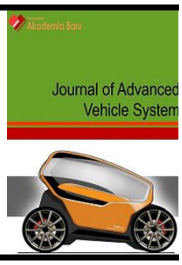
Penerbit Akademia Baru

Journal of Advanced Vehicle System

Journal homepage: www.akademiabaru.com/aravs.html
ISSN: 2550-2212



Two efficient approaches for generating topologies of ad-hoc networks

Open Access

Ibrahim S. Fayed [1,*], Mokhtar A. A. Mohamed [1], Gamal Attiya [1], Nawal A. El-Fishawy [1]

[1] Department of Computer Science & Engineering, Faculty of Electronic Engineering Menoufia University, Egypt

ARTICLE INFO

Article history:
Received 22 December 2016
Received in revised form 28 January 2017
Accepted 28 February 2017
Available online 6 March 2017

ABSTRACT

The field of Ad hoc networks has seen a rapid expansion of visibility and work due to the rapid increase of inexpensive, widely available wireless devices and the network community's interest in mobile computing. Such networks are taking special great attention from research community due to their wide range of applications. However, because of the expensive and difficulty of real experiments, simulation technique is the primary methodological framework for research and development of such networks. One of the most important things in simulation of ad hoc network is generating a connected graph to represent the network. This paper first presents a literature survey of the most recent methods that concern the generation of network graphs. Then, it introduces two novel and fast algorithms for generating topologies of Ad-hoc networks. The proposed approaches enable the user to generate various network topologies by deciding number of nodes, radio range and minimum distance between any two adjacent nodes in the graph. Finally, the proposed approaches are evaluated and compared with the most recent algorithms. Numerical results demonstrate that the proposed approaches speed up simulation of ad hoc networks and achieve an essential computational cost reduction in comparing with the most recent methods.

Keywords:
Ad hoc networks, Simulation, Graph generation, Network topology

1. Introduction

Recently, the field of Ad hoc networks has seen a rapid expansion of visibility and work due to the rapid increase of inexpensive, widely available wireless devices and the network community's interest in mobile computing. Such networks are used in a wide range of applications. These applications include military and battlefield scenarios where no infrastructure is present, emergency services (search and rescue operations, replacement of fixed infrastructure in case of environmental disasters and policing and firefighting, disaster recovery), commercial and civilian environment (vehicular services: road or accident guidance, transmission of road and weather conditions, taxi cab

* *Corresponding author.*
E-mail address: ibrahim_fayed@outlook.com (Ibrahim S. Fayed)

network and inter-vehicle networks) and educations (virtual classrooms and ad hoc communications during meetings or lectures) [1-4].

Ad hoc network is a group of wireless mobile nodes creating a temporary network without any fixed infrastructure. It does not need centralized administration or fixed infrastructure like base stations or access points. Nodes in ad hoc network that are near enough to each other are assumed to have a direct radio link. The maximum distance into which a node can have a direct radio link to other node is called the node's transmission range. Nodes in ad hoc network cooperate through forwarding packets to each other to make communication outside range of direct wireless transmission. It is typically assumed that all nodes have a similar transmission range.

There are many challenges concerning ad hoc networks such as limited wireless transmission range, broadcast nature of the wireless medium, packet losses due to transmission errors, battery constraints, potentially frequent network partitions and ease of snooping on wireless transmissions [5]. These challenges open the door to the research community to do more research and development to produce algorithms able to overcome such challenges. However, simulation technique is the primary methodological framework for research and development of communication networks. This is because testing algorithms in a real experiment is costly and difficulty. Moreover, running real experiments are always time consuming. Therefore, simulation is very important for ad hoc network research [3, 4].

One of the most important things in simulation is the graph (network topology) generator. That is, in simulation, it is necessary to generate connected graph that simulate ad hoc network to evaluate the performance of different network protocols. A network is said to be connected if the entire network nodes can reach each other either directly or through a multi hop communication nodes that forward a message to the next node on the path to the destination. Typically, as the selected topology often influences the outcome of the simulation, realistic topologies are required to produce realistic simulation results. So, efficient algorithms are needed to generate such graphs with different requirements.

