Transactions on Networks and Communications - Vol. 9, No. 4
Publication Date: August, 25, 2021
DOI: 10.14738/tnc.94.10754.
Bagona, P. L., & Fendji, J. L. K. E. (2021). Fault-Tolerant Placement of Additional Mesh Nodes in Rural Wireless Mesh Networks: A Minimum Steiner Tree Based Centre of Mass With Bounded Edge Length. Transactions on Networks and Communicaitons, 9(4). 39-50.

Fault-Tolerant Placement of Additional Mesh Nodes in Rural Wireless Mesh Networks: A Minimum Steiner Tree Based Centre of Mass With Bounded Edge Length

Patience Leopold Bagona
Mathematics and Computer Science Department
Faculty of Science, University of Ngaoundere
Ngaoundere, Cameroon

Jean Louis Kedieng Ebongue Fendji
Computer Engineering Department
University Institute of Technology
University of Ngaoundere, Cameroon

ABSTRACT

Wireless mesh networks are presented as an attractive solution to reduce the digital divide between rural and developed areas. In a multi-hop fashion, they can cover larger spaces. However, their planning is subject to many constraints including robustness. In fact, the failure of a node may result in the partitioning of the network. The robustness of the network is therefore achieved by carefully placing additional nodes. This work tackles the problem of additional nodes minimization when planning bi and tri-connectivity from a given network. We propose a vertex augmentation approach inspired by the placement of Steiner points. The idea is to incrementally determine cut vertices and bridges in the network and to carefully place additional nodes to ensure connectivity, bi and tri-connectivity. The approach relies on an algorithm using the centre of mass of the blocks derived after the partitioning of the network. The proposed approach has been compared to a modified version of a former approach based on the Minimum Steiner Tree. The different experiments carried out show the competitiveness of the proposed approach to connect, bi-connect, and tri-connect the wireless mesh networks.

Keywords: Biconnectivity, Triconnectivity, Wireless Mesh Networks, Minimum Steiner Tree, Centre of Mass, Bounded Edge.

INTRODUCTION

Wireless mesh networks (WMNs) are presented as appealing alternatives to connect the unconnected in difficult to wire areas [1,2]. They allow wireless devices to connect next to each other dynamically and/or statically without a central hierarchy, forming a mesh-like structure [3]. Mesh technology also allows the connection of new relays (MR) without the need for tedious manual network configuration. Their emergence is the result of their rapid evolution, and cost effectiveness. Despite their advantages, wireless mesh networks remain fragile since the failure of a node may result in the partitioning of the network. Especially in rural areas where there is a lack of technical skills, a network partitioning may last weeks or months.

Despite the efforts to design energy efficient routing protocols to improve the lifetime of nodes, especially in rural areas where the network is powered by limited energy sources[4], a node can still fail due to other raisons. To increase the lifetime of the network it is therefore important to increase the robustness of the network by ensuring a bi-connectivity or a triconnectivity of the network.

Increasing the connectivity degree of a network is not cost-free. In the presence of a limited budget, it is important to reduce the number of additional nodes to achieve the targeted degree of connectivity. This is a challenging problem that has attracted the attention of academics during the last decades. Usually, a network is assimilated to a graph with nodes and links in the network representing vertices and edges respectively in the corresponding graph. Then, algorithms proposed in the graph theory are used to propose solutions.

Several algorithms have been proposed in the literature to solve connectivity problems in graph. One of the oldest papers dealing with connectivity appeared in 1976, when Eswaran and Tarjan [5] considered the problem of adding a set of edges with minimum weight to a graph in order to reach a given connectivity degree. Like [6] the goal of their work was to solve biconnectivity problems. Later, biconnectivity was addressed in [7–10], which rely some on ear decomposition based on Depth First Search, and others on greedy or GRASP algorithms. The work in [10] considers the space as a grid and additional nodes are added depending on their position to the nearest node, based on a modified version of the Breadth First Search algorithm and simple arithmetic calculations on trees and on the Hamiltonian cycle.

In a real network deployment, two nodes can communicate only if they are located within their transmission range. If the distance between two nodes is greater than their transmission range, no link (edge in graph theory) can be established. Despite the plethora of works on the design of Wireless Mesh networks in rural regions [11–16], most works just focus on the simple connectivity of the network. The work in [10] takes a connected graph, and tries to design a biconnected one through node augmentation. But it considers a grid with predefined location of nodes.

