IOSR Journal of Mathematics (IOSR-JM)
e-ISSN: 2278-5728,p-ISSN: 2319-765X, Volume 8, Issue 4 (Sep. - Oct. 2013), PP 33-41
www.iosrjournals.org

The conditional neighborhood for graph and its algorithm.

Mabrok EL- Ghoul[1] , HabibaEl-Zohny[2] , Hend El- Morsy[3]
[1]*Mathematics Department Faculty of science Tanta University*
[2,3]*Mathematics Department Faculty of science Alazhar University*

Abstract: *In this paper we will define the conditional neighborhood for graph and classified the conditions into many types. In each type we will compute the algorithm for graph . We will prove that the neighborhood will be give different neighborhood by different algorithm.*
Keywords: *Neighborhood , Graph , Algorithm.*
AMS Subject Classification:*05C85, 68R10*

I. Definitions:

DefinitionofGraph: An (undirected) graph G is defined by two finite sets.a non-void set Xof elements called vertices, a set E (which can be empty) of elements callededges, with for each edge e two associated vertices,x and y, distinct or not,called the endvertices
of e [3].
Definition of weighted graph: Is a graph for which each edge has anassociated real number weight[4].
Definition of degree: The degree of a vertex x in a graph G is the number of edges in G incident to x, that is edges with x as an endvertex , loops being counted twice. This integer is denoted by d(x) or d_G (x) [3].
Definitionof Algorithm: In mathematics and computer science, an algorithm is an effective method expressed as a finite list of well-defined instructions for calculating a function.In simple words an algorithm is a step-by-step procedure for calculations[5].
DefinitionofShortest path algorithm: An algorithm that is designed essentially to find a path of minimum length between two specified vertices of connected weighted graph [3].
Definition of curvature:In general, there are two important types of curvature: extrinsic curvature and intrinsic curvature. The extrinsic curvature of curves in two- and three-space was the first type of curvature to be studied historically, culminating in the Frenet formulas, which describe a space curve entirely in terms of its "curvature," torsion, and the initial starting point and direction [1].
Definition ofTorsion: The torsion of a space curve, sometimes also called the "second curvature" is the rate of change of the curve's osculating plane. The torsion τ is positive for a right-handed curve, and negative for a left-handed curve. A curve with curvature $\kappa \neq 0$ is planar iff $\tau = 0$. The torsion can be defined by $\tau \equiv -\mathbf{N} \cdot \mathbf{B}'$, where $\mathbf{N}$ is the unit normal vector and **B'**is the unit binormal vector[2].
Kruskal's algorithm:
Input :*G* (a weighted graph with*n* vertices.)
Algorithm body:
(Build a subgraph*T* of *G* to consist of all the vertices of G with edges added in order of increasing weight. At each stage, let *m* is the number of edges of *T*). Initialized *T* to have all vertices of *G* and no edges.
1. Let *E* be the set of all edges of *G* , and let*m*: = 1.
[pre-condition: *G* is connected.]
3. **While** ($m \le n - 1$)
3a. Find an edge *e* in *E* of least weight.
3b. Delete *e* from *E*.
3c. If addition of *e* to edge set of *T* doesn't produce a circuit
Then add *e* to the edge set of *T* and set *m* : = *m* +1
End while
[post-condition : *T* is minimum spanning tree for *G* .]
Output: ***T***[4].

II. Main Results:

Definition:
Conditional neighborhood: Is a neighborhood in which we put a condition to find all its vertices.
Types of conditions to find neighborhood:
The conditions can be classified into three types:

1. Condition describes the algorithm.
2. Condition describes the geometric classification of graph.
3. Condition describes the algorithm and the geometric classification together.

Type (1): Condition describes the algorithm:
In this type we can change the condition on algorithm such as:
1. Shortest path algorithm's condition.
2. Longest path algorithm's condition.

