Load Balancing Technique Based on Network Segmentation and Adaptive Sleep Scheduling for 5G-IoT Networks

Rashad Rashad (rashadphd@gmail.com)

GITAM Institute of Technology: Gandhi Institute of Technology and Management Institute of Technology

Sudhir Sudhir

GITAM Institute of Technology: Gandhi Institute of Technology and Management Institute of Technology

---

Research Article

Keywords: Fifth Generation (5G) network, Internet of Things (IoT), heterogeneous, Load balancing, Sleep scheduling, Segmentation

Posted Date: February 15th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-825633/v1

Load balancing Technique based on Network Segmentation and Adaptive Sleep scheduling for 5G-IoT Networks

[1]T.S.Rashad and [2]A.Ch.Sudhir

[1] Research Scholar, Dept of EECE, GIT, GITAM (Deemed to be University),Visakhapatnam,A.P,India.

[1]rashadphd@gmail.com

[2]Asst. Professor, Dept. Of EECE, GIT,GITAM (Deemed to be University), Visakhapatnam,A.P,India.

[2]camanapu@gitam.edu

Abstract: In Fifth Generation (5G) network, small cells have been incorporated to meet the growing demands for mobile traffic and ubiquitous service. It also enables Internet of Things (IoT) applications that will be positioned all over with mobile broadband technology. As modern IoT applications have active and varied necessities, several systems arrange mixed sorts of network sources consequently. Prevailing studies recommend methods for sustaining the distribution of network segments to diverse applications proficiently.Balancing the loads amongst the cells turns out to be one of the significant problems in cellular network. In this paper, a load balancing technique based on network segmentation and adaptive sleep scheduling for 5G-IoT networks is proposed. In this technique, for each network segment,sub-segments are formed and grouped to process IoT application shaving different Quality of Service (QoS)requirements. The improved DBSCAN algorithm is used for sub-segmentation and grouping. based on a combined rank of QoS factors. In the next phase, each small cell base station (SBS) executes adaptive dynamic sleep scheduling based on its load level. In the load balancing policy of SBS, when the average load of any SBS increases beyond the load of the Macrocell, the overloaded traffic can be moved to the Macrocell. Simulation results are validated against the analytical results and hence prove that the proposed LBNSASS technique achieves higher success probability and power efficiency with reduced energy consumption and packet drops of SBSs when compared to existing technique.

Keywords: Fifth Generation (5G) network; Internet of Things (IoT); heterogeneous; Load balancing; Sleep scheduling; Segmentation

1. Introduction

IoT is a universal architecture that envisages the connectivity of enormous devices and information matters to attain the vision of smart cities. This amazing growth in the number of associated users will make an enormous sum of IoT network traffic. Furthermore, the produced traffic may change frequently [1]. IoT devices are habitually battery-restrained, and it is thus not energy competent to permit an IoT device to direct its IoT content to a huge number of clients, who attempt to recover the content of the IoT device. [2].

IoT deployment is certainly anticipated to depend on the mixture of numerous diverse technologies that will comprise the 5G and beyond 5G landscape. Such systems offer huge attention and interference-restricted access and can therefore help mMTC services toward the so called cellular IoT (cIoT) [3].

Cell loads can be well-accustomed if a User Equipment (UE) recognizes the loads of neighbouring cells and selects the least loaded cell. Though, users can be connected to more than one Base Station (BS) to lessen the load, positioning dense small cell networks creates user association more perplexing.Load balancing is an effectualtechnique for balancing the traffic and easing the traffic amongstvaried networks in the forthcoming 5G networks [5].Prevailing works on load balancing require precise hardware deployments.

In our preceding article [6], we have intended handoff prediction and target network selection scheme for 5G-IoT networks. For VHO triggering circumstance, Multi-layer Feed Forward Network (MFNN) is used which will calculate the user mobility based on distance, RSS, mobile speed and direction factors. For target cell selection, Fuzzy decision model is used based on the network level metrics like traffic load, handover latency, battery power and user level metrics like security and cost.

1.1 Problem Statement and Objectives

Modern IoT applications have active and variednecessities. Therefore several systems highlight mixed kinds of network resources consequently. Prevailing studies recommend measures for upholding the distribution of network segments to various applications effectively [7].

Depending on the problem statement, the principal goal of our investigation work is to plan a load balancing method to content the varied 5G-IoT applications with various QoS necessities. Therefore as an extension to this work, we suggest to strategy a load balancing method based on network segmentation and adaptive sleep scheduling for 5G-IoT networks.

2. Related Works

Mostafa Mouawad et al [1] have suggested a resolution for enhancing the QoS depending on predicted load and developing the network remapping ability of CRANs. The network load is anticipated by applying Markov Model based on present user's location in each cell. The poised traffic load causes the lessenning of congestion in the IoT network and therefore progresses the QoS.

Xiang Sun et al [2] have suggested to reallocate the resources from profoundly loaded devices into frivolously loaded devices so as to balance the loads amongst devices. They expressed the common resource reallocation problem as an optimization problem, which is verified to be NP-hard. They intended the Latency aware populAr Resource recachiNg (LEARN) algorithm to effectively resolve the issue, and validate the act of algorithm through simulations.