This paper first introduces a literature survey of the most recent methods that concern with the generation of random graphs. Then, it presents two novel and fast algorithms for generating topologies of Ad-hoc networks. The proposed approaches are called Range Based Generation Algorithm (RBGA) and Central Range Based Generation Algorithm (CRBGA). They enable the user to generate various topologies by deciding number of nodes, radio range and minimum distance between any two adjacent nodes in the graph. Finally, the proposed approaches are evaluated and compared with the most recent approaches. Numerical results demonstrate that the proposed approaches speed up simulation of ad hoc networks and achieve an essential computational cost reduction in comparison with the most recent approaches.

The rest of this paper is organized as follows. Section 2 introduces some of related work. Section 3 presents the two proposed algorithms while Section 4 describes the performance evaluation of proposed algorithms. Finally, concluding remarks are listed in Section 5.

2. Related work

There are many studies interested in the creation of wired network graphs [6,7]. Unfortunately, the approaches used for generation of wired network graphs cannot be useful in wireless network. This is because, in wired network, you have the choice to make a link between any two nodes regardless the distance between them, unlike in wireless networks where you are restricted by the transmission radio range of each node.

Other studies are interested in the creation of graphs of wireless networks [9-12]. Existing models for generation of node graph of ad hoc network can be classified into homogeneous and heterogeneous models. The most common approaches in homogeneous models are uniform, chain and grid. In chain model, ad hoc network nodes are generated on a line and the distance between them are equal. In uniform model, n nodes are generated and placed inside an area of size A with uniform distribution. If the area is rectangular with dimensions Xmax and Ymax, node distribution can achieve by sampling x coordinate of a node from 0 to Xmax and y coordinate from 0 to Ymax. In grid model, nodes are generated at intersections of a rectangular grid.

In [10], Camilo et al. used the grid structure for creating a tool for generation of sensor network topology. Generally, these approaches are suitable only for simulation of networks that will have structure like this in the real (like the distribution of tanks in the country borders in Chain model and the distribution of sensors in Square area in grid model for coverage requirement). In [9], Jorgic et al. developed an algorithm where all the N nodes are placed uniformly and independently. Then, they sort all potential edges in the network by their lengths in ascending order to making sure that the graph generated has average degree (the average number of neighbors of each node) equal to d. The length of shortest edge (Nd/2) is chosen as the transmission range R. So, all edges between any two nodes whose distance exceed the new R are eliminated. Then, this graph is checked for connectivity by Dijskstra algorithm [8]. In case of not connected, the process is repeated till a connected graph is generated. In [11], Furuzan et al. had developed two algorithms, called MIN-DPA and MAX-DPA, which create connected topologies with high probability. In MIN-DPA, the main idea is to add new node near to node with minimum degree. In MAX-DPA, the main idea is that when add a new node to the graph, no node has a degree greater than or equal to the maximum degree. So, before a new node is added, all nodes degree in the graph is calculated assuming the new node is placed to this new position. If none of these degrees is bigger than or equal to the maximum degree allowed, the position is accepted. Here, all algorithms allow user to choose the average node degree, but the transmission range is calculated by the algorithms and the user have no control of it. In [12], Shakhov et al. first divided the deployment area into L×L cells. Where, L=A/Lstep, Lstep = R/√5, R is the transmission range, and A is the deployment area. Then, they place one node in every cell (to satisfy the coverage condition). After that, for each cell, they place a random number of nodes to this cell. Here, the main interest of the authors is to provide full coverage of wireless sensor network deployment area and this is unrealistic scenario in ad hoc network where the full coverage is not a point of interest due to its mobility behavior.