And so on,
We will illustrate some examples in each case.
Example 1 :
Consider a graph shown in Fig.(1), if the condition is : The neighborhood for any vertex is all vertices belongs to shortest path from v_0 to v_6 we have:

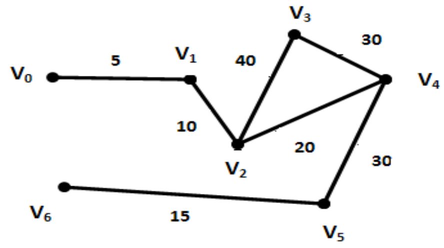


Fig.(1)

V	N(V)
V_0	$\{V_1, V_2, V_4, V_5, V_6\}$
V_1	$\{V_2, V_4, V_5, V_6\}$
V_2	$\{V_4, V_5, V_6\}$
V_3	$\{V_4, V_5, V_6\}$
V_4	$\{V_5, V_6\}$
V_5	$\{V_6\}$
V_6	$\{V_6\}$

But if the condition is all vertices belongs to longest path from V_0 to V_6 we have:

V	N(V)
V_0	$\{V_1, V_2, V_3, V_4, V_5, V_6\}$
V_1	$\{V_2, V_3, V_4, V_5, V_6\}$
V_2	$\{V_3, V_4, V_5, V_6\}$
V_3	$\{V_4, V_5, V_6\}$
V_4	$\{V_5, V_6\}$
V_5	$\{V_6\}$
V_6	$\{V_6\}$

For the previous graph , if we find minimum spanning tree by using Kruskal's algorithm we have:

Iteration no.	Edge considered	weight	Action taken
1	V_0 __ V_1	5	added
2	V_1 __ V_2	10	added
3	V_2 __ V_3	40	added
4	V_3 __ V_4	30	added
5	V_4 __ V_5	30	added
6	V_2 __ V_4	20	not added
7	V_5 __ V_6	15	added

And minimum spanning tree will be:

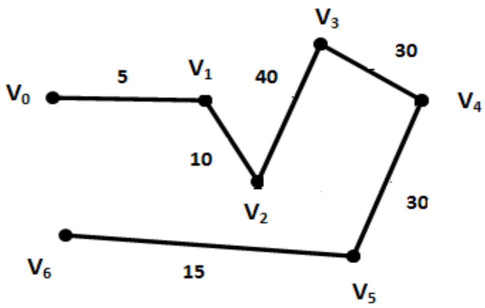


Fig.(2)

Then the neighborhood for condition (1) become:

V	N(V)
V_0	$\{V_1, V_2, V_4, V_5, V_6\}$
V_1	$\{V_2, V_4, V_5, V_6\}$
V_2	$\{V_4, V_5, V_6\}$
V_3	$\{V_4, V_5, V_6\}$
V_4	$\{V_5, V_6\}$
V_5	$\{V_6\}$
V_6	$\{V_6\}$

And for condition (2) will be:

V	N(V)
V_0	$\{V_1, V_2, V_4, V_5, V_6\}$
V_1	$\{V_2, V_4, V_5, V_6\}$
V_2	$\{V_4, V_5, V_6\}$
V_3	$\{V_4, V_5, V_6\}$
V_4	$\{V_5, V_6\}$
V_5	$\{V_6\}$
V_6	$\{V_6\}$

Note: From the previous example we find that neighborhood of $V_6 \leq$ neighborhood of $V_5 \leq$ n. $V_4 \leq$ n. $V_3 \leq$ n. $V_2 \leq$ n. V_1 .

Example 2 :

Consider a graph shown in Fig.(3) , if the condition is (The neighborhood for any vertex is all vertices belongs to shortest path From a to g) we have:

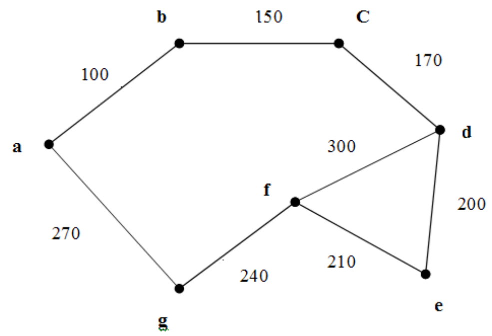


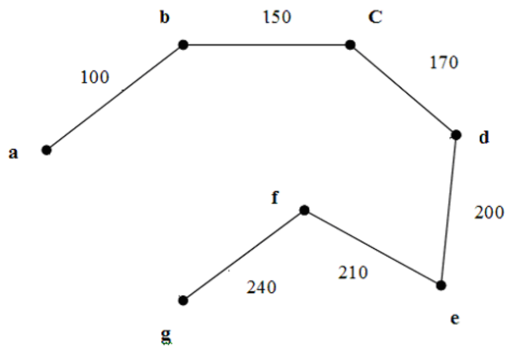
Fig.(3)