This paper deals with connectivity in wireless mesh networks, considering the transmission range of the equipment while generalizing the placement problem by imposing no constraint on the spatial arrangement of the equipment. The objective is to propose an efficient node augmentation method for biconnectivity and triconnectivity problems in wireless mesh networks. To achieve this, we propose an approach based on the center of mass using the Minimum Steiner Tree called the Minimum Steiner Tree based Centre of Mass with Minimum number of Steiner Point and bounded edge length (MST-CM-MSPB).

The rest of the paper is arranged as follows: in the second section we define and present some concepts necessary for the understanding of the reader. In the third section we present and discuss our robustness approach to provide bi and triconnectivity. In the fourth section we conduct experimentation and discuss our results. This paper ends a conclusion and directions for further work.

PRELIMINARIES

Definitions

A graph G is a pair $G = (V, E)$, where V is a set of vertices and E is a (multi)set of unordered pairs of vertices. The elements of E are called edges [17].

A directed graph or digraph G consists of a vertex set $V(G)$ and an edge set $E(G)$, where each edge is an ordered pair of vertices. We use the ordered pair (u, v) for the edge uv, where u is the tail and v is the head.

A graph $H = (U, F)$ is a subgraph of a graph $G = (V, E)$ if $U \subseteq V$ and $F \subseteq E$. If $U = V$ then H is called spanning.

A loop is an edge (v, v) for some $v \epsilon V$. An edge $e = (u, v)$ is a multiple edge if it appears several times in E. A graph is simple if it has neither loops nor multiple edges.

Connectivity

A walk in the graph G is a finite sequence of the form $v_{i0}, e_{j1}, v_{i1}, e_{j2} \ldots, e_{jk}, v_{ik}$, which consists of alternating vertices and edges of G. The walk starts at the initial vertex v_{i0} and ends at the terminal vertex v_{ik}. k is the length of the walk. A zero-length walk is just a single vertex v_{i0}. It is allowed to visit a vertex or go through an edge more than once. A walk is open if $v_{i0} \neq v_{ik}$. Otherwise, it is closed.

A trail is a path if any vertex is visited at most once except possibly the initial and terminal vertices when they are the same. A closed path is a circuit. We identify the paths and circuits with the subgraphs induced by their edges.

Theorem 1: Every vertex of G belongs to exactly one component of G. Similarly, every edge of G belongs to exactly one component of G.

Let the walk starting at u and ending at v be called an $u-v$ walk. u and v are connected if there is a $u-v$ walk in the graph (meaning there is also a $u-v$ path). A graph is connected if all the vertices are connected to each other.

Theorem 2: If the graph G has a vertex v that is connected to a vertex of the component G_1 of G, then v is also a vertex of G_1.

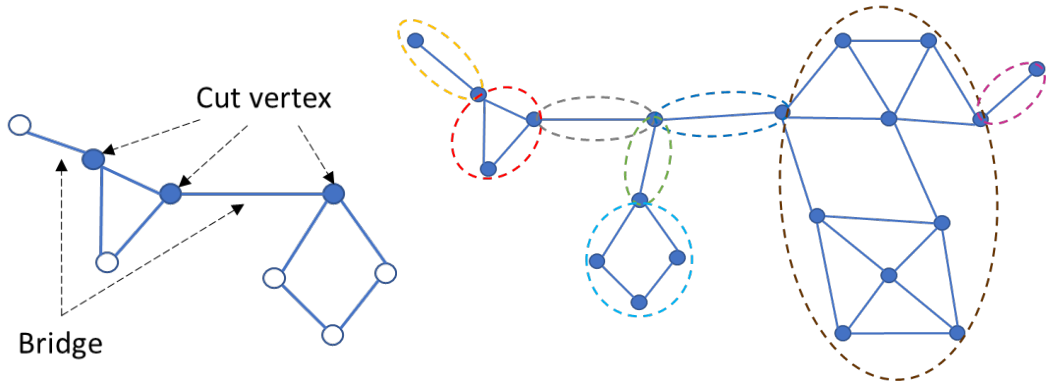
A **cut vertex** in a connected graph $G = (V, E)$ is a set $S \subseteq V$ such that $G \backslash S := G[V \backslash S]$ has more than one connected component.