The traditional methods presented in [11] and [12] first put nodes randomly and independently, and then check connectivity at the end. If the generated graph is not connected, then it is rejected and repeats the method again. This strategy increases the generation time. Our main interest is to generate efficient network graph faster than the traditional methods. Inspired by the method described in [9] and others described in [11] where nodes are placed one by one, the proposed approaches use this technique but our main interest is to use transition range as independent variable rather than the average number of neighbors that used in [9] and [11].

3. Proposed algorithms

This section presents two novel and fast algorithms, called Range Based Generation Algorithm (RBGA) and Central Range Based Generation Algorithm (CRBGA), for generating topologies of Ad-hoc networks. The proposed approaches enable the user to generate various topologies by deciding number of nodes, radio range and minimum distance between any two adjacent nodes in the graph.

3.1. Range based generation algorithm

The Range Based Generation Algorithm (RBGA) first creates a node to the graph and specifies its position randomly by choosing random value to x coordinate between 0 and L and another random value to y coordinate between 0 and L. Where, L is the dimension of the deployment area (A= L× L). Then, for each new node, the RBGA generates random values to both x and y coordinates and tests the following two conditions for this new position:

- Is the node with this new position become close enough to any other node in the graph to be connected? In other words, the distance between the new node and any other node is less than the transmission radio R.
- Is the node with this new position far enough from all nodes in the graph to represent actuality scenario. For example, when simulating the communication between tanks in battle filed, the size of tank should be taken in consideration so that there is at least 5m between any two nods. In addition, in simulation of sensors, there is minimum distance between any two nodes to avoid the interference. Therefore, the distance between the new node and other nodes in the graph must not be less than the Dmin, where Dmin is the smallest allowed distance between two nodes.

If the new position passes these two conditions, the new node is added to the graph at this position. Otherwise, a new random position is generated until a position passes the above two conditions. Fig. 1 shows the pseudo code of the proposed RBGA.

```
• X=rand(0,l)
• Y =rand(0,l)
• P1=(x, y)
• Add node with p1 to graph
• For k=2,..,N do
•         Accepted = false
•         While (not Accepted) do
•                  X=rand(0,l)
•                  Y =rand(0,l)
•                  Pk=(x,y)
•                  If pk in the range of any node in the graph and not very close to
  all
•                          Add node with pk to graph
•                          Accepted = True
•                  End if
•         End while
• End for
```

Fig. 1. Pseudo code of the first proposed RBGA

3.2. Central range based generation algorithm (CRBGA)

The Central Range Based Generation Algorithm (CRBGA) first creates a node to the graph by choosing random value to x coordinate and another random value to the y coordinate between 0 and L. Where, L is the dimension of the deployment area (A= L× L). Then, for each new node, the algorithm selects a random node from the graph, as a center node, and adds the new node around it. This is done by choosing a random distance (D) between [D_{min}, R] and a random angle (α) from the center node. Where, D_{min} is the smallest allowed distance between any two nodes to represent actuality scenario, R is the radio range and α is an angle between 0 and 2π. Therefore, the coordinates of the new node will be:

X = x coordinate of the center node + D Cos(α) (1)

Y = y coordinate of the center node + D Sin(α) (2)

Finally, the algorithm tests the following two conditions for the new position:
- Is the node with this new position located within the required area? This condition is done because if the randomly selected center node is close to the boundary of the area, there is a probability to the new node to become out of the area.
- Is the node with this new position far enough to all nodes in the graph to represent actuality scenario. This condition is done by check the distance between the new node and all nodes in the graph and ensures that it is not less the (D_{min}).