Sushil Kumar Singh et al [7] have suggested a ML based network sub-slicing architecture in 5G networks to address load balancing issues. In this work, every logical slice is split into a virtual sub-slice of sources. Every sub-slice offers the application with various ordered resources as needed. One sub-slice emphases on spectral efficiencywhile the other emphases on offering low latency with condensed power depletion.

KaigeQu et al [8] have suggested a traffic engineering (TE) framework for effectual resource management amongst slices, to evade congestion and avert the resultant QoS reduction. NFV architecture combined with hierarchical SDN controllers positioned in tenant and infrastructure domains can help the suggested TE framework. A sample study is offered to assess the efficiency of

the suggested TE framework, regarding QoS performance assurance, enhanced resource utilization, and condensed reconfiguration overhead.

Zhu Xiao et al [9] have well-defined the operating possibility for the SBS using the suggested dynamic sleep mode connected to its measured load level. The expressions of success probability for coverage that is utilised to choose whether adynamic user can attach to a SBS fruitfully are obtained for the suggested sleep modes. Energy reductions are offered for the suggested sleep modes subject to the success probability restraint.

Misikir et al [10] have used the theory of MDPs to improve a near-best state based policy, both for a biased sum of the act and energy, where energy is lessened conditional on a restraint on the act. Precisely, they used the initial phase of the renowned policy iteration technique for every arrival needs assessing the marginal future cost of summing the arrival in the small cell or the Macrocell.

3. Proposed Solution

3.1 Overview

This paper designs a load balancing technique based on network segmentation and adaptive sleep scheduling for 5G-IoT networks. The block diagram of the proposed technique is shown in Figure 1.

In this technique, separate network segments are formed to process heterogeneous IoT applications with various QoS necessities. Initially, the set of IoT devices are grouped proficiently based on their general necessities or services. Group ids are assigned depending on same sorts of application facilities. The main QoS factors considered for segmentation are latency, power efficiency, average traffic load. A combined rank is then estimated for segment based on these QoS factors. In the next phase, dynamic sleep mode is determined for each small cell base station (SBS) based on the combined rank of each segment.In the load balancing policy of SBS, when the average load of any SBS increases beyond the load of the Macrocell, the overloaded traffic can be moved to the Macrocell.

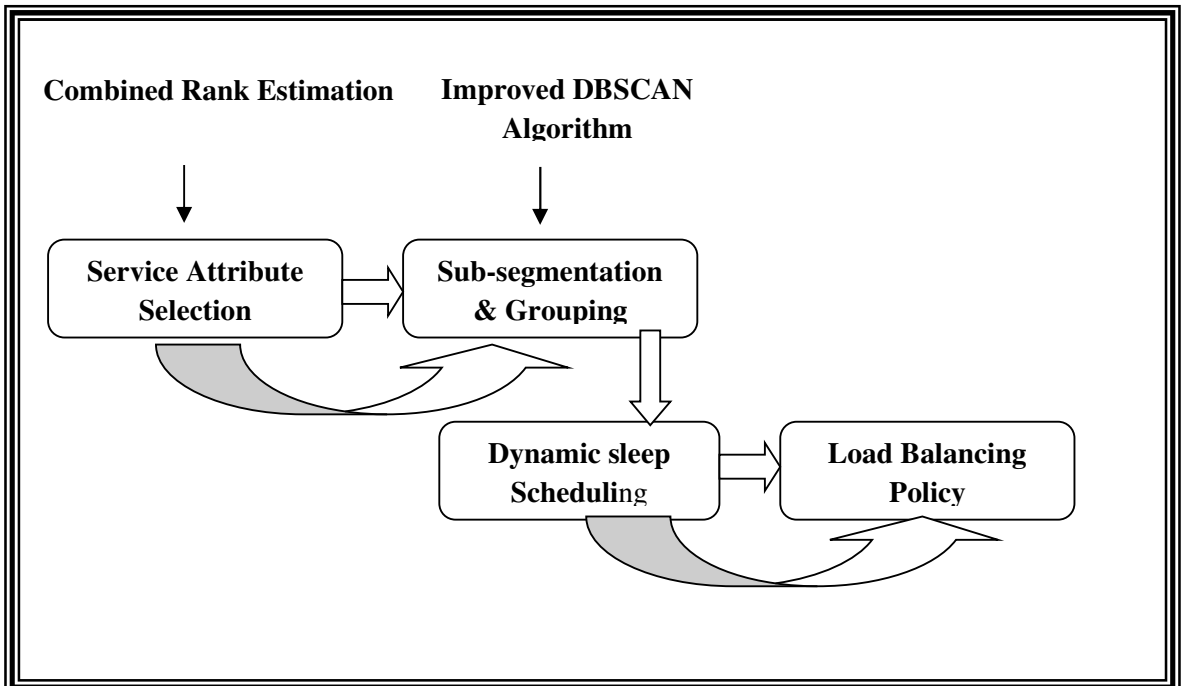

Figure 1 Block Diagram of Load balancing technique

3.2 System Model

The system model consists of 1 Macrocell with BS (MBS) and N small cells with base station (SBS) situated in the coverage region of Macrocell. It is assumed that the transmission of each SBS does not interfere with each other.

