Algorithm On Complete Graph And Their Folding

H. Ahmed[1,2] , Sh.Adel[1]

H.Ahmed (Heba Ahmed Khalaf)
Sh.Adel (Shereen Adel Abd El-Rhman)
1 Mathematics Department, college of Women, Ain Shams University, Cairo, Egypt
2Mathematics Department, college of arts and sciences, Prince asttam bin Abdulaziz University, Wade El-dawaser, Saudi Arabia

Email address
h.ashour@psau.edu.sa (author H.Ahmed)
shereen.3010@hotmail.com (author Sh.Adel)

Article History: Received: 11 January 2021; Accepted: 27 February 2021; Published online: 5 April 2021

Abstract: In this paper, introduce algorithm on complete graph K_4, when the graph weighted, and discusses the folding of algorithm graph of weighted complete graph K_4, the folding at some cases such as folding of the edges as all cases, and folding of the vertices, some theorems related to these result are obtained and prove of this theorems are obtained, also some life applications are introduced.

Keywords: Algorithm, weighted graph, complete graph, folding.

1. *Introduction and background* :

In mathematics a graph is intuitively a finite set of points in space, called the vertices of the graph, some pairs of vertices being joined by arcs, called the edges of the graph [2,4, 6].

The complete graph is a graph in which every two distinct vertices are joined by
One edge is called a complete graph. The complete graph on n vertices is usually denoted
By K_n, also K_n has exactly 1/2 n (n-1) edges. Fig.1 shows the complete graphs K_n for n=1, 2, 3, 4. The graph K_1 is sometimes called the "trivial graph" [3, 8].

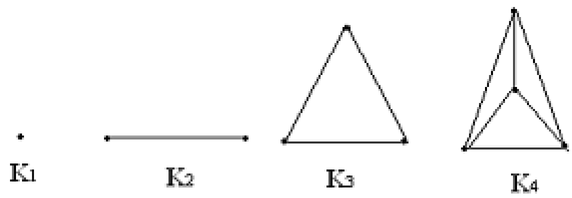

Fig.1

Fig.1 represents the complete graphs in some types.

In mathematics and computer science, an algorithm is an effective method expressed
As a finite list of well-defined instruction for calculating a function. In simple words an algorithm is a step-by-step procedure for calculations [10].

Graph algorithms are one of the oldest classes of algorithms and they have been studied for almost 300 years (in 1736) which solve problems related to graph theory. There are some of important algorithms for solving these problems [10].

Weighted graph is a graph for which each edge has an associated real number weight [4,7]. In Kruskal's algorithm, the edges of weighted graph are examined one by one in order. What will of increasing weight. At each stage the edge being examined is added to become the minimum spanning tree, provided that this addition doesn't create a circuit.

After n -1 edges have been added (where n is the number of vertices of the graph), these edges, together with the vertices of the graph form a minimum spanning tree for the graph [4, 7].

The weighted Kruskal's algorithm

How it works:
Input: G (weighted connected undirected graph with n vertices).
Algorithm body:
Build a sub graph T of G which consists of all the vertices of G with edges added at each stage.
1. Initialized T (empty graph) to have all vertices of G.
2. Let E be the set of all edges of G.
3. Find an edge e in E of least weight.
4. Delete e from E.
5. If addition of e to edge set of T doesn't produce a circuit.
Then add e to the edge set of T.
T is a minimum spanning tree of G [4, 7].

Minimal spanning tree for a weighted graph is a spanning tree that has at least possible total weight compared to all other spanning trees for the graphs.

It is minimum spanning tree in a connected weighted graph with $n \geq 1$ vertex carry out the following procedure:
Step (1) *Find an edge of least weight and call this* e^1. *Set* $k=1$
step (2) *While* $k < n$, *if there exists an edge e such that* $\{e\} \cup \{e^1,..., e^k\}$ *does not contain a circuit, let* e^{k+1} *be such an edge of least weight replace k by k+1, else output* $e^1, e^2,..., e^k$ *and stop.*
End while [4].

The field of folding began with S. A. Robertson's work , in 1977 , on isometric folding of Riemannian manifold M into N, which send any piecewise geodesic path in M to a piecewise geodesic path with the same length in N [1].

