Two-Step Load Balancing Scheme for Fairness Improvement in HetNets

A. D. Usman*, A. M. S. Tekanyi, M. Abdulkarim

Department of Communications Engineering, Faculty of Engineering, A.B.U Zaria, Nigeria.

ABSTRACT: The problem of load imbalance in HetNets among wireless access technologies is addressed in this article. A two-step strategy algorithm was adopted in this work, which considers both load and bandwidth. The first step is a randomized algorithm based on the Monte Carlo scheduling strategy while the second step was a load leveling algorithm that used the brute force method to classify the load on access technologies as average, below average and above average. The obtained results when compared to those of Least Connected Algorithm (LCA) performed better. The results achieved a global load balancing fairness of 0.9119 which was 12.37% better than LCA. The significance of the achieved result translates to better resource utilization among the wireless access technologies and better Quality of Service (QoS) for users.

KEYWORDS: two-step strategy, load balancing, least connected algorithm, quality of service, Monte Carlo algorithm

[Received October 31 2017; Revised January 29 2018; Accepted March 24 2018]

I. Introduction

The concept of wireless access networks emerged in the late 1980s as a by-product of cellular wireless technology. As the demand for cellular service exploded worldwide, the cost of wireless network components decreased, while the cost for deploying and maintaining the conventional copper-based subscriber network increased (Lee and Choi, 2008). During the past few decades wireless technology has seen a tremendous growth. In the upcoming wireless cellular systems such as LTE-Advanced (Long Term Evolution - Advanced) or Beyond 4G systems, wireless access technologies are heterogeneously collocated to meet the growing demands for connectivity and thus inter-cell interference becomes more critical (Nakamura, 2009; Hongseok, 2010). The trend now is collocating different RATs together as depicted in Figure 1.

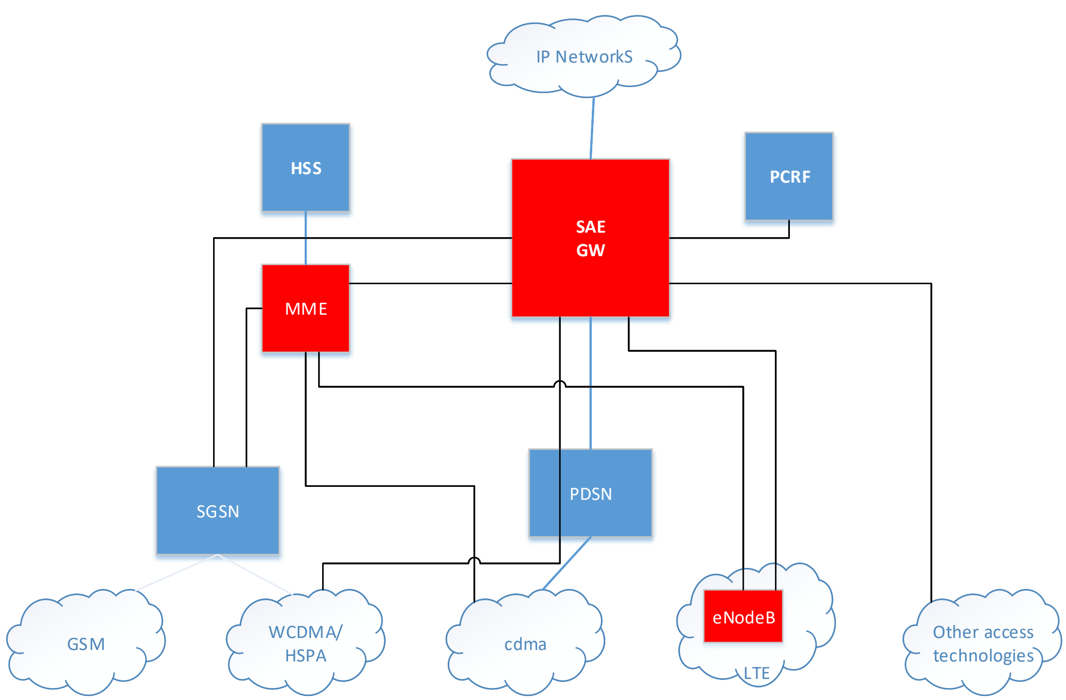

Figure 1: Flat Architecture for the LTE (Ericsson, 2007)