A disconnecting set of edges is a set $F \subseteq E(G)$ such that $G \backslash F$ has more than one component. Given $S, T \subset V(G)$, the notation $[S, T]$ specifies the set of edges having one endpoint in S and the other in T. An edge cut is an edge set of the form $[S, S]$, where S is a non-empty proper subset of $V(G)$. A graph is k-edge-connected if every disconnecting set has at least n edges. The edge-connectivity of G, written $k'(G)$, is the minimum size of a disconnecting set. One edge disconnecting G is called a bridge.

A **block** of a graph G is a maximal connected subgraph of G that has no cut-vertex [25]. Suppose that G itself is connected and has no cut vertex. Then G is a block. Suppose that H is a block of G. Then, H is a graph which has no cut-vertex. However, H may contain vertices that are cut vertices of G.

An edge $e \in E$ is a **bridge** if its removal disconnects G. A graph is k-connected if every vertex cut has at least k vertices. The connectivity of G, written $k(G)$, is the minimum size of a cut vertex. In other words, $k(G)$ is the maximum k such that G is k-connected.

Figure 1 illustrates the concepts of cut vertex, bridge, and block.



(a) Cut vertices and bridges **(b) A graph with eight blocks**

Figure 1. Illustration of cut vertex, bridge, and block.

ROBUSTNESS APPROACH

General Idea

The proposed approach is illustrated in Figure 2. The first step is to verify the distance between nodes to check if they are in the transmission range to establish links. The next step is to check the connectivity of the given graph, in other terms, if the graph is composed of only one block. Otherwise, the different blocks are connected using the proposed connecting approach based on the combination of the Steiner Minimum Tree and the centre of mass. Next, the biconnectivity is checked by looking for cut vertices. If cut vertices exist, they are removed and then the graph is connected using the proposed connecting approach. Finally, the triconnectivity test is done. If there are bridges, then they are removed, and the remaining graph is connected using the same connecting approach.

EXPLANATION

Checking the connectivity, biconnectivity and triconnectivity

The CCDFS (Connectivity-Checking based DFS) algorithm presented in Algorithm1 is used to check the connectivity. The biconnectivity is checked by deleting one vertex at a time and checking the connectivity of the remaining graph using the CCDFS algorithm. It is done using the CheckingBiconnectivity algorithm presented in Algorithm3. Finally, the tri-connectivity of the graph is checked by deleting one edge at a time and checking the connectivity of the remaining graph. It is done using the CheckingTriconnectivity algorithm presented in Algorithm4.