If the new position passes the above two conditions, the new node is added to the graph at this position. Otherwise, a new random position is generated until pass the above two conditions. Fig. 2 shows the pseudo code of the proposed CRBGA.

```
• X=rand(0,l)
• Y =rand(0,l)
• P1=(x,y)
• Add node with p1 to graph
• For k=2,..,N do
•     Accepted = false
•     While (not Accepted) do
•        D=rand(Dmin,R)
•        α = rand(0,2π)
•        Select random node M from the graph
•        X= M(X) + D Cos(α)
•        y= M(y) + D Sin(α)
•        PK=(x,y)
•        If pK inside the area and not very close to any node
•             Add node with pK to graph
•           Accepted = True
•        End if
•     End while
• End for
```

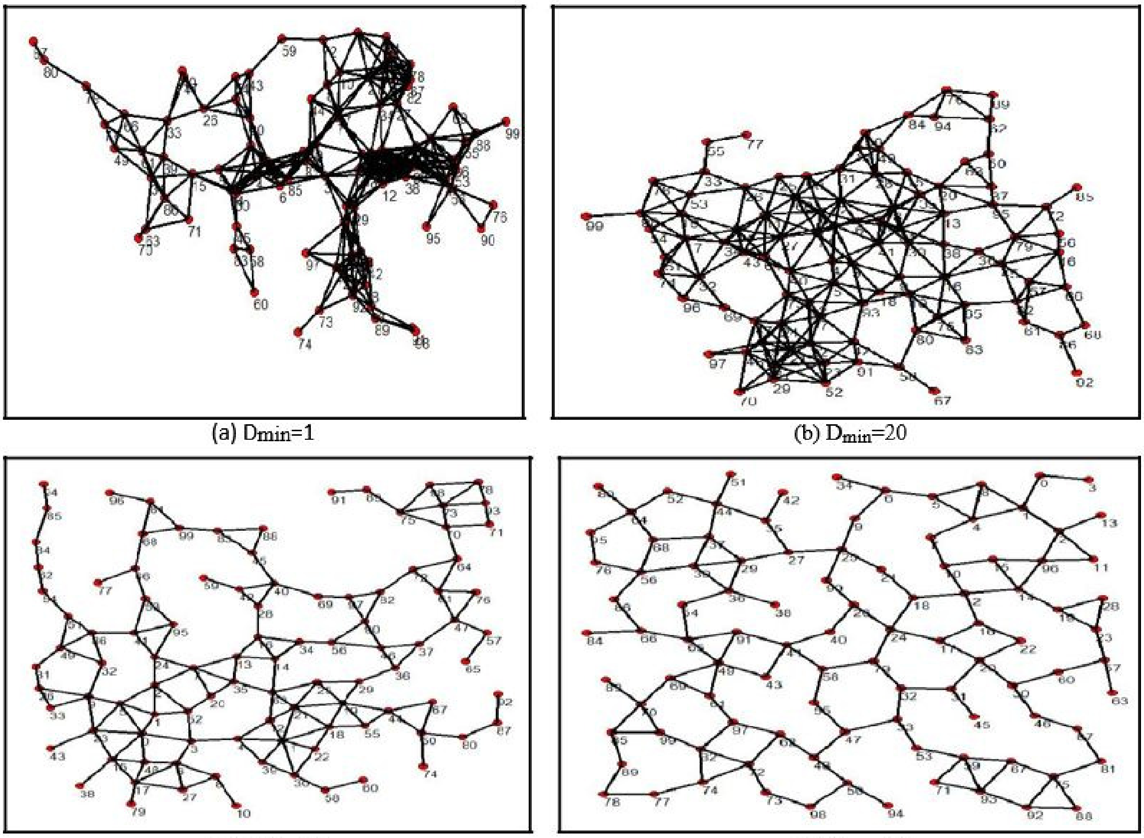
Fig. 2. Pseudo code of the second proposed CRBGA

4. Performance evaluation

To evaluate the proposed algorithms, they are coded in **C#** and used to generate networks topologies. The proposed algorithms allow the user to select the number of nodes, radio range and the minimum allowed distance (D_{min}) between any two nodes of the network.

4.1. Effect of changing D_{min}

Figure 3 presents the effect of changing D_{min} on the ad hoc network topology, when the radio range R is set to 60 and the number of nodes equals to 100. The minimum distance D_{min} is set to 1, 20, 35 and 50.