*Corresponding author's e-mail address: aliyuusman1@gmail.com

doi: http://dx.doi.org/10.4314/njtd.v15i2.5

The novelty of multi-RATs implementation is in the fair utilization of the radio resource among the HetNets. In this paper, a heterogeneous network scenario is defined as comprising seven (7) wireless access technologies: EDGE, HSPA, WiMax, HSPA+, WiFi G, WiFi and LTE.

When wireless access networks co-exist to provide services for a number of mobiles, the heterogeneity creates serious issues from how mobiles are seamlessly handed over across the access technologies to how fairly the limited radio resource is utilized. This research solved load imbalance in terms of the radio resource and in the end, measures the degree of fairness of the balancing across the heterogeneous wireless technologies.

II. RELATED WORKS

In a heterogeneous wireless network environment, load balancing could either mean balancing the transmit power or the radio resources which are limited. When considering the later, each Radio Access Technology (RAT) has a maximum capacity and over stretching the resource brings about load imbalance. If some RATs are overloaded while some are underutilized, it will result in poor utilization of radio resources. Balancing of traffic load among multiple RATs in heterogeneous wireless network allows for a better utilization of the radio resources (Falowo and Chan, 2008).

The least connection scheduling algorithm was studied in Donoso *et al.* (2014), Magade and Patankar (2014), Mustapha and Ibrahim (2015), Shengsheng *et al.* (2005). It directs network loads to the network with the least number of established connections. This is one of the dynamic scheduling algorithms that counts live connections for each network dynamically. Least connection scheduling algorithm assigns new request to the network with the least connection (Magade and Patankar, 2014).

In cases where the RATs have the same capacities, the LCA will perform excellently (Ray and De Sakar, 2012), but is not the case with HetNets. However, the system performance is not ideal when the processing capabilities of the networks are different (Mustafa and Ibrahim, 2015; Mahmood and Rashid, 2011). Han *et al.* (2017) proposed weighted LCA is a remedy to LCA but it brings about rounds of computation which in itself is not desirable. The LCA can be represented by the following flowchart (Donoso *e t al.* 2014).

Table 1: Least Connected Algorithm (LCA).

Least Connected Algorithm (LCA)

Require: List of the set of Available Networks $AN_{j,k} = \{t_1, . . . , t_p\}$, $\forall_j \in M$, $\forall_k \in S$ and $p \leq n$

Require: Set of actual connection of mobiles and their services $X = \{x^1_{1,1}, . . . , x^s_{n,m}\}$

Ensure: A (re) allocation of each used service by each connected mobile device.

$\boldsymbol{i_{new}} \leftarrow \boldsymbol{-1}$;

for $1 \leq j \leq \boldsymbol{m}$ **do**

for $1 \leq k \leq \boldsymbol{s}$ **do**

$\boldsymbol{i_{act} \leftarrow i | x^k_{i,j} = 1}$;

$\boldsymbol{i_{new}} \leftarrow$ **Network with lesss number of connections**;

If $\boldsymbol{t_{i_{new}} \in AN_{j,k}}$, **then**

$\boldsymbol{x^k_{i_{act},j}} = 0$, $\boldsymbol{x^k_{i_{new},j} = 1}$;

else

select a random network index from $\boldsymbol{AN_{j,k}}$;

$\boldsymbol{x^k_{i_{act},j}} = 0$, $\boldsymbol{x^k_{i,j} = 1}$;

end if

end for

end for

return A set of connections of mobiles and their services $X = \{\boldsymbol{x^1_{1,1}}, ... \boldsymbol{x^s_{n,m}}\}$

III.MATERIALS AND METHODS

As a result of limited nature of resources, an improper allocation of resources would seriously impair on the performance of the network and consequently give a wrong perception to the users. So, there is a need to define a mathematical model that encodes user requirements and factor in some constraints to make it perform well when deployed in real life scenario.