IoT communication gadgets are utilised in IoT layers with network segmentation. It consists of various IoT devices linked with sensors that sense and direct data facilities in several forms. IoT data will be calibrated and observed in present with specific IoT amenities.

The traffic model is a mixture of real-time multimedia and text files. Let A_S denote the arrival rate of a flow served by SBS and A_M denote the arrival rate of flows served by MBS.

3.3 Networking Segmentation

Network segmentation is an indispensable technique for using network services in the supportable 5G atmosphere.

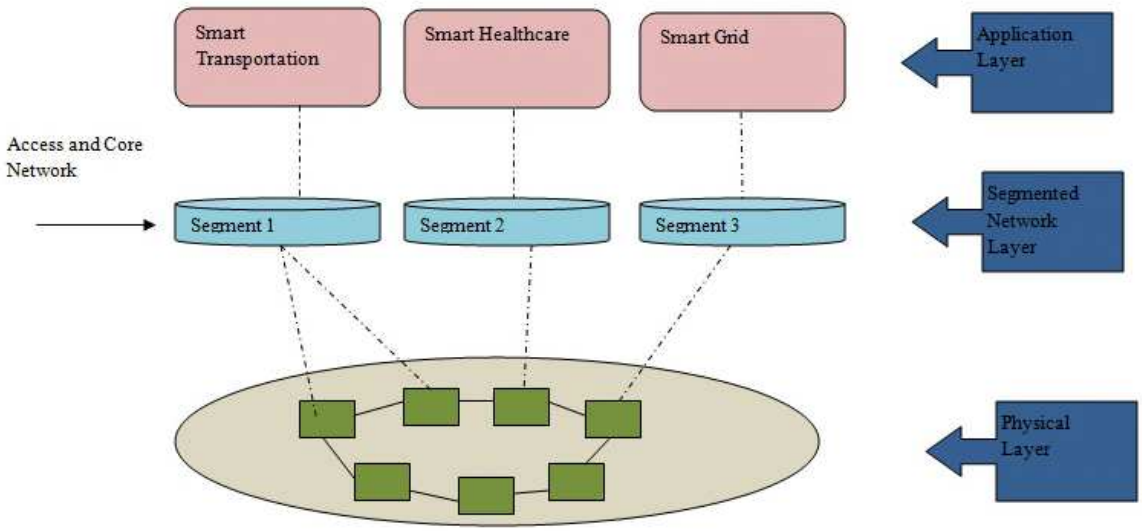

Figure2 Network segmentation architecture

As running many logical networks has nearly non-dependent IoT processes on a normal physical environment, it has the benefit of virtual architectures where numerous applications are utilised. Virtualized sources let the supple, effectualuse of restricted resources and offerutmost services, basically mobile and IoT, amongst others.

The existing network segmentation architecture as exposed in Figure 2. It permits a node to offer facilities to individual users depending on their necessities. These users offer separate facilities as per their settings. Network segmentation is achieved using Network Functions Virtualization (NFV) [7].

Every single network segment is utilised for specific IoT application facilities.The physical network has several IoT devices with diverse detached aspects.

However, network segmentation is not enough to offerthe entire kinds of amenities to the IoT application. Hence, network sub-segmentation notions are utilized giving the entire sorts of facilities for all IoT-5G heterogeneous applications.

3.3.1 Network Sub Segmentation (NSS)

The network Sub-segmentation(NSS) techniquesplits the parent network segment to provide load balancing in 5G network segmentation.

This section presents the steps involved in the proposed NSS framework in 5G networks. It contains the subsequent three distinct elements:

(1) Service attributes selection, which describes how to choose the qualities depending on the facilities of intelligent applications in the IoT-5G setting. IoT devices are ranked according to the some QoS factors created in attribute selection process of the devices.The attributes with the lowest ranking are eliminated. Lastly, we get the declining order of the entire service qualities of IoT devices for NSS.

(2) Grouping and sub-segmentation are inter-related at the IoT layer. The DBSCAN algorithm is used for grouping and sub-segmentation.Each sub-segment is applied to specific sub services in many IoT applications. It creates sub-segments using the DBSCAN algorithm and allocates the cluster based on the combined rank of the IoT devices.

3.3.2 Combined Rank Estimation

The IoT devices in each cell are allotted a combined rank based on the average cell load, average battery power and transmission latency.

The cell load is represented in terms of the data rate R.

The load for a traffic request l in a collection C_i can be obtained by

$$H_i^l = R(l)$$

At time instant t, the total load for all the traffic requests in C_i can be subsequently

$$H_i(t) = \sum_{l=1}^{L} H_i^l \qquad (1)$$

The averageresidual battery power of each node (RPW) following a data communication is estimated using the following Eqn.

$$RPW = [BP_i - (BP_{tx} + BP_{rx})] \quad (2)$$

Where BP_i denotes the initial battery power,

BP_{tx} denotes the battery power consumed during transmission

BP_{rx} denotes the battery power consumed during reception

Latency (L) is the least time for transferring the data from a given node to the distant node.