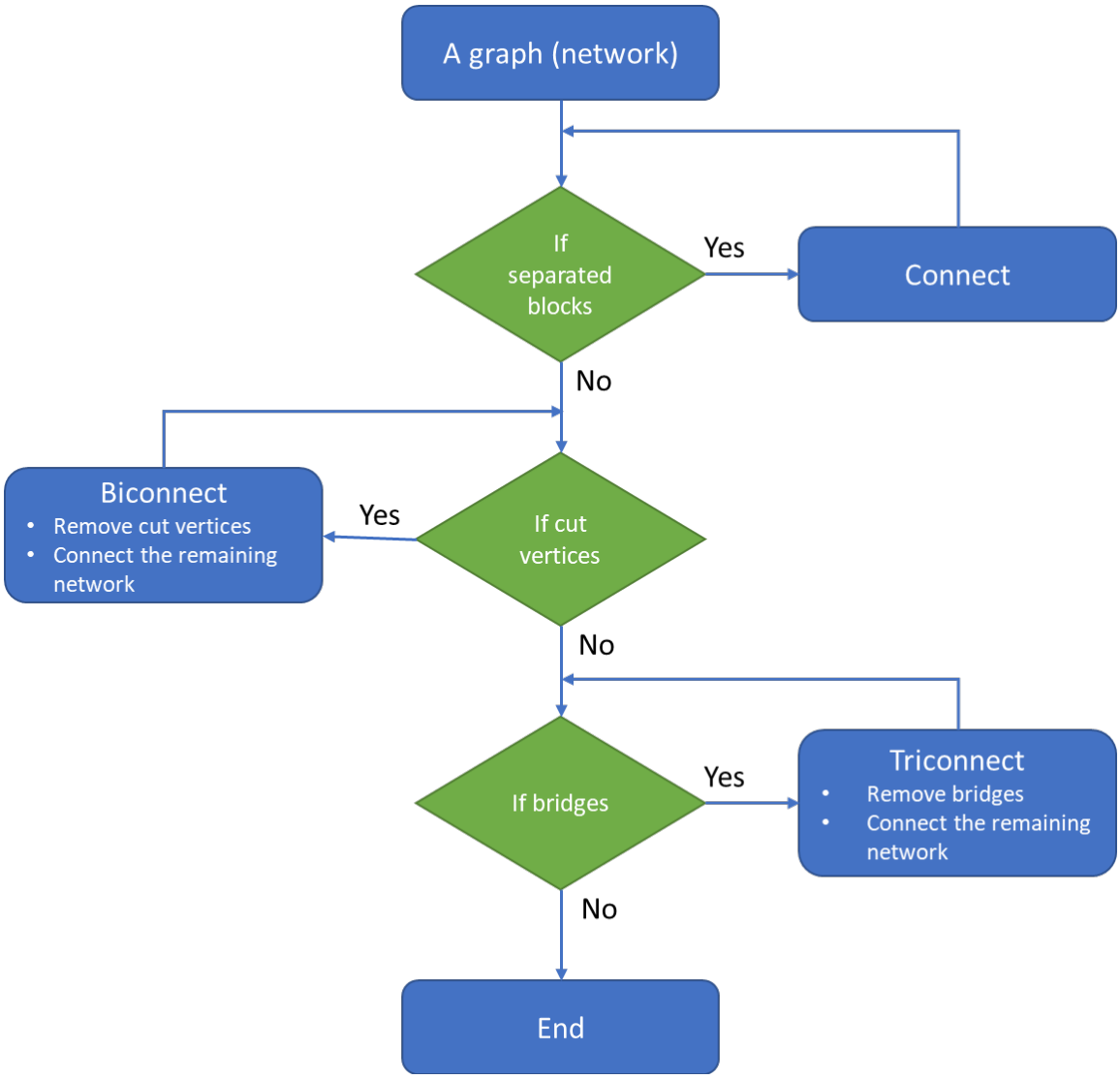


Figure 2. Robustness approach

Algorithm 1: CCDFS

Input: G : A Graph

Output: Ans: A Boolean to indicate if the graph is connected

```
1   Begin
2     Ans=True
3     For each vertex v in G do
4         If v is not visited then
5             ExploreTree(G,v)
6     End for
7     For each vertex v in G do
8         If v is not visited then
9             Ans=False
10            Break
11    End for
12    Return Ans
13  End
```

Algorithm 2: ExploreTree

Input: G : A Graph; s : A vertex
Output: G : A Graph; t : A vertex

```
1  Begin
2  | For each vertex t in the neighbourhood of v do
3  |     If t is not visited then
4  |         ExploreTree(G, t)
5  | End for
   | End
```

Algorithm 3: CheckingBiconnectivity

Input: G : A Graph
Output: Ans: A Boolean to indicate if the graph is Biconnected; CV: The set of cutting vertices

```
1  Begin
2  | Ans=True
3  | CV=∅
4  | For each vertex v in the of G do
5  |     G' = G\ {v}
6  |     If not CCDFS(G') then
7  |         Ans=False
9  |         CV = CV ∪ {v}
10 | End for
11 | Return Ans, CV
12 | End
```

Algorithm 4: CheckingTriconnectivity

Input: G : A Graph
Output: Ans: A Boolean to indicate if the graph is Triconnected; BR: The set of bridges

```
1  Begin
2  | Ans=True
3  | BR=∅
4  | For each edge e in the of G do
5  |     G' = G\ {e}
6  |     If not CCDFS(G') then
7  |         Ans=False
9  |         BR = BR ∪ {e}
10 | End for
11 | Return Ans, BR
12 | End
```

Minimum Steiner Tree based Centre of Mass with Minimum number of Steiner Point and bounded edge length (MST-CM-MSPB)

Following the approach presented in Figure 2, when the graph is unconnected, we directly connect the graph using the proposed MST-CM-MSPB algorithm. To biconnect the graph, we first remove the cut vertices and connect the remaining graph using the MST-CM-MSPB algorithm. In the same way, to triconnect the graph, we first remove the bridges and then use the MST-CM-MSPB algorithm to connect the remaining graph. The MST-CM-MSPB algorithm is presented in Algorithm 5.