A. Network Model

Let N, M and S be the sets of n Radio Access Technologies (RATs), m mobiles and s services that compose a Cellular System, respectively. Additionally, let $y_{j,K} \in [0, 1]$ be a binary parameter that indicates that the service k of the mobile j is activated or not. The load of the network $i(\alpha_i)$ is calculated as the sum of demanded bandwidth (D_k) of each connected service (k), for each mobile (j) over the total capacity of the network channel (C_i) as given by (Donoso *et al.* 2014):

$$\alpha_i = \frac{\sum_{j=1}^{m}\sum_{k=1}^{s} D_k \cdot x^k_{i,j} \cdot y_{j,k}}{C_i}, \forall_i \quad (1)$$

$$\text{where } x^k_{i,j} = \begin{cases} 1, & if\ connected \\ 0, & otherwise \end{cases}$$

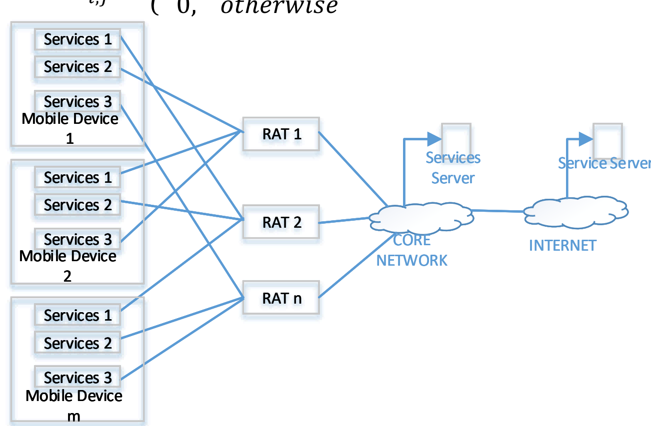

Figure 2: Network diagram for the model (Donoso et al, 2014).

Mobile devices will access the RATs as shown in Figure 2. Since the demanded bandwidth should not go beyond the available bandwidth, there is need for a constraint to ensure that the available bandwidth is not exceeded as follows (Donoso *et al.* 2014):

$$D_k . x_{i,j}^k \leq AB_i \ \forall_i \in N, \forall_i \in M, \forall_i \in S \qquad (2)$$

Where AB_i is the available bandwidth of the network i.

The Received Signal Strength Indication (RSSI) is given as (Shengsheng *et al.* 2005):

$$x_{i,j}^k \leq Z_{i,j}^k \ \forall_i \in N, \forall_i \in M, \forall_i \in S \qquad (3)$$

$$\text{Where } Z_{i,j}^k = \begin{cases} 1, & \frac{RSSI_{i,j}^k}{RSSI_{th}} \geq 1 \\ 0, & otherwise \end{cases} \qquad (4)$$

B. Jain's Fairness Index

Load balancing is aimed at achieving fair distribution of loads among networks. Usually, there would be a measure of fairness at the end of the load balancing. One such measures is the Jain's fairness index widely used for this purpose and is given by Ronok and Mengistie (2012) as:

$$f_x = \frac{\left[\sum_{i=1}^{h} \alpha_i\right]^2}{h . \sum_{i=1}^{h} \alpha_i^2}, \alpha_i \geq 0, \forall_i \qquad (5)$$

where, h is the number of networks constituting the Heterogeneous Wireless Networks (HWN) and α_i is the network's load.

$'f_x'$ is the objective function that the algorithm seeks to maximize.

C. Proposed Algorithm

The algorithm used to address the load balancing problem in this research work is a 2-step algorithm comprising Monte Carlo's Algorithm (MCA) and Load Levelling Algorithm (LLA). The flow charts for MCA and LLA are shown in Figure 3 and Figure 4. The key to Monte Carlo scheduling process is randomness. A final schedule load obtained from MCA is fed as input for LLA as shown in Figure 4.