The combined rank CR for a user U_j of the cell C_i is given by

CR^i_j =(w1. H) (w2. L)/ (w3. RPW) (3)

3.3.3Attribute Selection and Grouping

The IoT devices attributes such as device id, cell id, BS id, the load for traffic request R(l), PW and latency L are collected.

Let S be the subset of attributesS= [1, 2,…….n] and $B = \phi$ be the rank list.

The algorithm for attributes selection and ranking is given below:

Algorithm-1:AttributesSelection and Ranking

1: Attribute Sorting repeat until $S = \phi$

2:Restrict to good attribute indices: $A_0 = A(:, s)$

3.Compute the load factor, latency and battery power.

4Compute the weight values w1, w2 and w3

5Calculate the combined rank using Eq.(3)

6 Find the attributes with the least IoT device rank

$$F = \arg_k \min G_k$$

7 Terminate this aspect with the least ranking principles B in order that $B = s(1: f-1, f+1, length(s))$

3.3.4 Improved DBSCAN Algorithm for NSS

(i) Basic Algorithm

The DensityBased Spatial Clustering (DBSCAN)algorithm [11] is designedto discover the clusters and the noise in a spatialdatabase D.

It realises groups of diverse shapes and sizes from a huge amount of data comprising noise and outliers.In DBSCAN clustering, it is not essential to stipulate the number of groups to utilise it. It is sufficient to compute the distance amid values. DBSCAN also creates more sensible outcomes than k-means clustering cross a range of diverse dissemination.

The DBSCAN algorithm utilises two factors:

- The radius distance X of the adjacent around a point r
- The least number of adjacent points Y_{Min}within X.

By conducting different set of experiments, the optimal parameters for X and Y_{Min} are estimated to be 5 and 7, respectively.

Let DS be the data set.

The basic DBSCAN algorithm is given below:

Algorithm-2: DBSCAN Algorithm

---

1. For each unvisited point r_i in DS,
2. Identify the neighbourhood of x_i that includes at least Y_{Min}points.
3. Mark $r_i$as visited.
4. End For
5. For r_i, which are not assigned to a specific cluster,
6. Generate a novel cluster M.
7. Include the points in the adjacent of x_i to a candidate set CS.
8. Include any points in CS to M
9. For each point b in CS,
10. Discover the adjacent of b that comprises at least Y_{Min}points.
11. Include these points in the CS
12. Assigned these points to cluster M.
13. Label b as visited.
14. End For
15. End For
16. If some points do not belong to any cluster, then
17. Label them as outliers
18. End if

.

(ii) Improved DBSCAN Algorithm

The traditional DBSCAN algorithm has many shortcomings with respect to the efficiency of execution. Moreover, if the seed objects are less, the procedure may miss the object in the cluster growing procedure.

To overcome these shortcomings of existing DBSCAN algorithm, an improved version of DBSCAN algorithm is proposed [12] for Heterogeneous data. The main idea of the algorithm consists of the following phases: (i) graded processing of data sets with irregular density distribution; (ii) rapid cluster development processing.

Let p be the sample point.

Let P be an unprocessed object

PDF be the probability density function

The improved DBSCAN algorithm is shown in Algorithm 3.

Algorithm-3: Improved DBSCAN Algorithm

Input:

D - dataset, m-order of the Gaussian mixture model, Y_{Min} - threshold factor

Output:

Group obtained by clustering

1.Define D, initialize the object and grade p as unspecified.

2. Using the PDF of each p, the datasets of different density levels $\{C_1, C_2, \ldots\ldots C_n\}$ are obtained.

3. Y_{Min} is set to compute the value of

$X_{C_i} (1 \le i \le n)$ parameters

4. Local grouping of the data set C_i is utilised by the basic DBSCAN procedure.

5.Select P dynamically in C_i to discover the aggregate number of objects T whose X and Y_{Min} density can reach,

6. If $T >= Y_{Min}$

pis consideredas core point.

All the reachable objects in X(p) are grouped as a cluster.

7. If $T < Y_{Min}$,

P is marked as a noise

8. Repeat from Step 6, until all points C_i are processed

// Cluster Expansion Process

9.Choose *k* reference objects in the group to join {Seeds}.

10. If s_i€ {Seeds} is not the core point,

Remove $s_i$from the {Seeds}.

Else

Add neighbours of s_i into Nes_i

11. Select k' representative objects in Ne$s_i$to join {Seeds*}

12. For eachs'€ {Seeds*}'

13. If s' isuncategorized

Add s' to {Seeds}.

Remove s_i from the {Seeds}.