2. Main results:
El-Ghoul, M. submitted the work of a complete graphs and their folding, in this paper introduce algorithm on complete graph K_4 and the folding of algorithm graph of weighted complete graph K_4, the folding at some cases such as folding of the edges and folding of the vertices.

Algorithm on weighted complete graph K_4
Compute the algorithm on complete graph K_4 weighted by Kruscal's algorithm.
Let G be a complete graph K_4 with four vertices v_0, v_1, v_2, v_3 and six edges e_0, e_1, e_2, e_3, e_4, e_5 with then we can compute its by Kruscal's algorithm.
The weight of the complete graph is knowing, such as: G ($v_0 v_1 = e_0 = \varepsilon_0$, $v_1 v_2 = e_1 = \varepsilon_1$, $v_2 v_0 = e_2 = \varepsilon_2$, $v_0 v_3 = e_3 = \varepsilon_3$, $v_1 v_3 = e_4 = \varepsilon_4$, $v_2 v_3 = e_5 = \varepsilon_5$), where ($\varepsilon_0 > \varepsilon_1 > \varepsilon_2 > \varepsilon_3 > \varepsilon_4 > \varepsilon_5$), see Fig.2.

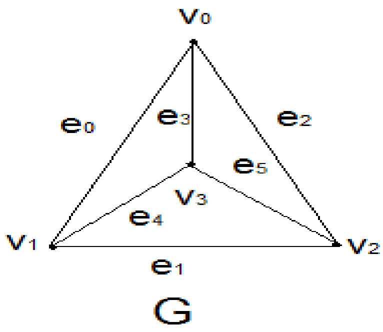

Fig.2

Fig.2 represents the weighted complete graph K_4.
By using Kruscal's algorithm the minimum spanning tree as follows in table 1.

Iteration no.	Considered	Weight	Action taken
1	v_2-v_3	ε_5	Added
2	v_1-v_3	ε_4	Added
3	v_0-v_3	ε_3	Added
4	v_2-v_0	ε_2	Not added
5	v_1-v_2	ε_1	Not added
6	v_0-v_1	ε_0	Not added

Table 1

Table 1 represent Kruscal's algorithm in a weighted complete graph K_4.
The minimum spanning tree is a tree, shown in Fig.3.

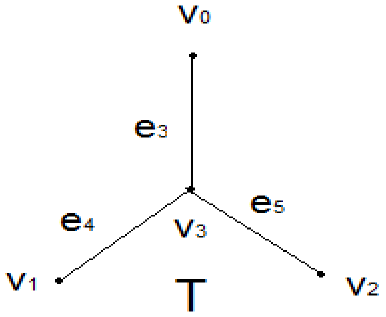

Fig.3

Fig.3 represents the result of weighted complete graph K_4 after Kruscal's algorithm.

Folding of the weighted complete graph K_4

In this section discusses the folding of the weighted complete graph K_4, the folding of algorithm on the weighted complete graph have many cases, such as edge to edge and vertex to vertex on folding of the weighted complete graph.

First: Folding of the edges:

Case (1)

Let f_1: $G_1 \rightarrow G_2$, f_2: $G_2 \rightarrow G_3$.
$f_1 (e_4) = e_0$, $f_1 (e_3) = e_2$, $f_1 (e_5) = e_1$, and $f_2 (e_1) = e_0$, $f_2 (e_2)$ = loop e_2 at v_2, see Fig.4 and Fig.5.

