_i,d_i lying on bottom line of a rectangular channel. An undirected graph $G=(V,E)$ with vertex set $V=\{v_1,v_2,\ldots,v_n\}$ and edge set $E=\{e_1,e_2,\ldots,e_m\}$ is called a trapezoid graph if a trapezoid representation can be obtained such that each vertex v_i in V corresponds to a trapezoid T_i and $(v_i,v_j)\in E$ if and only if the trapezoids T_i and T_j corresponding to the vertices v_i and v_j intersect. For simplicity the vertices $v_1,v_2,\ldots,v_n$ are represented respectively by 1, 2, $\ldots$, n. Thus the edge $(i,j)\in E$ if and only if T_i and T_j intersect in the trapezoid representation.

Theorem 9. *The time complexity to find all pairs shortest distances on trapezoid graphs is* $O(n^2)$.

Theorem 10. *The time complexity to compute Winner index of a trapezoid grapg is* $O(n^2)$.

REFERENCES

1. A.Graovac and T.Pisanski, On the Wiener index of a graph, *Journal of Mathematical Chemistry*, 8 (1991) 53-62.
2. R.Balakrishnan and K.Renganathan, *A Text Book of Graph Theory,* Springer-Verlag, New York, 2000.
3. R.Balakrishnan, The energy of a graph, *Linear Algebra and its Applications,* 387 (2004) 287–295.
4. G.V. Ghodasara and J.P. Jena, Prime cordial labeling of the graphs related to cycle with one chord, twin chords and triangle, *Intern. J. Pure and Applied Mathematics,* 89(1) (2013) 79-87.
5. I.Gutman and O.E.Polansky, *Mathematical Concepts in Organic Chemistry,* Springer-Verlag, Berlin, 1986.
6. F. Harary, *Graph Theory*, Addison –Wesley, Reading, MA, 1971.
7. M. Pal and G. P. Bhattacharjee, An optimal parallel algorithm for all-pairs shortest paths on interval graphs, *Nordic J. Computing*, 4 (1997) 342-356.
8. I.Gutman and D.Cvetkovi´c, *Selected Topics on Applications of Graph Spectra*, Beograd: Matematick institut SANU, 2011.
9. S. Radenković and I. Gutman, Relation between Wiener index and spectral radius, *Kragujevac J. Sci.*,30 (2008) 57-64.
10. K.Thilakam and A.Sumathi, How to Compute the Wiener index of a graph using MATLAB, *Intern. J. Applied Mathematics and Statistical Sciences*, 2(5) (2013) 143-148.
11. S.K.Vaidya and C.M.Barasara, Product cordial labeling for some new graphs, *Journal of Mathematics Research*, 3(2) (2011) 206-211.
12. H. Wiener, Structural determination of paraffin boiling points, *J. Am chem. Soc*., 6 (1947) 17-20.
13. K. Maity, M. Pal and T.K.Pal, An optimal algorithm to find all-pairs shortest paths problem on weighted cactus graphs, *V.U.J. Physical Sciences*, 6 (2000) 45-57.
14. http://mathworld.wolfram.com/WienerIndex
15. K.Thilakam and A.Sumathi, Wiener index of a cycle in the context of some graph operations, *Annals of Pure and Applied Mathematics*, 5(2) (2014) 183-191.
16. S. Mondal, M. Pal and T.K.Pal, An optimal algorithm for solving all-pairs shortest paths on trapezoid graphs, *Int. J. Comput. Eng. Sc.*, 3(2) (2002) 103-116.
17. S. Mondal, M. Pal and T.K.Pal, An optimal algorithm to solve the all-pairs shortest paths problem on permutation graph, *J. Mathematical Modelling and Applications*, 2(1) (2003) 57-65.
18. A. Saha, M. Pal and T. K. Pal, An optimal parallel algorithm to find all-pairs shortest paths on circular-arc graphs, *J. Applied Mathematics and Computing,* 17 (1-2) (2005) 1-23.
19. A.Pal and M.Pal, Interval tree and its applications, *Advanced Modeling and Optimization*, 11(3) (2009) 211-226.
20. M.Pal and G. P.Bhattacharjee, A sequential algorithm for finding a maximum weight k-independent set on interval graphs, *Intern. J. Computer Mathematics*, 60 (1996) 205-214.

21. A.Saha, M.Pal and T.K.Pal, Selection of programme slots of television channels for giving advertisement: A graph theoretic approach, *Information Sciences*, 177 (12) (2007) 2480-2492.
22. D.Bera, M.Pal and T.K.Pal, An efficient algorithm for generate all maximal cliques on trapezoid graphs, *Intern. J. Computer Mathematics*, 79 (10) (2002) 1057--1065.
23. M.Hota, M.Pal and T.K.Pal, An efficient algorithm to generate all maximal independent sets on trapezoid graphs, *Intern. J. Computer Mathematics*, 70 (1999) 587--599.
24. M.Pal and G.P.Bhattacharjee, A data structure on interval graphs and its applications, *Journal of Circuits, System and Computers*, 7(3) (1997) 165-175.
25. A.Saha and M.Pal, Maximum weight k-independent set problem on permutation graphs, *International J. of Computer Mathematics*, 80(12) (2003) 1477--1487.
26. S.C.Barman, S.Mondal and M.Pal, An efficient algorithm to find next-to-shortest path on trapezoidal graph, *Advances in Applied Mathematical Analysis*, 2(2) (2007) 97-107.
27. S.Mandal and M.Pal, A sequential algorithm to solve next-to-shortest path problem on circular-arc graphs, *Journal of Physical Sciences*, 10 (2006) 201-217.
28. S.Mondal, M.Pal and T.K.Pal, An optimal algorithm to solve 2-neighbourhood covering problem on interval graphs, *Intern. J. Computer Mathematics*, 79(2) 189--204 (2002).
29. D.Bera, M.Pal and T.K.Pal, An efficient algorithm for finding all hinge vertices on trapezoid graphs, *Theory of Computing Systems*, 36(1) (2003) 17-27.
30. M.Pal, Efficient algorithms to compute all articulation points of a permutation graph, *The Korean J. Computational and Applied Mathematics*, 5(1) (1998) 141-152.
31. M.Pal and G.P.Bhattacharjee, An optimal parallel algorithm to color an interval graph, *Parallel Processing Letters*, 6 (4) (1996) 439-449.
32. M.Pal and G.P.Bhattacharjee, An optimal parallel algorithm for computing all maximal cliques of an interval graph and its applications, *J. of Institution of Engineers (India)*, 76 (1995) 29-33.
33. A.Saha, M.Pal and T.K.Pal, An efficient PRAM algorithm for maximum weight independent set on permutation graphs, *Journal of Applied Mathematics and Computing*, 19 (1-2) (2005) 77-92.
34. D.Bera, M.Pal and T.K.Pal, An optimal PRAM algorithm for a spanning tree on trapezoid graphs, *J. Applied Mathematics and Computing*, 12(1-2) (2003) 21-29.
35. N.Khan, M.Pal, A.Pal, *L*(0,1)-Labelling of Cactus Graphs, *Communications and Network*, 4 (2012), 18-29.
36. P.K.Ghosh and M. Pal, An optimal algorithm to solve 2-neighbourhood covering problem on trapezoid graphs, *Advanced Modeling and Optimization*, 9(1) (2007) 15-36.
37. M.Hota, M.Pal and T.K. Pal, Optimal sequential and parallel algorithms to compute all cut vertices on trapezoid graphs, *Computational Optimization and Applications*, 27 (2004) 95-113.