14. Repeat from step 9 until all the s_i€ {Seeds} are processed.

16.While (all the data sets C_i are clustered)

3.4 Determining Dynamic Sleep Mode for SBS

3.4.1 Dynamic Sleep mode

For every small cell, we use the dynamic sleep state control model. If there are pending flows in the small cells, the state of the SBS is said to be ACTIVE. When all the flows have been served, the state becomes IDLE. The SBS remains IDLE until a new flow arrives or the IDLE timer (T_{IDLE}) expires.In the later case, the state of SBS becomes SLEEP.

In this model, the length I_k of the IDLE timer of small cell k is assumed to be exponentially distributed with mean

$$E[I_k] = 1/\omega_k \qquad (4)$$

The SBS will be in SLEEP state for entire sleep duty cycle T_{SLEEP} which is determined based on the load of the cell. After expiration of the duty cycle D, the SBS changes to AWAKE state and again changes to ACTIVE state when a new flow is arrived.

The state of SBS_k at time t is denoted by the pair $(F_k(t), ST_k(t))$, where $F_k(t)$ denotes the total flows and $ST_k(t)$ to refers to the state of SBS.

The state changes of SBS are summarized as follows:

STATE	CONDITION
ACTIVE	$F_k(t) > 0$
IDLE	$F_k(t) = 0$
SLEEP	$F_k(t) = 0$ and T_{IDLE} expires
AWAKE	T_{SLEEP} expires

3.4.2 Determination of Sleep Duty Cycle

Rather than arbitrarily switching off SBSs, every SBS implements load-awareness dynamic sleep based on its load level.

For a SBS_k with load level L_k, the active probability is $A(L_k)$ and sleep probability is $(1-A(L_k))$

The active probability $A(L_k)$ can be defined as a function which keeps increasing until the load level L_i reaches the threshold L_{th}, where L_{th} is the predefined threshold for load level.

Hence, the Active probability $A(L_k)$ can be defined as follows:

$A(L_k)$	CONDITION
0	$L_k = 0$
$e^{(L_k - L_{th})}$	$L_k < L_{th}$
1	$L_k >= L_{th}$

Hence the sleep duty cycle T_{SLEEP} is given by

$$T_{SLEEP} = 1 - A(L_k) \qquad (5)$$

It is presumed that the load level threshold L_{th} is identical for the whole SBSs. We presume that the SBS can spontaneously calculate its load level regarding traffic demand in precise time period.

3.5 Load Balancing Policy

Consider now a load balancing policy P_{LB}determined by the vector $v = (v_1,.....v_n)$, where each term v_n gives the probability of assigning the incoming request to the small cell C_n and $(1-v_n)$gives the probability toMacrocell C_M.

The small cells, $C_1,.....C_N$, are sorted in the descending order based on their cell loads, i.e $H_1 > H_2.....> H_N$

For each of the C_i, where the previous load is less than the load of the Macrocell H_M, no traffic can be shifted to the Macrocell, i.e., the resultant $P_{LB} = 1$

Hence, let K^* signify the maximum index of C_i from which load can be shifted to the Macrocell, i.e.,

$$k^* = \{\max k = 1,....., N : H_k > H_M\} \quad (6)$$

Algorithm:-4 Load Balancing Policy

1.PLB = $\{V_1,.....V_n\}$

2. Sort $\{C_k, k=1,..N\}$such that $H_1 > H_2 ..... > H_N$

3. For each C_k

4.If $H_k < H_M$, then

5. $P_{LB}(k) = V_k = 1$

6. Else if If$H_k >= H_M$,then

7. k* = k

8. break

9. End if

8. End For

9. For each C_j, j=k*, ...N (k* > k)

10.$P_{LB}(j)$ = Move $H_j$to C_M

11. End For

12. Stop

3.6 Mathematical Model

The aggregate power used by SBS in active state(P_{act}) can be expressed as

$$P_{act} = P_T + P_R + P_{PA} \quad (7)$$

Where P_T, P_R and P_{PA}represents the power consumption of RF transmitter, RF receiver and RF power amplifier, respectively.

Similarly, the aggregate power consumption of SBS in the idle state (P_{idle}) is given by

$$P_{idle} = P_{BC} + P_0 \quad (8)$$

Where P_0 signifies the static power used by the SBS in IDLE state and P_{BC} signify the power consumed at the backhaul

The success probability can be defined as

$$q_s = E[P(SINR > T)]$$

The aggregate power usage P_{DS} for the SBS after using the suggested load-aware dynamic sleep scheduling is computed as

$$P_{DS} = \sum_{i=0}^{\frac{\lambda_s}{\lambda_m}} \{[A(L_k)(P_{act} + \beta P_T) + [1 - A(L_k)].P_{idle}]t\} \quad (9)$$

$$q_s^{DS} \geq \varphi$$
$$s.t\ s(x_i) \in [0,1]$$
$$x_i \geq 0 \quad (10)$$

Power efficiency can be obtained by the ratio of power consumption and success probability

$$PE = P_{DS} / q_s \quad (11)$$

4.Experimental Results

4.1 Simulation Parameters

The proposed Load balancing based on Network Segmentation and Adaptive Sleep scheduling (LBNSASS) technique is simulated in NS2 and compared with Machine Learning based Network Subslicing Framework (MLBNSF)[7].The performance metrics energy consumption, success probability, computational overhead, packet drop and power efficiency are analyzed in the experiments.