Algorithm 5: MST-CM-MSPB

Input: G : An unconnected Graph; R: Maximal distance between two vertices (transmission range)

Output: G: Connected Graph

Begin

1. Associate each vertex v in G to his block B_i
2. If $|B| > 2$ then
   a. Compute the centre of mass CM_i of each block B_i
   b. Select an extremum X in the set of centres of mass CM
   c. Seek the 2 closest vertices Y and Z to X in the set of centres of mass CM
   d. Determine the Fermat point F of points X, Y, Z
   e. For each of the selected blocks, determine the closest vertex v_c to F
   f. For each v_c, establish link with F (Add new vertices to G subject to R
   g. Goto 1
3. If $|B| = 2$ then
   a. Seek the closest points X, Y between the two blocks
   b. Establish link between X, Y (Add new vertices to G subject to R)
4. Return G

End

R is the transmission range of router in Wireless networks. The Fermat point [18] is computed using equation 1. The establishment of links at the step 2.f and 3.b is done by checking the distance between the Fermat point and a closest point v_c (step 2.f) or between the closest points (step 3.b). if the distance d is less than the transmission range R, the points are connected. Otherwise, $\left\lfloor \frac{d}{R} \right\rfloor$ extra nodes will be added between the two nodes to connect. The MST-CM-MSPB algorithm is illustrated in Figure 3.

$$F = \frac{\frac{YZ}{\sin(\hat{X} + \frac{\pi}{3})}X + \frac{XY}{\sin(\hat{Y} + \frac{\pi}{3})}Y + \frac{ZX}{\sin(\hat{Z} + \frac{\pi}{3})}Z}{\frac{YZ}{\sin(\hat{X} + \frac{\pi}{3})} + \frac{XY}{\sin(\hat{Z} + \frac{\pi}{3})} + \frac{ZX}{\sin(\hat{Y} + \frac{\pi}{3})}} \quad (1)$$

Complexity Analysis

Let $G = (V, E)$ be the given graph. The complexity of the CCDFS algorithm is $O(|V| + |E|)$. The CheckingBiconnectivity algorithm allows to check the biconnectivity of G and to output the set of cut vertices. The complexity of this algorithm is $O(|V|^2 + |V||E|)$ .The CheckingTriconnectivity algorithm allows to check the triconnectivity of G and to output the set of bridges. The complexity of this algorithm is $O(|V||E| + |E|^2)$. Removing either cut vertices or bridges requires respectively $O(|V|)$ and $O(|E|)$. So, the worst complexity of this phase is $O\left(Max\left((|V|^2 + |V||E|), (|V||E| + |E|^2)\right)\right)$.

To analyse the complexity of the MST-CM-MSPB, will consider each component of the algorithm. Step 1 requires a worse complexity of $O(|V|^2 + |V||E|)$. First the computation of the Centre of Mass of a block requires $O(|V|)$. Steps 2.a and 2.b require $O(|V|)$. Step 2.c requires $O(|V|^2)$. Step 2.d requires $O(1)$. Step 2.e requires $O(|V|)$. Step 2.f requires $O(3)$. Step 3.a requires $O(|V|^2)$.

And finally Step 3.b requires $O(3)$. Since there is a loop on the set of vertices, the complexity if $O(|V|^3)$.

So, the complexity of the complete approach is $O(|V|^3)$.