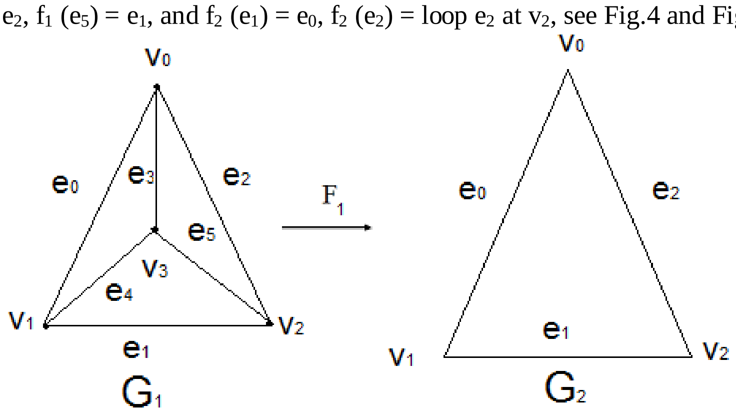

Fig.4

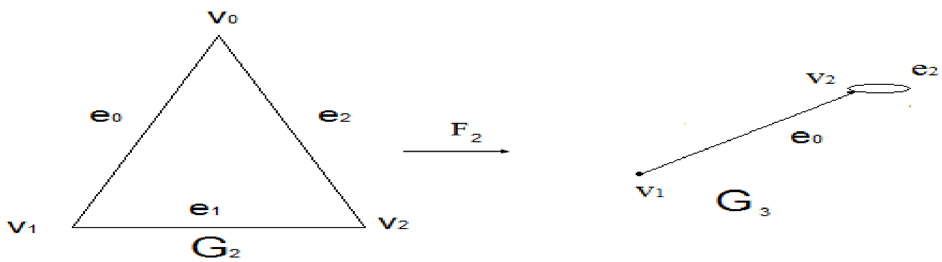

Fig.5

Fig.4 represents the folding (f_1), and Fig.5 represents the folding (f_2), of complete graph K_4, in case folding the edges (in case (1)).
Here, the weight of G_1 (v_0 $v_1 = e_0 = \varepsilon_0$, v_1 $v_2 = e_1 = \varepsilon_1$, v_2 $v_0 = e_2 = \varepsilon_2$, v_0 $v_3 = e_3 = \varepsilon_3$, v_1 $v_3 = e_4 = \varepsilon_4$, v_2 $v_3 = e_5 = \varepsilon_5$), G_2 (v_0 $v_1 = e_0 = \varepsilon_0$, v_1 $v_2 = e_1 = \varepsilon_1$, v_2 $v_0 = e_2 = \varepsilon_2$),
G_3 (v_1 $v_2 = e_0 = \varepsilon_0$, $e_2 = \varepsilon_2$ loop at v_2), where ($\varepsilon_0 > \varepsilon_1 > \varepsilon_2 > \varepsilon_3 > \varepsilon_4 > \varepsilon_5$).
By using Kruscal's algorithm the minimum spanning tree of the result of the folding as follows in table 2.

Iteration no.	Considered	Weight	Action taken
1	v_2-v_2	ε_2	Not added
2	v_2-v_1	ε_1	Added

Table 2

Table 2 represent Kruscal's algorithm in folding of the weighted complete graph K_4 in the first case of folding the edges.
The minimum spanning tree is a simple graph, shown in Fig.6.

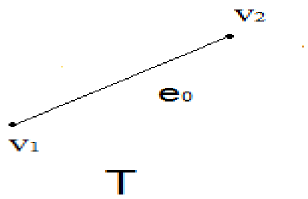

Fig.6

Fig.6 represents the result of Kruscal's algorithm to the folding of the weighted complete graph K_4 in the first case of folding the edges.

Case (2)

Let $f_1: G_1 \rightarrow G_2$, $f_2: G_2 \rightarrow G_3$.
$f_1(e_4) = e_0$, $f_1(e_5) = e_2$, $f_1(e_3)$ = loop e_3 at v_0, and $f_2(e_1) = e_0$, $f_2(e_2)$ = loop e_2 at v_0, see Fig.7 and Fig.8.