Our simulation settings and parameters are summarized in table 1

Number of Macro cells	6
Number ofsmall cells	12
Area size	500 X 500m
Bandwidth	100Mbps
Cell Range	Macro cells- 5 km small cells – 25m

Number of users	6 users per cell
Load	10 to 20Mb
IoT Traffic models	Constant Bit Rate (Non real-time), Exponential (real-time) and TCP (Text)
Transmit power	0.4 KW
Receive power	0.6 KW
Idle power	0.03 KW
Initial Energy	20 Joules

Table 1: Simulation parameters

4.2 Results

In simulation experiment,the load of each SBS is varied from 10 to 20 Mb

4.2.1 Comparison with MLBNSF

This section presents the simulation results of LBNSASS and MLBNSF techniques.

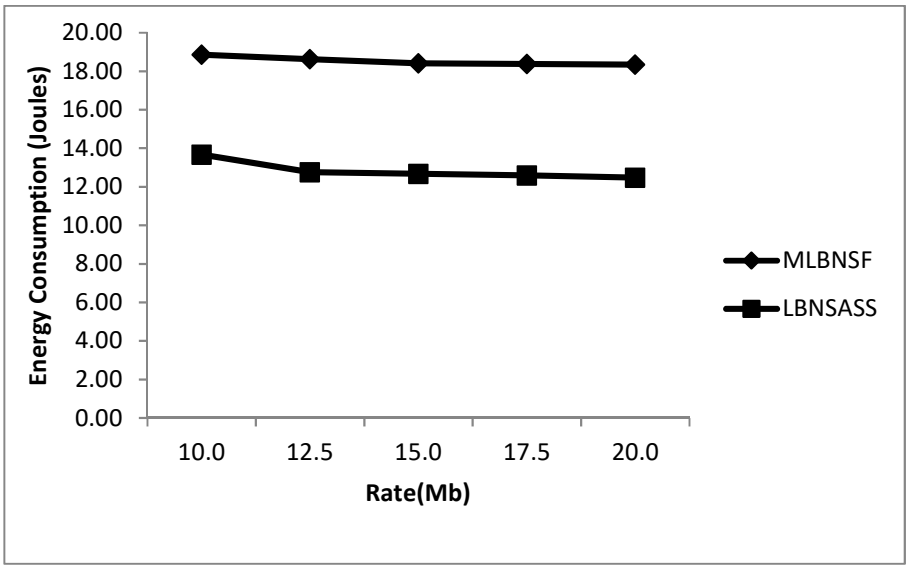

Figure 3 Energy consumption for varying Rate

The average energy consumption for varying the traffic rate is shown in Figure 3. The energy consumption slightly decreases from 13.6 to 12.4 joules for LBNSASS and decreases from 18.8 to 18.3 joules for MLBNSF. Since LBNSASS provides adaptive sleep scheduling technique for small cells, it has 30% lesser energy consumption than the MLBNSF protocol.

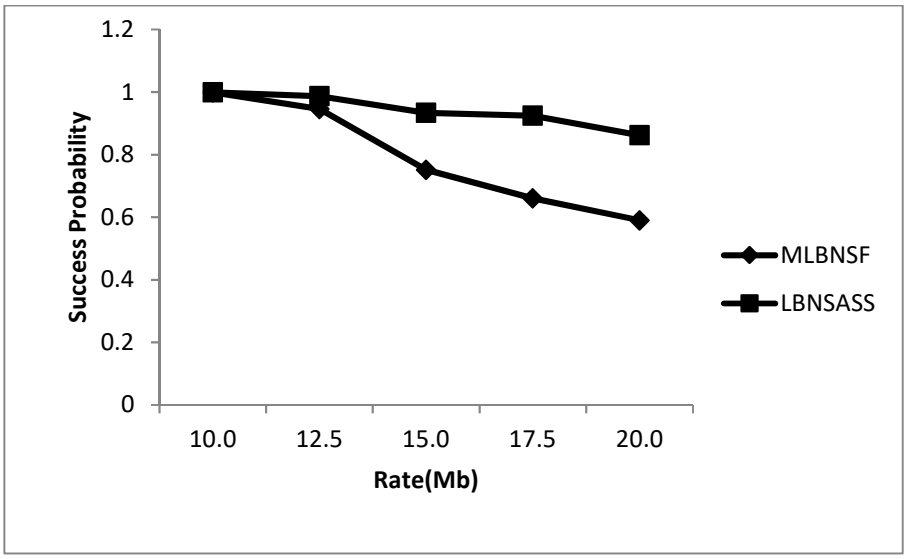

Figure 4: Packet delivery ratio for varying Rate

The success probability of data delivery for various rates, is shown in Figure 4. It reduces from 0.99 to 0.86 in case of LBNSASS and reduces from 0.99 to 0.59 in case of MLBNSF. Since LBNSASS consider combined rank of devices based on the load, it has 17% higher success probability when compared to MLBNSF.