V	N(V)
a	{ b , c , d , f , g}
b	{ c , d , f , g}
c	{ d , f , g}
d	{f , g}
e	{f , g}
f	{ g }
g	{ g}

But if the condition is: all vertices belongs to longest path from a tog we have:

V	N(V)
a	{ b,c,d,e,f,g }
b	{c,d,e,f , g}
c	{ d, e ,f , g}
d	{ a , b ,c, g}
e	{a,b,c,d,g}
f	{a,b,c,d,e,g }
g	{ g }

If we find minimum spanning tree for graph Fig.(4) (By Kruskal's algorithm) and compute the previous computations we obtain



:

Fig.(4)

N

V	N(V)
a	{ b,c,d,e,f,g }
b	{c,d,e,f , g}
c	{ d, e ,f , g}
d	{ e,f, g}
e	{f, g }
f	{ g }
g	{ g }

eighborhood for condition 1:
And for condition 2:

V	N(V)
a	{ b,c,d,e,f,g }
b	{c,d,e,f , g}
c	{ d, e ,f , g}
d	{ e,f, g}
e	{f,g}
f	{ g }
g	{ g }

With this observation, we state the following theorem:

Theorem 1:

For conditional neighborhood which describe the algorithm for graph such as , shortest and longest path , if we compute the neighborhood for each vertex after finding minimum spanning tree for graph we find that:
Neighborhood computed for shortest path condition equal to neighborhood computed for longest path condition.

Proof:

Suppose G is connected weighted graph with vertices from V_0 to V_n , if G has at least one circuit , then shortest path from V_0 to V_n must contain at least two edges of that circuit with least weights , and longest path from V_0 to $V_n$must contain at least two edges of that circuit with long weights which varies from shortest path and the neighborhood for each vertex must change.
But if G is circuit_free , then G is its own spanning tree we have , then the shortest and longest path from V_0 to V_n are the same which imply that the neighborhood of each vertex is the same.

Type (2): The condition which describes the geometric classification of graph:
In this type we can put conditions on a graph such as ,
1.Condition depends on degree of vertices.
2. Condition depends on curvature of graph.
3. Condition depends on torsion of graph. And so on.
We will illustrate some examples in this type and compute the algorithm for each example.

1. For the condition depends on the degree of vertex:

Example 3:
Consider a graph as shown in Fig.(5) , if the condition is (The neighborhood of the vertex is all vertices with degree 3 we have:

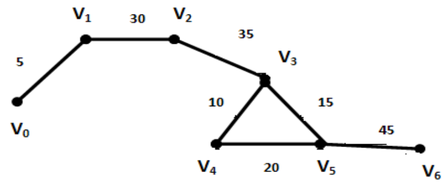


Fig.(5)

The neighborhood for each vertex will be:

V	N(V)
V_0	Ø
V_1	Ø
V_2	Ø
V_3	{V_2 , V_4 , V_5 }
V_4	Ø
V_5	{ V_3 V_4 , V_6 }
V_6	Ø

But if the neighborhood is the vertices with degree 2 we have:

V	N(V)
V_0	Ø
V_1	{ V_0 , V_2 }
V_2	{V_1 , V_3}
V_3	Ø
V_4	{ V_3,V_5 }
V_5	Ø
V_6	Ø

Finally, if the neighborhood is vertices with degree 1 we have:

V	N(V)
V_0	{ V_1 }
V_1	Ø
V_2	Ø
V_3	Ø
V_4	Ø
V_5	Ø
V_6	{ V_5 }

If we find minimum spanning tree for graph (By Kruskal's algorithm) Fig.(6) , the neighborhood for each example will be changed as follows:

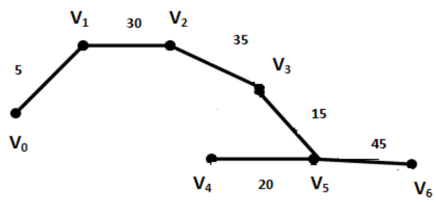


Fig.(6)

V	N(V)
V_0	{ V_1 }
V_1	Ø
V_2	Ø
V_3	Ø
V_4	{V_5}
V_5	Ø
V_6	{ V_5 }

Example 4:

Consider a graph shown in Fig.(7) , if the neighborhood for each vertex are all vertices with degree 1 we obtain:

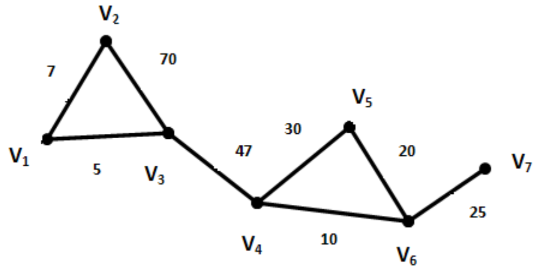


Fig.(7)

V	N(V)
V_1	Ø
V_2	Ø
V_3	Ø
V_4	Ø
V_5	Ø
V_6	Ø
V_7	{ V_6 }

But if we applying the algorithm to find minimum spanning tree the neighborhood will be changed:

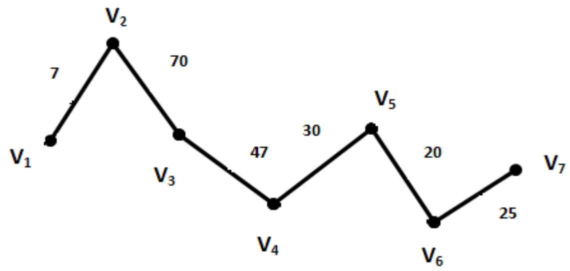


Fig.(8)

V	N(V)
V_1	{V_2}
V_2	Ø
V_3	Ø
V_4	Ø
V_5	Ø
V_6	Ø
V_7	{ V_6 }

Lemma 1:
For conditional neighborhood which describe the geometric classification of graphsuch as the degree of vertex , the neighborhood of every vertex changed before and after applying the algorithm.

2. For the condition depend on curvature of graph:

Note: We suppose that the positive curvature is in the form 

And the negative curvature is in the form 

Example 5:
For graph shown in Fig.(9), if the neighborhood for each vertex is all vertices with positive curvature we have:

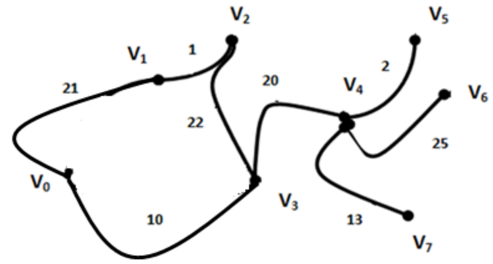


Fig.(9)

V	N(V)
V_0	{ V_1 , V_3}
V_1	{ V_0 , V_2}
V_2	{ V_1 , V_3}
V_3	{ V_2 }
V_4	{ V_7 }
V_5	Ø
V_6	Ø
V_7	{V_4}

But if the condition is all vertices with negative curvature then:

V	N(V)
V_0	Ø
V_1	Ø
V_2	Ø
V_3	{ V_4 }
V_4	{ V_3, V_5}
V_5	{V_4}
V_6	{V_4}
V_7	Ø

Example 6:
For graph shown in Fig.(10), compute the neighborhood for each vertex if the condition is all vertices with negative curvature.