(a) D_{min}=1 (b) D_{min}=20

(c) D_{min}=35 (d) D_{min}=50

Fig. 3. Effect of changing D_{min} on the generated graph (N=100 and radio range R=60)

4.2. Comparative study

This section presents a comparative study between the proposed algorithms and traditional algorithm [11, 12], by simulation experiment. In this study, the dimensions of the deployment area (A) equal to 600m × 600m. For each experiment, the average value of the results is taken after executing each algorithm 10 times.

Figure 4 shows the average time of each algorithm for generating10 connected graphs when number of node N equals to100 and radio range is set to 75, 80, 85, 90, 95 and 100 m. From Fig. 4, when the range is less than 75m, the time to generate connected graph by the traditional algorithm is too much and the probability to generate connected graph is low. On the other hand, when range less than 65m, the probability to generate connected graph is almost zero and the program go through infinite loop. However, in the proposed algorithms, the average time is proximity constant and does not affect by the changing in the radio range and they. In addition, the proposed algorithms are faster than the traditional algorithm. Where, the RBGA is fourth times faster than traditional algorithm while the CRBGA is ten times faster than traditional algorithm.

Figure 5 shows the average time elapsed by each algorithm to generate 10 connected graphs, when the radio range R=100 and number of node N is set to 100, 300, 500, 700 and 900.

From Fig. 5, the average time increases by increasing number of nodes in the graph by all the algorithms. However, the time required for graph generation by the proposed algorithms is less than that required by the traditional algorithm. In addition, the RBGA is twice times faster than the traditional algorithm while the CRBGA is fourth times faster than traditional algorithm.