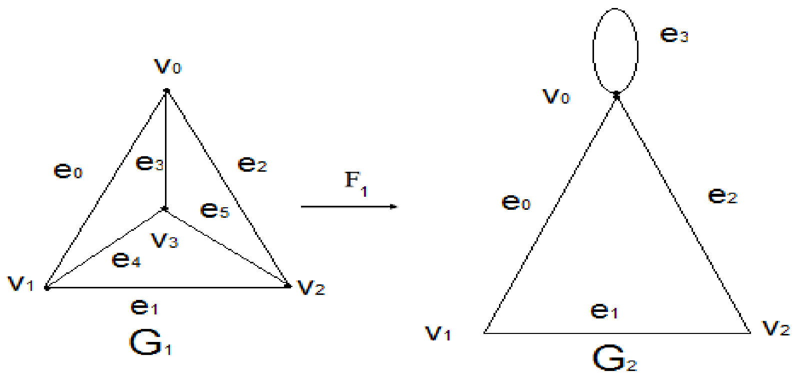

Fig.7

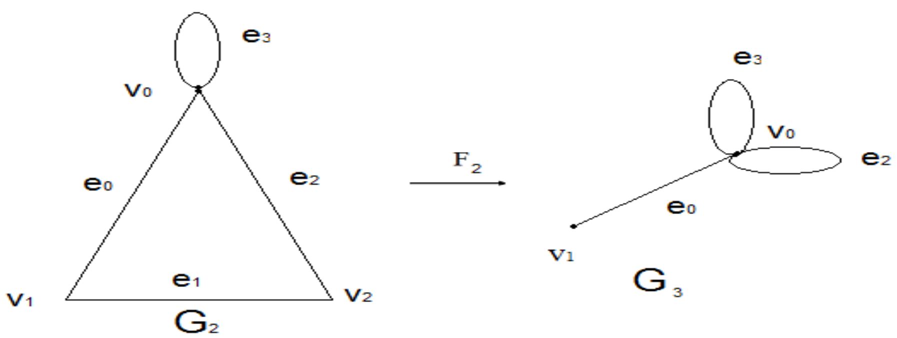

Fig.8

Fig.7 represents the folding (f_1), and Fig.8 represents the folding (f_2), of complete graph K_4, in case folding the edges (in case (2)).
Here, the weight of G_1 ($v_0 v_1 = e_0 = \varepsilon_0$, $v_1 v_2 = e_1 = \varepsilon_1$, $v_2 v_0 = e_2 = \varepsilon_2$, $v_0 v_3 = e_3 = \varepsilon_3$, $v_1 v_3 = e_4 = \varepsilon_4$, $v_2 v_3 = e_5 = \varepsilon_5$), G_2 ($v_0 v_1 = e_0 = \varepsilon_0$, $v_1 v_2 = e_1 = \varepsilon_1$, $v_2 v_0 = e_2 = \varepsilon_2$, $e_3 = \varepsilon_3$ loop at v_0), G_3 ($v_1 v_0 = e_0 = \varepsilon_0$, $e_2 = \varepsilon_2$ loop at v_0, $e_3 = \varepsilon_3$ loop at v_0), where ($\varepsilon_0 > \varepsilon_1 > \varepsilon_2 > \varepsilon_3 > \varepsilon_4 > \varepsilon_5$).
By using Kruscal's algorithm the minimum spanning tree of the result of the folding as follows in table 3.

Iteration no.	Considered	Weight	Action taken
1	v_0-v_0= e_3	ε_3	Not added
2	v_0-v_0= e_2	ε_2	Not added
3	v_0-v_1	ε_0	Added

Table 3

Table 3 represent Kruscal's algorithm in folding of the weighted complete graph K_4 in the second case of folding the edges.
The minimum spanning tree is a simple graph, shown in Fig.9.

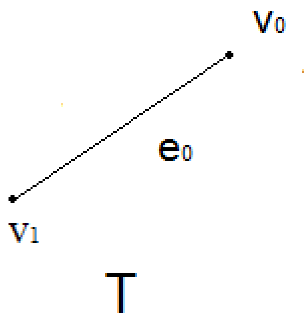

Fig.9

Fig.9 represents the result of Kruscal's algorithm to the folding of the weighted complete graph K_4 in the second case of folding the edges.

Case (3)
Let f_1: $G_1 \rightarrow G_2$, f_2: $G_2 \rightarrow G_3$.
$f_1(e_4) = e_3$, $f_1(e_5) = e_3$, $f_2(e_1) = e_0$, $f_2(e_3) = e_2$, $f_2(e_2) = e_0$, see Fig.10 and Fig.11.