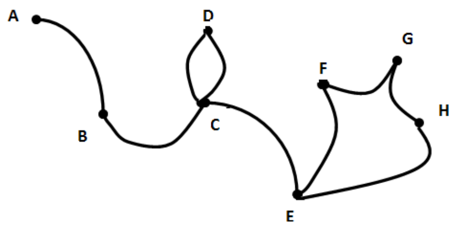


Fig.(10)

V	N(V)
A	{B}
B	Ø
C	{D,E}
D	{ C}
E	{ F}
F	{E}
G	Ø
H	Ø

3. For the condition depend on the torsion of graph:
Example 7 :
Consider a graph as shown in Fig.(11), if its vertices V_1 , V_2 , V_3 are in R^1 while vertices V_4 , V_5 are in R^2 , we can condition that the neighborhood of each vertex is the vertices in R^1 only , then we obtain:

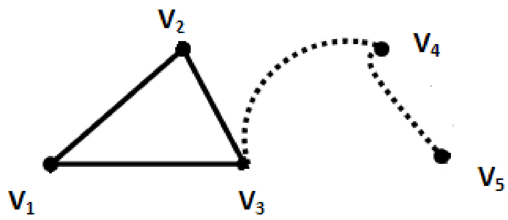


Fig.(11)

V	N(V)
V_1	$\{V_2, V_3\}$
V_2	$\{V_1, V_3\}$
V_3	$\{V_1, V_2\}$
V_4	Ø
V_5	Ø

Example 8:
For graph shown in Fig.(12), for which vertices A , B , C , D are in R^1 while the other vertices are in R^2 If we condition that the neighborhood of each vertex is the vertices in R^2 only , then we obtain:

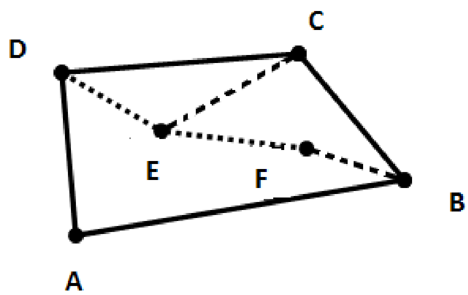


Fig.(12)

V	N(V)
A	Ø
B	{F}
C	{ E}
D	{ E}
E	{F , C , D}
F	{B , E}

Type (3) : Condition describes the algorithm and the geometric classification together:
In this type we will discuss conditions on a graph such as :
1. Condition describe the degree and shortest path together.
2. Condition describe the degree and longest path together.

Example 9:
Consider a graph as shown in Fig.(7) , we can compute the neighborhood for each vertex if we condition that its all vertices with degree 2 and belong to the shortest path from V_1 to V_7 we have:

V	N(V)
V_1	Ø
V_2	Ø
V_3	$\{V_4 , V_1\}$
V_4	$\{V_3 , V_6\}$
V_5	Ø
V_6	$\{V_4 , V_7\}$
V_7	Ø

But if the condition is its all vertices with degree 2 and belong to the longest path from V_1 to V_7 we have:

V	N(V)
V_1	Ø
V_2	$\{V_1, V_3\}$
V_3	$\{V_4, V_2\}$
V_4	$\{V_3, V_6\}$
V_5	$\{V_4, V_6\}$
V_6	$\{V_5, V_7\}$
V_7	Ø

But if we apply the algorithm to find minimum spanning tree the neighborhood will be changed:
For the first table:

V	N(V)
V_1	Ø
V_2	$\{V_1, V_3\}$
V_3	$\{V_2, V_4\}$
V_4	$\{V_3, V_5\}$
V_5	$\{V_4, V_6\}$
V_6	$\{V_5, V_7\}$
V_7	Ø

And for the second table:

V	N(V)
V_1	Ø
V_2	$\{V_1, V_3\}$
V_3	$\{V_2, V_4\}$
V_4	$\{V_3, V_5\}$
V_5	$\{V_4, V_6\}$
V_6	$\{V_5, V_7\}$
V_7	Ø

this implies the same results as in theorem 1.

References:

[1] Casey J,. *Exploring Curvature,*Wiesbaden, Germany, 1996.
[2] Gray, A. "Drawing Space Curves with Assigned Curvature." in Modern Differential Geometry of Curves and Surfaces with Mathematica, 2nd ed. Boca Raton, FL: CRC Press, pp. 222-224, 1997.
[3] Fournier,Jean-Claude,Graph Theory and applications with Exercises and problems,ISTE Ltd,2009.
[4] Susanna S.Epp, Discrete Mathematics With Application, Third Edition,Thomson Learning,Inc,2004.
[5] http://en.wikipedia.org/wiki/Algorithm .