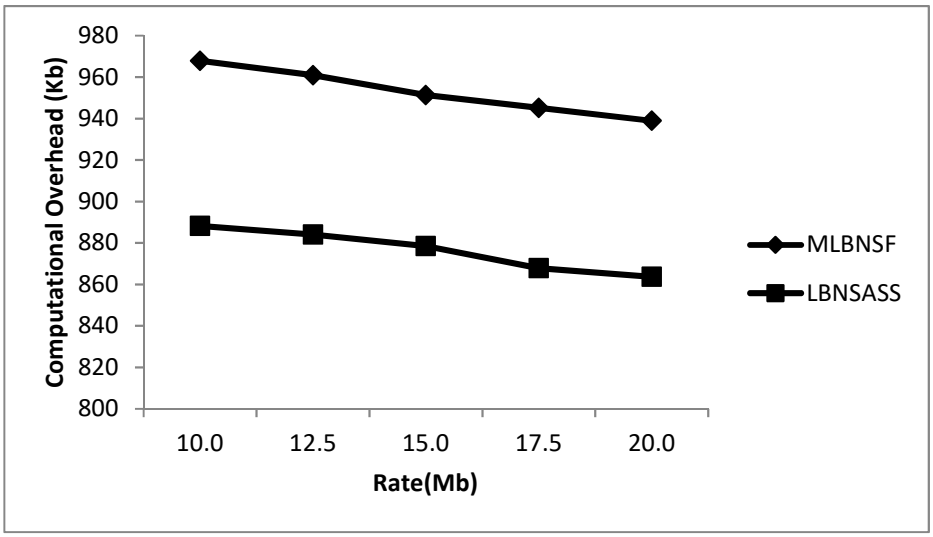

Figure 5 Computational Overhead for varying Rate

The computational overhead of both the schemes for various data rates is shown in Figure 5. The overhead of LBNSASS varies in the range of 888 to 863 Kb whereas the overhead of MLBNSF varies in the range of 967 to 938 Kb. Due to the DBSCAN algorithm, LBNSASS has 8% lesser overhead than MLBNSF.

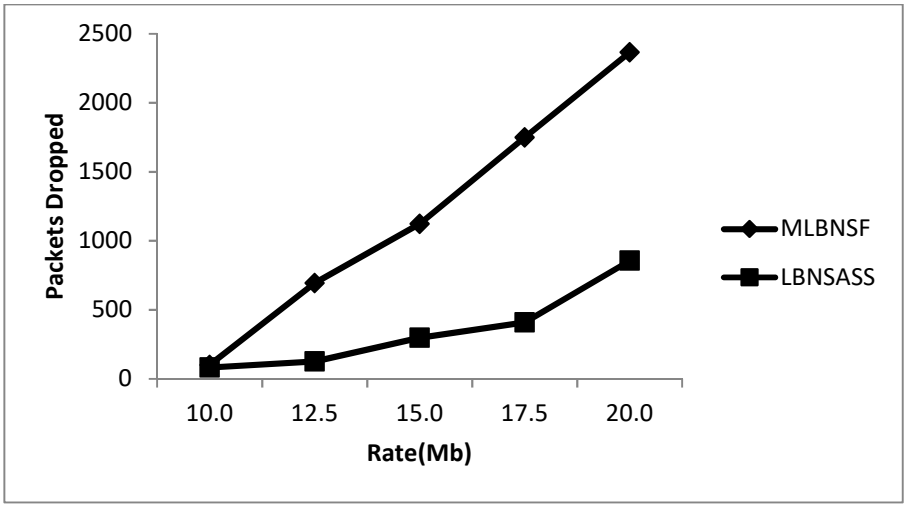

Figure6 Packet Drop for Rate

The average packet drop of both the scheme for various data rates is shown in Figure 5. The packet drop of LBNSASS falls in the range of 83 to 857 packets whereas the packet drop of MLBNSF falls in the range of 101 to 2366 packets. Ultimately, the LBNSASS has 64% lesser packet drops than MLBNSF.