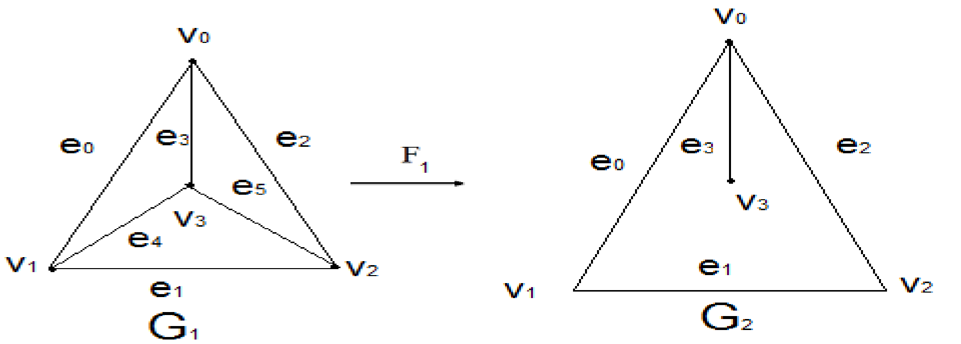

Fig.10

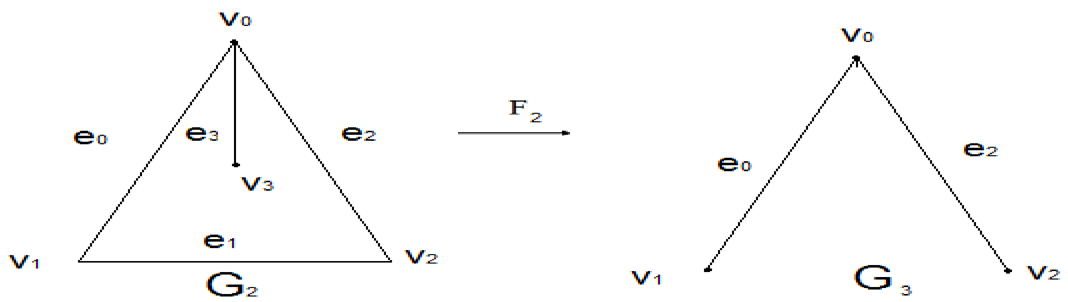

Fig.11

Fig.10 represents the folding (f_1), and Fig.11 represents the folding (f_2), of complete graph K_4, in case folding the edges (in case (3)).
Here, the weight of G_1 ($v_0 v_1 = e_0 = \varepsilon_0$, $v_1 v_2 = e_1 = \varepsilon_1$, $v_2 v_0 = e_2 = \varepsilon_2$, $v_0 v_3 = e_3 = \varepsilon_3$, $v_1 v_3 = e_4 = \varepsilon_4$, $v_2 v_3 = e_5 = \varepsilon_5$), G_2 ($v_0 v_1 = e_0 = \varepsilon_0$, $v_1 v_2 = e_1 = \varepsilon_1$, $v_2 v_0 = e_2 = \varepsilon_2$, $v_0 v_3 = e_3 = \varepsilon_3$), G_3 ($v_1 v_0 = e_0 = \varepsilon_0$, $v_0 v_2 = e_2 = \varepsilon_2$), where ($\varepsilon_0 > \varepsilon_1 > \varepsilon_2 > \varepsilon_3 > \varepsilon_4 > \varepsilon_5$).
By using Kruscal's algorithm the minimum spanning tree of the result of the folding as follows in table 4.

Iteration no.	Considered	Weight	Action taken
1	v_0-v_2= e_2	ε_2	Added
2	v_0-v_1= e_0	ε_0	Added

Table 4

Table 4 represent Kruscal's algorithm in folding of the weighted complete graph K_4 in the third case of folding the edges.
The minimum spanning tree is a simple graph, shown in Fig.12.

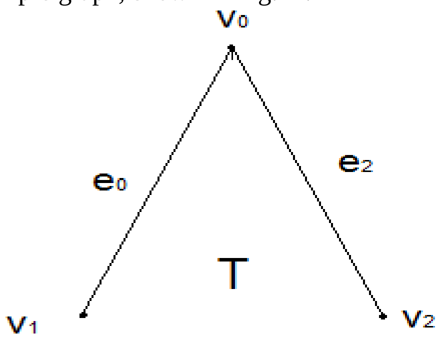

Fig.12

Fig.12 represents the result of Kruscal's algorithm to the folding of the weighted complete graph K_4 in the third case of folding the edges.