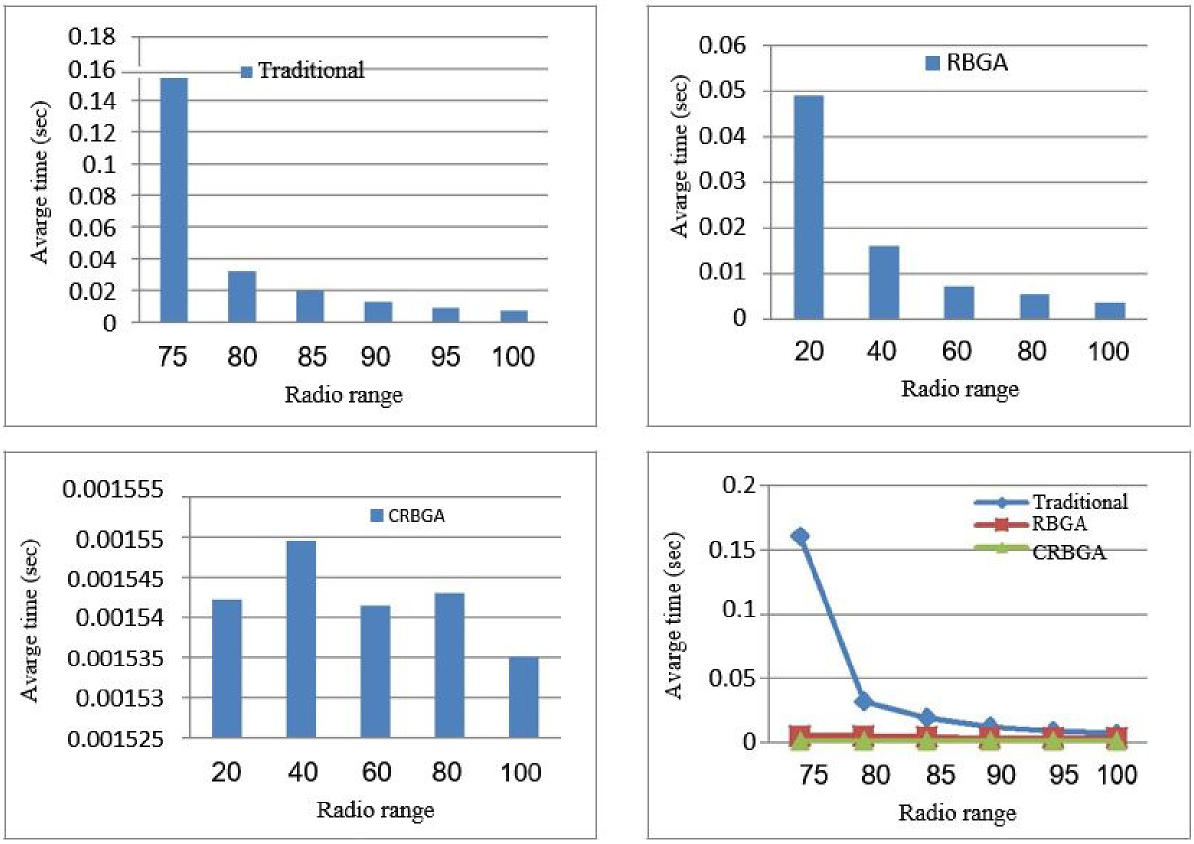


Fig. 4. Average time of each algorithm to generate 10 connected graphs (N=100 and R=75, 80, 85, 90, 95 and 100 m)

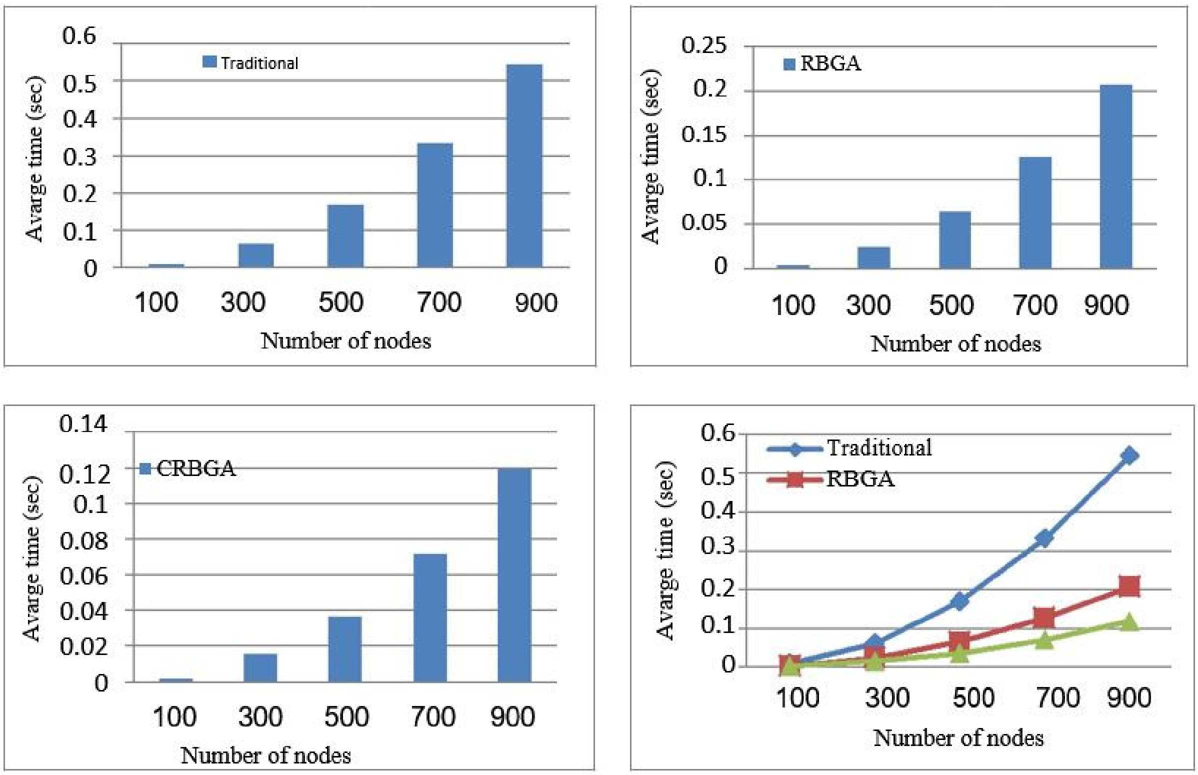


Fig. 5. Average time of each algorithm to generate 10 connected graphs (R=100 and N=100, 300, 500, 700 and 900)