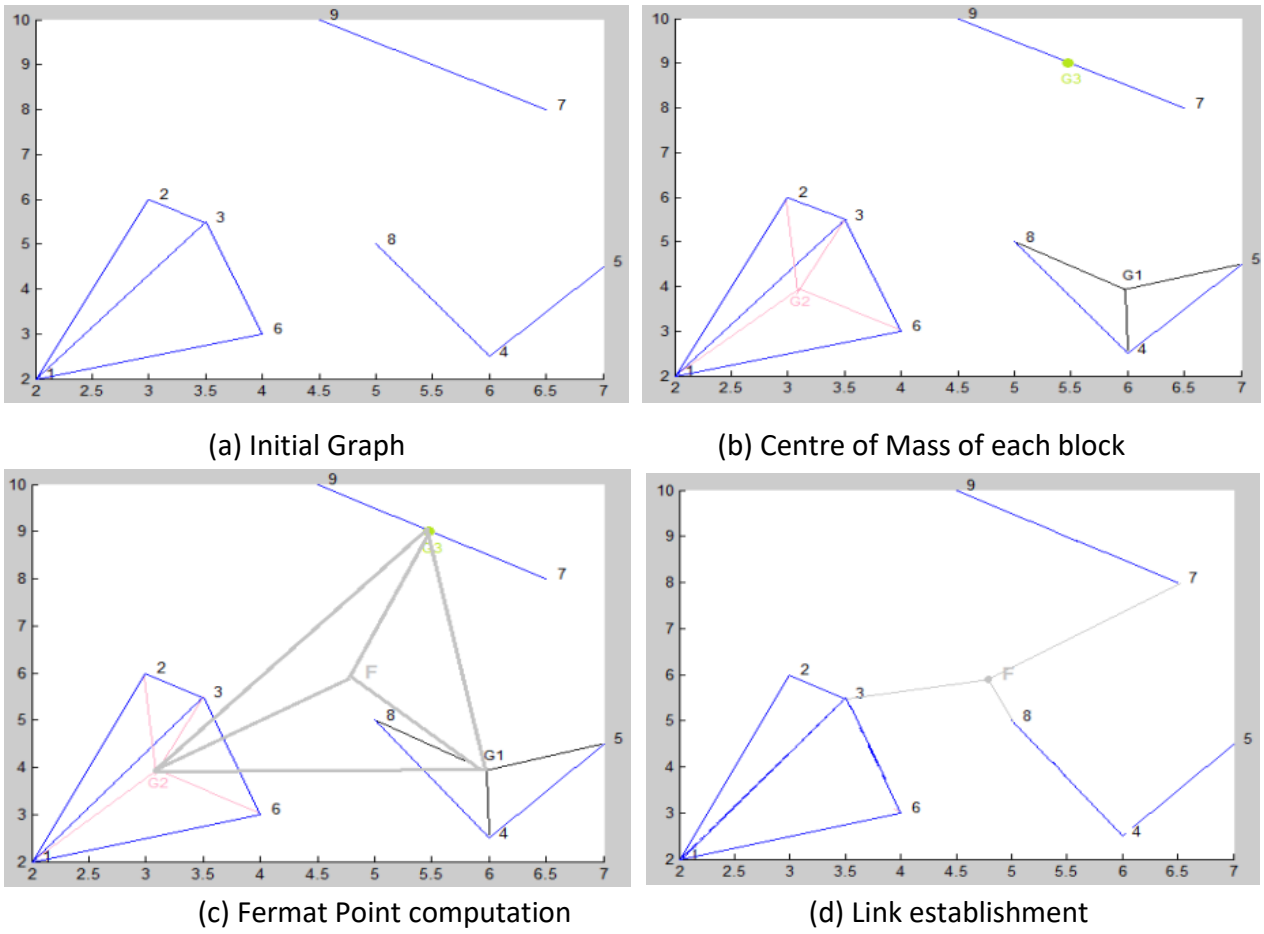


Figure 3. Illustration of MST-CM-MSPB algorithm

EXPERIMENTS AND RESULTS

Evaluation

The algorithm is evaluated on two sets of randomly generated networks: the set of sparse graphs and the set of dense graphs. We consider a graph as sparse if the number of edges is less than $\frac{|V|(|V|-1)}{8}$, otherwise we consider it as a dense graph. Each dataset is composed 30 graphs. The proposed MST-CM-MSPB approach is compared to a modified version of the MST-MSP algorithm whose original version has been proposed in [19] for relay sensor placement. The percentage of additional vertices (in this case routers) given by Equation 2 will be computed for each approach.

$$P = \frac{NumberAdditionalVertices}{InitialNumberVertices} \quad (2)$$

RESULTS

Results on sparse networks

Table 1 presents the results of the set of 30 sparse graphs. From this Table, MST-CM-MSPB and MST-MSP provide the same number of additional vertices for connecting, biconnecting and triconnecting graphs in most cases. However, in some instances, MST-CM-MSPB provides a smaller number of additional vertices. For example, MST-CM-MSPB requires only two vertices to biconnect the instance 13, compared to 4 vertices required by MST-MSP. The average percentage of additional vertices is given in Table 2. This Table shows the superiority of MST-CM-MSPB over MST-MSP.