Figure 4 gives the flowchart for the Load Levelling Algorithm, it completes the second step of the developed algorithm and basically it fine tunes the load balancing to achieve a fair allocation of the load across the heterogeneous wireless technologies. It utilizes the randomized quick sort strategy for sorting an array of elements. Precisely, the brute force method is applied by picking a network among the wireless technologies constituting the heterogeneous wireless environment at random and making it the pivot network otherwise called the 't_{av}'. In other to complete the sorting, a total of (n-1) comparisons are made against the the t_{av}, networks with load below the the t_{av} are labeled t_{min} while those above t_{av} are labeled t_{max}, logically this looks like arranging the remaining (n-1) networks along the two sides of the pivot. The algorithm further searches for any service with highest bandwidth on any of the networks classified as t_{min}, and also searches the corresponding mobile I.d. on any of the networks classified as t_{max} and then exchanges the services.

In the final lap, the algorithm calculates the new network loads for each network and the new t_{av} then repeats the processes until all the mobiles are sorted in terms of their services.

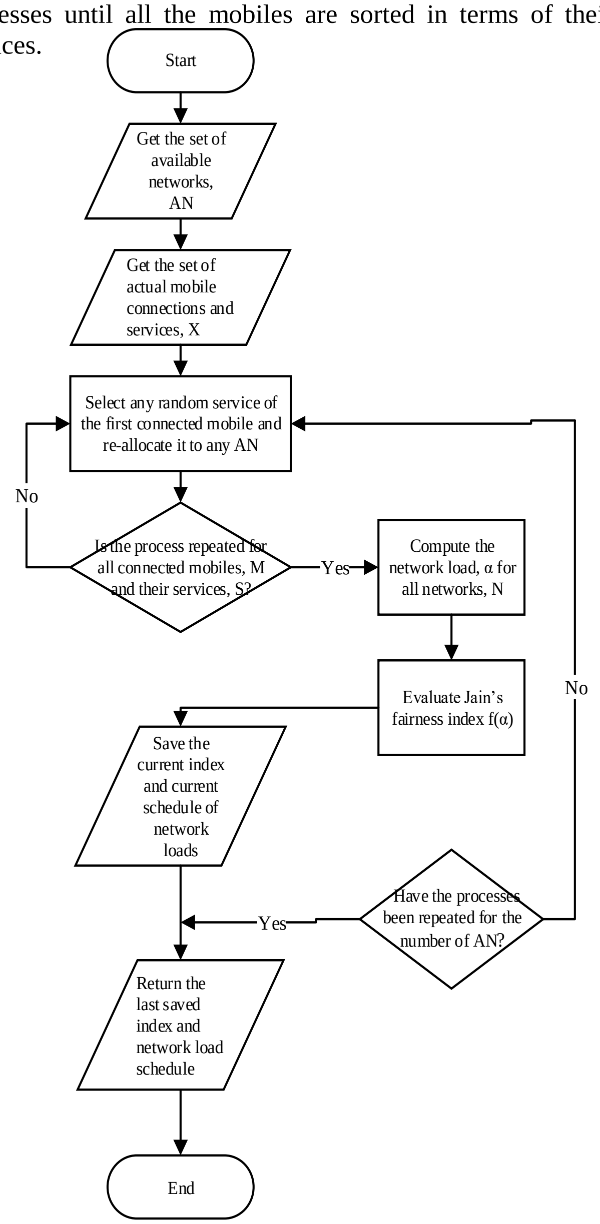

Figure 3: Flowchart for Monte Carlo Algorithm (MCA).

Table 2: Monte Carlo Algorithm (MCA).