Second: Folding of the vertices:

Let $f_1: G_1 \rightarrow G_2$, $f_2: G_2 \rightarrow G_3$.
$f_1(v_3) = v_0$ and loop e_3, $f_1(v_3) = v_1$ and loop e_4, $f_1(v_3) = v_2$ and loop e_5, and $f_2(v_1) = v_2$ and loops e_3, e_5, $f_2(v_0) = v_2$ and loop e_2, see Fig.13 and Fig.14.

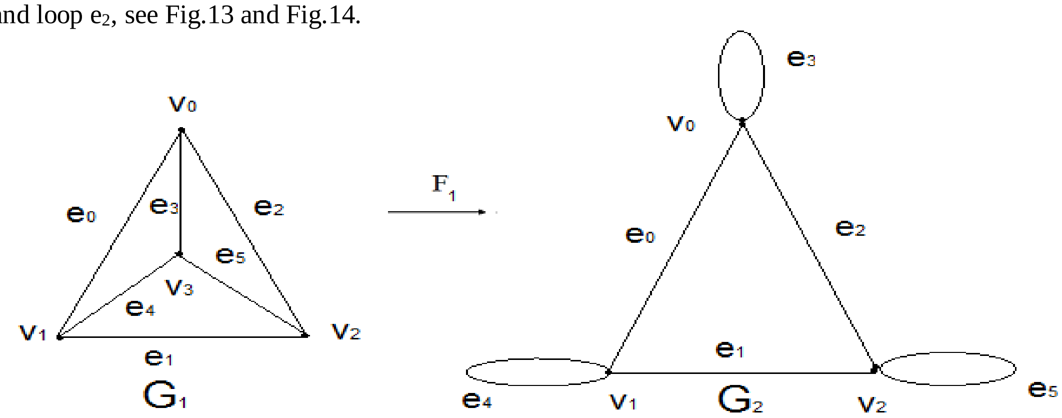

Fig.13

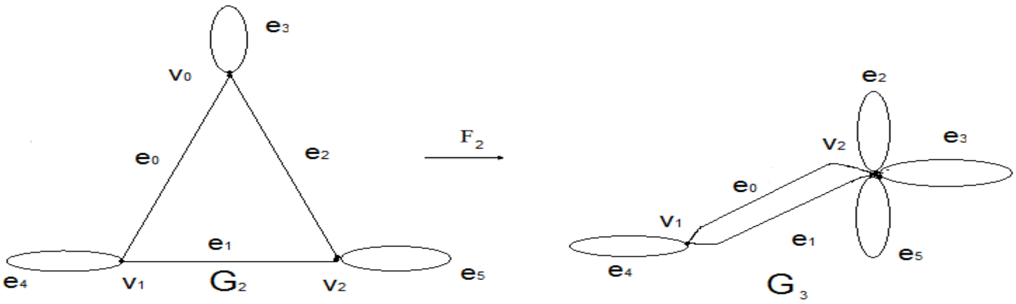

Fig.14

Fig.13 represents the folding (f_1), and Fig.14 represents the folding (f_2), of complete graph K_4, in case folding the vertices.

Here, the weight of G_1 ($v_0 v_1 = e_0 = \varepsilon_0$, $v_1 v_2 = e_1 = \varepsilon_1$, $v_2 v_0 = e_2 = \varepsilon_2$, $v_0 v_3 = e_3 = \varepsilon_3$, $v_1 v_3 = e_4 = \varepsilon_4$, $v_2 v_3 = e_5 = \varepsilon_5$), G_2($v_0 v_1 = e_0 = \varepsilon_0$, $v_1 v_2 = e_1 = \varepsilon_1$, $v_2 v_0 = e_2 = \varepsilon_2$, $e_3 = \varepsilon_3$ loop at v_0, $e_4 = \varepsilon_4$ loop at v_1, $e_5 = \varepsilon_5$ loop at v_2), G_3 ($v_1 v_2 = e_0 = \varepsilon_0$, $v_1 v_2 = e_1 = \varepsilon_1$, $e_2 = \varepsilon_2$ loop at v_2, $e_3 = \varepsilon_3$ loop at v_2, $e_5 = \varepsilon_5$ loop at v_2, $e_4 = \varepsilon_4$ loop at v_1), where ($\varepsilon_0 > \varepsilon_1 > \varepsilon_2 > \varepsilon_3 > \varepsilon_4 > \varepsilon_5$).
By using Kruscal's algorithm the minimum spanning tree of the result of the folding as follows in table 5.