5. Conclusion

In this paper, two new efficient algorithms are developed for generate random graph of ad hoc networks. The main advantage of the proposed approaches is that there is no need to check the connectivity of the graph after its generated so the method is executed only once and never repeated like the traditional method. Also, beside the proposed approaches allow the user to choose the radio range, they also allow the user to choose the minimum distance between any two adjacent nodes in the graph. The experimental results show that performance improvement has been achieved by fast generating of random graphs with prescribed properties. In addition, the numerical results demonstrate that the proposed methods speed up simulation of ad hoc networks and achieves an essential computational cost reduction in comparison with the most recent approach.

References

[1] Rath, Mamata, Binod Kumar Pattanayak, and Bibudhedu Pati. "Energy Efficient MANET Protocol Using Cross Layer Design for Military Applications." *Defence Science Journal* 66, no. 2 (2016): 146-150.

[2] Cambra, Carlos, Sandra Sendra, Jaime Lloret, and Lorena Parra. "Ad hoc network for emergency rescue system based on unmanned aerial vehicles." *Network Protocols and Algorithms* 7, no. 4 (2016): 72-89.

[3] S. Kaur, S. Kaur, and C. Sharma. "An Overview of Mobile Ad Hoc Network: Application, Challenges and Comparison of Routing Protocols." *IOSR-JCE IOSR Journal of Computer Engineering* 11, no. 5 (2013): 7-11.

[4] R. Ramanathan and J. Redi. "A Brief Overview of Ad Hoc Networks: Challenges and Directions." *IEEE Commun. Mag. IEEE Communications Magazine* 40, no. 5 (2002): 20-22.

[5] S. Kumar and K. Dutta. "Securing Mobile Ad Hoc Networks." *International Journal of Handheld Computing Research* 7, no. 1 (2016): 26-76.

[6] K. Calvert, M. Doar, and E. Zegura. "Modeling Internet Topology." *IEEE Commun. Mag. IEEE Communications Magazine* 35, no. 6 (1997): 160-63.

[7] A. Medina, A. Lakhina, I. Matta, and J. Byers. “BRITE: an approach to universal topology generation”, In *Proceedings of the International Workshop on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS)*, 2001.

[8] D. P. Bertsekas, Data Networks, Chapter 5, pp. 401, Prentice Hall, 2nd ed., 1991.

[9] M. Jorgic, I. Stojmenovic, M. Hauspie and D. Simplot-Ryl. "Localized algorithms for detection of critical nodes and links for connectivity in ad hoc networks." *Mediterranean Ad Hoc Networking Workshop*, 2004.

[10] Camilo, Tiago, Jorge Sá Silva, André Rodrigues, and Fernando Boavida. "Gensen: A topology generator for real wireless sensor networks deployment." In *IFIP International Workshop on Software Technolgies for Embedded and Ubiquitous Systems*, pp. 436-445. Springer Berlin Heidelberg, 2007.

[11] Onat, Furuzan Atay, and Ivan Stojmenovic. "Generating random graphs for wireless actuator networks." In *World of Wireless, Mobile and Multimedia Networks, 2007. WoWMoM 2007. IEEE International Symposium on a*, pp. 1-12. IEEE, 2007.

[12] Shakhov, Vladimir, Olga Sokolova, and Anastasia Yurgenson. "A fast method for network topology generating." In *International Workshop on Multiple Access Communications*, pp. 96-101. Springer International Publishing, 2014.