Monte Carlo Algorithm (MCA)
Require: List of the set of Available Networks $AN_{j,k} = \{t_1, . . . , t_p\}$, $\forall_j \in M$, $\forall_k \in S$ and $p \leq n$ Require: Set of actual connection of mobiles and their services $X = \{x^1_{1,1}, . . . , x^s_{n,m}\}$ Ensure: A (re) allocation of each used service by each connected mobile device. Select at random required services from $\{x^1_{1,1}, . . . , x^s_{n,m}\}$ Allocate $AN_{j,k} \leftarrow \{x^1_{1,1}, . . . , x^s_{n,m}\}$ Repeat 1 and 2 $\forall_j \in M$, $\forall_k \in S$ Compute α_i, $\forall_i \in N$; Evaluate $f(\alpha_i,)$ Repeat 1 to 5, $\forall_i \in N$, Select $f(\alpha_i,) \leftarrow \mid f(\alpha_i,)_{max}$ Return a set of connections of mobiles and their services $X = \{x^1_{1,1}, . . . , x^s_{n,m}\}$ for selected $f(\alpha_i,)$

Table 3: Load Levelling Algorithm (LLA).

Load Levelling Algorithm (LLA)

Require: List of the set of Available Networks $AN_{j,k} = \{t_1, . . . , t_p\}$, $\forall_j \in M$, $\forall_k \in S$ and $p \leq n$
Require: Set of actual connection of mobiles and their services $X = \{x^1_{1,1}, . . . , x^s_{n,m}\}$ (from Algorithm 1)
Ensure: A (re) allocation of each used service by each connected mobile device.
 	
For $1 \leq j \leq m$ do:
Compute α_i,
Determine the pivot network $t_{av} \leftarrow t \mid t \in AN_{j,k}$
For $1 \leq p \leq n-1$
Classify AN into a set of $t_{min} \leftarrow t_P \mid t_p\{\alpha_t\} < t_{av}\{\alpha_t\}$, and a set of $t_{max} \leftarrow t_p \mid t_p\{\alpha_t\} > t_{av}\{\alpha_t\}$
End for
Select any $x^k_{tmin} \leftarrow x^k \mid x^k \in t_{min}(\max x^k)$ and
Select the corresponding $x^k_{tmax} \leftarrow x^k \mid x^k \in t_{max}(\min x^k)$
Exchange $x = x^k_{tmin.j}$ and $x^k_{tmin,j} = x$
end for
Return a set of connections of mobiles and their services $X = \{x^1_{1,1}, . . . , x^s_{n,m}\}$

III. RESULTS AND DISCUSSION

In this section, resource load balancing results in terms of fairness measured with the Jain's fairness index are presented. The achieved results are compared to those obtained through Least Connected Algorithms (LCA). The behaviour of the proposed algorithm was observed from the sample size of 10 mobiles to 1000 mobiles. Using the pseudo codes presented for MCA and LLA for the load aware load balancing and also coding equations (1) and (5) in MATLAB, the following results were generated and compared with the results of LCA as shown by Table 4.