Table 1: Results on the 30 generated sparse networks

		Connectivity		Biconnectivity		Triconnectivity	
Instance	**Vertices**	**MST-MSPM**	**MST-CM-MSPB**	**MST-MSPM**	**MST-CM-MSPB**	**MST-MSPM**	**MST-CM-MSPB**
1	5	4	**3**	3	3	2	**1**
2	3	1	1	2	2	-	-
3	22	1	1	7	7	1	1
4	21	1	1	3	3	1	1
5	21	2	2	3	3	3	3
6	5	1	1	2	2	1	1
7	10	2	2	4	**3**	-	-
8	12	1	1	3	3	-	-
9	24	1	1	2	2	-	-
10	27	1	1	4	4	2	2
11	27	2	2	1	1	-	-
12	8	4	4	5	5	-	-
13	5	5	**4**	4	**2**	3	**1**
14	8	2	2	3	3	-	-
15	24	2	2	6	6	1	1
16	17	1	1	1	**-**	-	-
17	119	2	2	1	1	-	-
18	37	1	1	3	3	2	2
19	169	2	2	-	-	-	-
20	156	3	3	-	-	-	-
21	138	1	1	-	-	-	-
22	55	4	**2**	-	-	-	-
23	82	1	1	2	2	-	-
24	81	4	**3**	2	2	-	-
25	42	4	4	-	-	-	-
26	198	3	3	2	2	-	-
27	184	1	1	-	-	-	-
28	98	2	2	1	1	-	-
29	90	2	2	-	-	-	-
30	16	1	1	2	2	-	-

Table 2: Average percentage of additional vertices for sparse networks

	Connectivity		Biconnectivity		Triconnectivity	
	MST-MSPM	**MST-CM-MSPB**	**MST-MSPM**	**MST-CM-MSPB**	**MST-MSPM**	**MST-CM-MSPB**
Additional Vertices	14.29%	12.80%	18.61%	16.74%	5.35%	3.35%

Results on dense network

Table 3 presents the results of the set of 30 dense graphs. From this Table, MST-CM-MSPB and MST-MSP provide the same number of additional vertices for connecting, biconnecting and triconnecting graphs in most cases, as it was observed with sparse graphs. Although MST-CM-MSPB provides a smaller number of additional vertices in some instances, in instance 10, MST-MSP provided a smaller number of additional vertices to connect the graph. The average percentage of additional vertices is given in Table 4. This Table shows the competitiveness of MST-CM-MSPB over MST-MSP even on dense networks.

Table 3: Results on the 30 generated dense networks

		Connectivity		Biconnectivity		Triconnectivity	
Instance	**Vertices**	**MST-MSPM**	**MST-CM-MSPB**	**MST-MSPM**	**MST-CM-MSPB**	**MST-MSPM**	**MST-CM-MSPB**
1	91	1	1	1	1	-	-
2	140	1	1	-	-	-	-
3	16	1	1	1	1	-	-
4	82	1	1	-	-	-	-
5	195	1	1	-	-	-	-
6	33	1	1	-	-	-	-
7	67	1	1	-	-	-	-
8	27	3	**2**	1	1	-	-
9	82	3	3	-	-	-	-
10	42	**3**	4	-	-	-	-
11	90	3	3	-	-	-	-
12	7	3	**2**	-	-	-	-
13	84	3	3	1	1	1	1
14	44	3	3	2	**1**	-	-
15	122	1	1	-	-	-	-
16	10	1	1	1	1	-	-
17	92	3	3	-	-	-	-
18	13	3	**2**	1	-	-	-
19	93	3	3	-	-	-	-
20	50	3	3	-	-	-	-
21	86	3	3	-	-	-	-
22	93	3	3	-	-	-	-
23	54	3	**2**	-	-	-	-
24	70	3	3	-	-	-	-
25	82	3	3	-	-	-	-

26	41	2	2	-	-	-	-
27	77	3	3	-	-	-	-
28	53	3	3	-	-	-	-
29	82	2	2	1	1	-	-
30	43	1	1	-	-	-	-

Table 4: Average percentage of additional vertices for dense networks

	Connectivity		**Biconnectivity**		**Triconnectivity**	
	MST-MSPM	**MST-CM-MSPB**	**MST-MSPM**	**MST-CM-MSPB**	**MST-MSPM**	**MST-CM-MSPB**
Additional Vertices	5.95%	5.12%	1.19%	0.86%	0.04%	0.04%

CONCLUSION

Robustness of Wireless Mesh Networks in rural areas remains a real challenge that has not yet received enough attention. This work introduces a new approach to connect, biconnect and triconnect such networks. The proposed approach called Minimum Steiner Tree based Centre of Mass with Minimum number of Steiner Point and bounded edge length (MST-CM-MSPB) is based on the calculation of the centre of mass and the Minimum Steiner Tree. The evaluation on randomly generated networks demonstrates the competitiveness of MST-CM-MSPB.