Iteration no.	Considered	Weight	Action taken

1	v_2-v_2=e_5	ε_5	Not added
2	v_1-v_1=e_4	ε_4	Not added
3	v_2-v_2=e_3	ε_3	Not added
4	v_2-v_2=e_2	ε_2	Not added
5	v_2-v_1	ε_1	Added
6	v_1-v_2	ε_0	Not added

Table 5

Table 5 represent Kruscal's algorithm in folding of the weighted complete graph K_4 in case folding the vertices. The minimum spanning tree is a simple graph, shown in Fig.15.

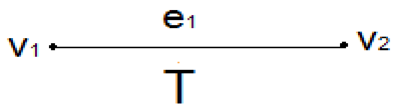

Fig.15

Fig.15 represents the result of Kruscal's algorithm to the folding of the weighted complete graph K_4 in case folding the vertices.

Theorem (1)

The folding of algorithm on weighted complete graph K_4 goes to weighted simple graph

Proof

The proof is clear from the above discussion.

3. Application in life:

1- Electrical connections, which represent weighted complete graph K_4, and also represent the folding of algorithm on weighted complete graph, see Fig.16 which represents a circuit.

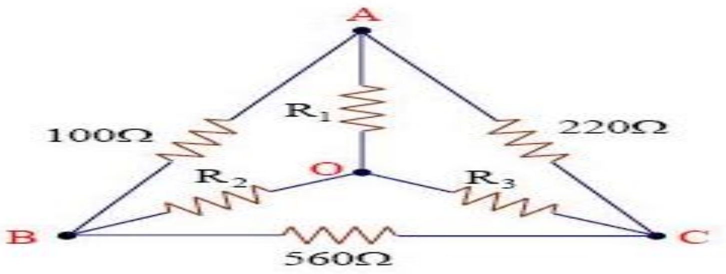

Fig.16

Fig.16 represents electrical connections as application of weighted complete graph K_4.

4. Conclusion:

In this paper done the algorithm of weighted complete graph K_4, and find the minimum spanning tree to this graph, and done the Folding of the weighted complete graph K_4 in cases edge to edge and vertex to vertex and conclusion some theorems, also some life applications were concluded.

References:

1. El-Ghoul, M.: Folding of fuzzy graphs and fuzzy spheres, Fuzzy Sets and systems, Germany, 58(1993) 355-363.
2. El-Ghoul, M& Ahmed, H.: Variation of Algorithm on Graphs and their Folding, Journal of Mathematical Archive. India.2015.
3. Gross J.L.: Tucker T. W.: Topological graph theory. Jon Willey& Sons, Inc, Canada. 1987.
4. Fournier, J.C. "Graph Theory and applications with Exercises and Problems", ISTE Ltd, 2009.
5. Giblin P.J.: Graphs, surfaces and homology, an Introduction to algebraic topology. Chapman and Hall Ltd, London. 1977.
6. Robin J.: Introduction to graph theory. Longman. 1972.
7. Susanna S. Epp, Discrete Mathematics with Application, Third Edition, Thomson Learning, Inc. 2004.
8. White A.T.: Graphs. : Groups and surfaces. Amsterdam, North-Holland Publishing Company. 1973.
9. Wilson R.J.:Watkins J.J.:Graph, an introductory approach, a first course in discrete mathematics. Jon Wiley and Sons Inc, Canda 1999.
10. http://www.softpanorama.org/Algorithms/graph_algorithms.shtml#History.