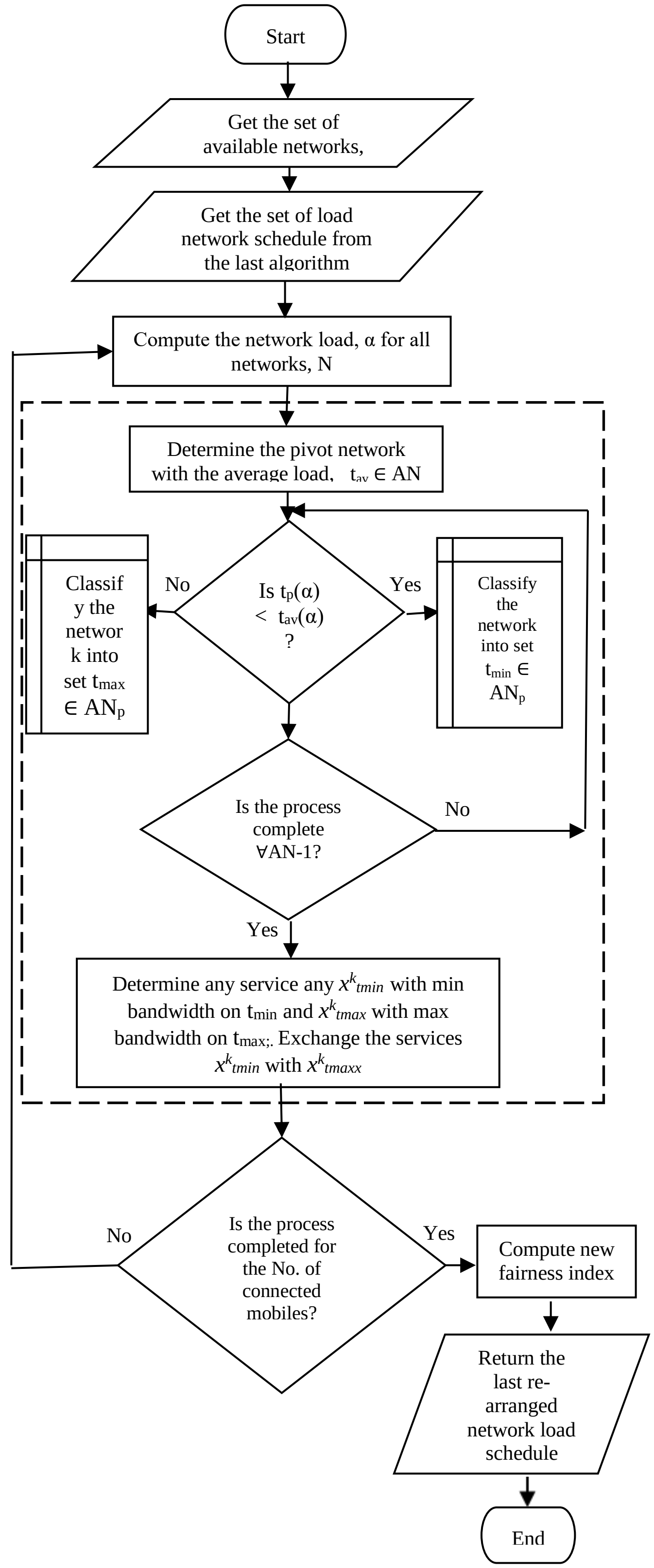

Figure 4: Flowchart for Load Levelling Algorithm (LLA).

Table 4: Number of Mobiles Computed Jain's Index for Proposed Algorithm (PA) and LCA.

Mobiles	LCA	PA
10	0.6754	0.7484
30	0.7319	0.8900
50	0.7779	0.8827
70	0.8310	0.8473
100	0.8301	0.8871
200	0.8077	0.8870
300	0.8181	0.8901
400	0.8305	0.8835
500	0.8254	0.8836
600	0.8079	0.9845
700	0.8159	0.9990
800	0.8229	0.9850
900	0.8304	0.9999
1000	0.8026	0.9994
Sum	0.8115	0.9119

Table 4 shows the fairness indices of varied number of mobiles from 10 to 1000 mobiles. As more mobiles nodes are added into the system, the PA moved closer to the desired fairness index of unity faster than the LCA. For the LCA, it only allocates the next set of mobiles to the access networks having least connections at that particular time. It does not take into consideration the available radio resource on those networks. In cases where the resources are overstretched, fluctuations as would be seen in Figure 5 become evident. On the other hand, the PA which exploits randomness in its allocation yielded better performance. It further uses the brute force method to quickly sort all networks into arrays of less stretched and overstretched. This helps further mitigate problems associated in the LCA.

A graphical representation of Table 4 shown in Figure 5 gives a clearer perspective of the improvement the PA over LCA. As evident from the graph, apart from outperforming LCA, the PA shows a faster convergence towards the desired fairness value of unity than the LCA.

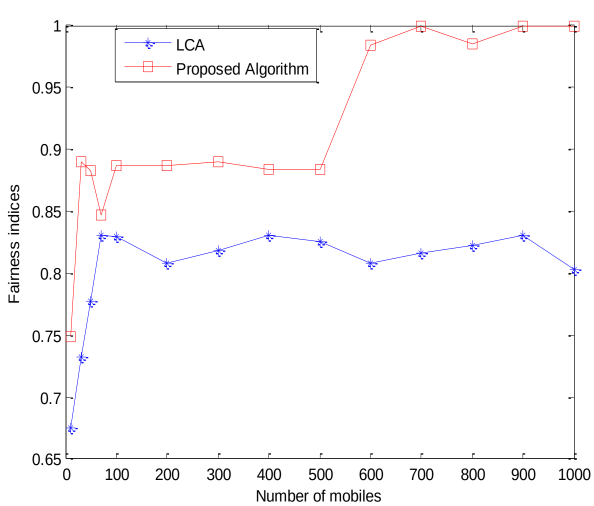

Figure 5: Fairness indices for LCA and PA.