The main limitation of this work that should be addressed in future works is the non-consideration of the interference. In fact, adding extra nodes is not without consequence on the quality of existing links in the network. So, a trade-off should be found between the quality of links that affects the quality of service of the network and the robustness of the network.

References

[1] J.L.F.K. Ebongue, Rethinking Network Connectivity in Rural Communities in Cameroon, in: ArXiv Prepr. ArXiv150504449, Lilongwe, Malaw, 2015. https://arxiv.org/ftp/arxiv/papers/1505/1505.04449.pdf.

[2] J.L.E.K. Fendji, J.M. Nlong, Rural wireless mesh network: A design methodology, ArXiv Prepr. ArXiv150408213. (2015).

[3] I.F. Akyildiz, X. Wang, Wireless mesh networks, John Wiley & Sons, 2009.

[4] S.D. Samo, J.L.E.K. Fendji, Evaluation of Energy Consumption of Proactive, Reactive, and Hybrid Routing Protocols in Wireless Mesh Networks Using 802.11 Standards, J. Comput. Commun. 06 (2018) 1–30. https://doi.org/10.4236/jcc.2018.64001.

[5] K.P. Eswaran, R.E. Tarjan, Augmentation problems, SIAM J. Comput. 5 (1976) 653–665.

[6] A. Rosenthal, A. Goldner, Smallest augmentations to biconnect a graph, SIAM J. Comput. 6 (1977) 55–66.

[7] G. Calinescu, P.-J. Wan, Range assignment for biconnectivity and k-edge connectivity in wireless ad hoc networks, Mob. Netw. Appl. 11 (2006) 121–128.

[8] R.E. Moraes, C.C. Ribeiro, Power optimization in ad hoc wireless network topology control with biconnectivity requirements, Comput. Oper. Res. 40 (2013) 3188–3196.

[9] Y. Zhou, X. Cai, Node Placement in Wireless Sensor Network to Achieve Biconnectivity, in: 2009 Int. Conf. Netw. Secur. Wirel. Commun. Trust. Comput., IEEE, 2009: pp. 309–314.

[10] D. Benyamina, A. Hafid, M. Gendreau, On the design of bi-connected wireless mesh network infrastructure with QoS constraints, in: GLOBECOM 2009-2009 IEEE Glob. Telecommun. Conf., IEEE, 2009: pp. 1–8.

[11] J.L. Fendji, C. Thron, J.M. Nlong, Simulated Annealing approach for mesh router placement in rural Wireless Mesh Networks, in: 7th Int. Conf. AFRICOMM, 2015: pp. 15–16.

[12] J.L.E.K. Fendji, C. Thron, J.M. Nlong, A metropolis approach for mesh router nodes placement in rural wireless mesh networks, ArXiv Prepr. ArXiv150408212. (2015).

[13] J.L.F.K. Ebongue, C. Thron, J.M. Nlong, Mesh Router Nodes placement in Rural Wireless Mesh Networks, ArXiv Prepr. ArXiv150503332. (2015).

[14] J.L.K.E. Fendji, C. Thron, A Simulated Annealing Based Centre of Mass (SAC) Approach for Mesh Routers Placement in Rural Areas, Int. J. Oper. Res. Inf. Syst. IJORIS. (2020). www.igi-global.com/article/a-simulated-annealing-based-centre-of-mass-sac-approach-for-mesh-routers-placement-in-rural-areas/243420 (accessed June 12, 2020).

[15] A. Ajayi, U. Roedig, C. Edwards, N. Race, A survey of rural Wireless Mesh Network (WMN) deployments, in: 2014 IEEE Asia Pac. Conf. Wirel. Mob., IEEE, 2014: pp. 119–125.

[16] A. Ting, D. Chieng, K.H. Kwong, I. Andonovic, Optimization of heterogeneous multi-radio multi-hop rural wireless network, in: 2012 IEEE 14th Int. Conf. Commun. Technol. ICCT, 2012: pp. 1159–1165. https://doi.org/10.1109/ICCT.2012.6511372.

[17] L.-H. Hsu, C.-K. Lin, Graph theory and interconnection networks, CRC press, 2008.

[18] J. Park, A. Flores, Fermat's point from five perspectives, Int. J. Math. Educ. Sci. Technol. 46 (2015) 425–441.

[19] X. Cheng, D.-Z. Du, L. Wang, B. Xu, Relay sensor placement in wireless sensor networks, Wirel. Netw. 14 (2008) 347–355.