IV. CONCLUSION

The Proposed Algorithm (PA) which is a two-step heuristic strategy made up of Monte Carlo scheduling algorithm and load levelling algorithm was proved to be a viable solution when compared to LCA. The average Jain's Fairness Index for LCA and the proposed algorithm are: 0.8115 and 0.9119, respectively. Hence, PA was found to be better than LCA by 12.37%.

The significance of the achieved result translates to better resource utilization among the wireless access technologies and good Quality of Service (QoS) for mobiles. Call drop rates and poor internet access which are typical of a congested HetNets scenario are minimal due to the improved QoS.

REFERENCES

Nakamura, T. (2009). Proposal for Candidate Radio Interface Technologies for IMT-Advanced Based on LTE Release 10 and Beyond (LTE-advanced). 3rd Workshop on IMT-Advanced. Dresden, Germany.

Donoso, Y.; C. Lozano-Garzon, M. Camelo and P. Vila. (2014). A Fairness Load Balancing Algorithm in HWN using a Multihoming Strategy. International Journal of Computers, Communications and Control, 9(5): 555–569.

Ericsson. (2007). Long Term Evolution (LTE): An Introduction. Available online at: www.telecoms.com/files/2009/03/lte_overview.pdf. Accessed on October 19, 2017.

Falowo, O. E. and Chan, H. A. (2008). Load Balancing in Heterogeneous Wireless Networks Implementing a User-Centric Joint Call Admission Control Algorithm. In Proceedings of Southern African Telecommunications and Applications Conference (SATNAC), Durban, South Africa, 160-165.

Han, G.; L. Liu, J. Jiang, L. Shu and G. Hancke. (2017). Analysis of Energy-Efficient Connected Target Coverage Algorithms for Industrial Wireless Sensor Networks. IEEE Transactions on Industrial Informatics, 13 (1): 135-143.

Hongseok K.; D. V. Gustavo, Y. Xiangying Yang and M. Venkatachalam. (2010). Distributed α-Optimal User Association and Cell Load Balancing in Wireless Networks. IEEE/ACM Transactions on Networking, 20 (1): 177 – 190.

Lee, B. G. and Choi, S. (2008). Broadband Wireless Access and Local Networks: Mobile WiMAX and WiFi. Artech House, Boston, M.A., U.S.A.

Magade, K. A. and Patankar, P. A. (2014). Techniques for Load Balancing in Wireless LANs. Paper presented at International Conference on Advances in Engineering and Technology (ICAET2014), USA, 1–11.

Mahmood, A. and Rashid, I. (2011). Comparison of Load Balancing Algorithms for Clustered Web Servers. Paper presented at the 5th International Conference on IT & Multimedia at UNITEN (ICIMU2011), Malaysia, 1-6.

Mustafa, M. E. and Ibrahim, A. M. (2015). Load Balancing Algorithms Round-Robin (RR), Least-Connection and Least Loaded Efficiency. International Journal of Computer and Information Technology, 4(2), 255-257.

Ray, S. and De Sarkar, A. (2012). Execution Analysis of Load Balancing Algorithms in Cloud Computing Environment. International Journal on Cloud Computing: Services and Architecture, 2 (5): 1-13.

Ronoh, K. and Mengistie, A. (2012). Load balancing in heterogeneous LTE-A networks. Doctoral Dissertation, Department of Science and Technology, Linkoping University, Sweden.

Shengsheng, Y.; Y. Lihui, L. Song and Z. Jingli. (2005). Least-Connection Algorithm based on Variable Weight for Multimedia Transmission. Available online at: http://www.wseas.us/e-library/conferences/skiathos2002/papers/447-144.pdf. Accessed on January 7, 2017.