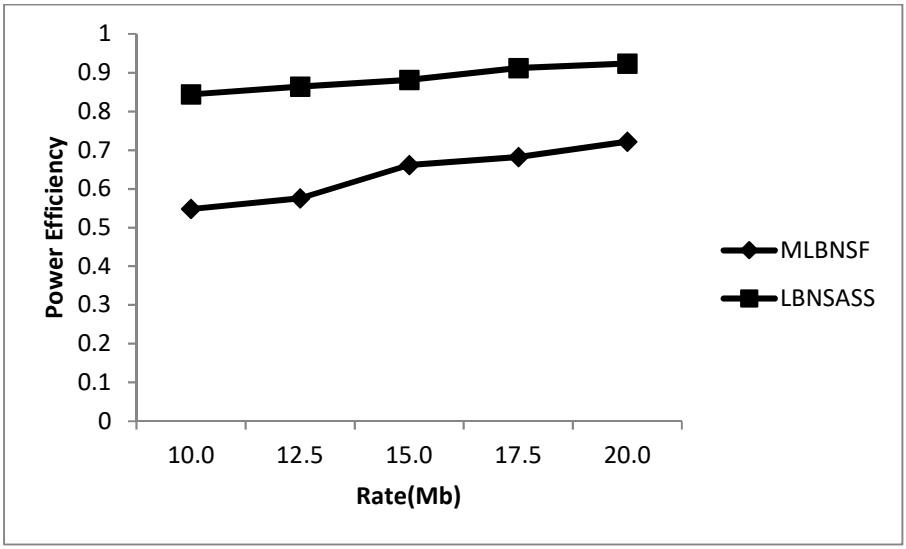

Figure 7 Power Efficiency for varying Rate

The average power efficiency for varying the traffic rate is shown in Figure 7. The efficiency slightly increases from 11.8 to 21.1 for LBNSASS and increases from 11.5 to 15.6 for MLBNSF. Since LBNSASS provides adaptive sleep scheduling technique for small cells, it has 28% higher efficiency than the MLBNSF scheme.

4.3 Comparison with Analytical Results

This section presents the comparison of simulation and analytical results of LBNASS technique.

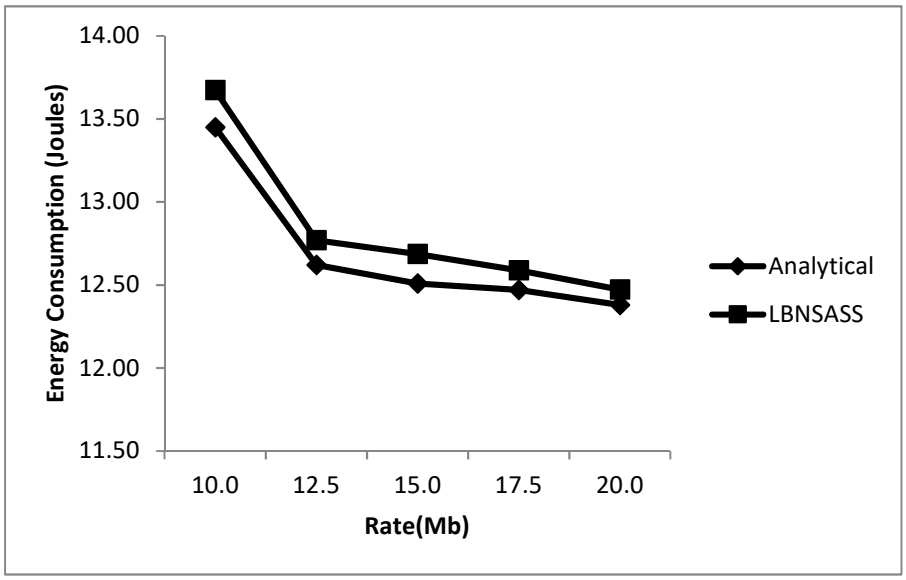

Figure 8 Energy Consumption for varying Rate

Figure 8 shows the comparison of energy consumption values for simulation and analytical results. It can be observed that difference between both types is nearly 1% only.

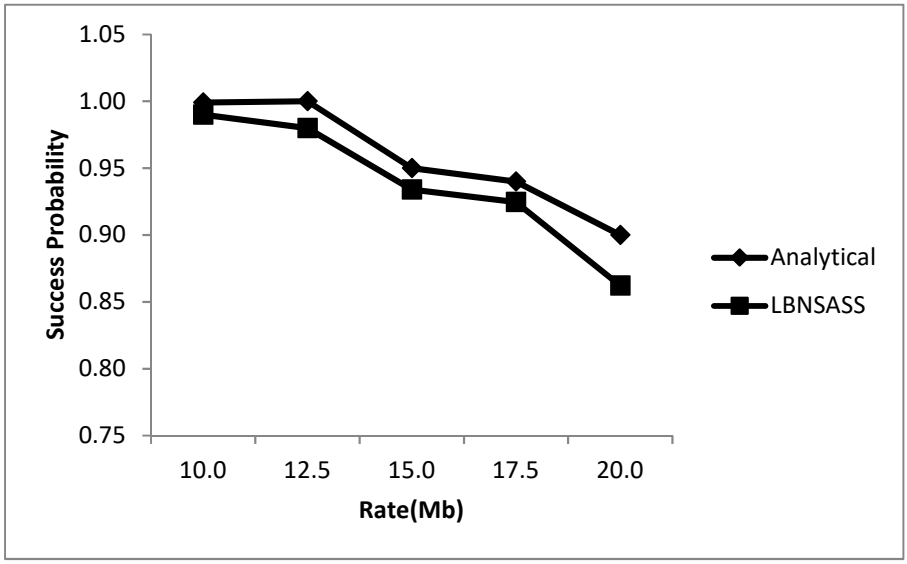

Figure 9 Success Probability for varying Rate

Figure 9 shows the comparison of Success probability values for simulation and analytical results. It can be observed that difference between both types is nearly 2% only.

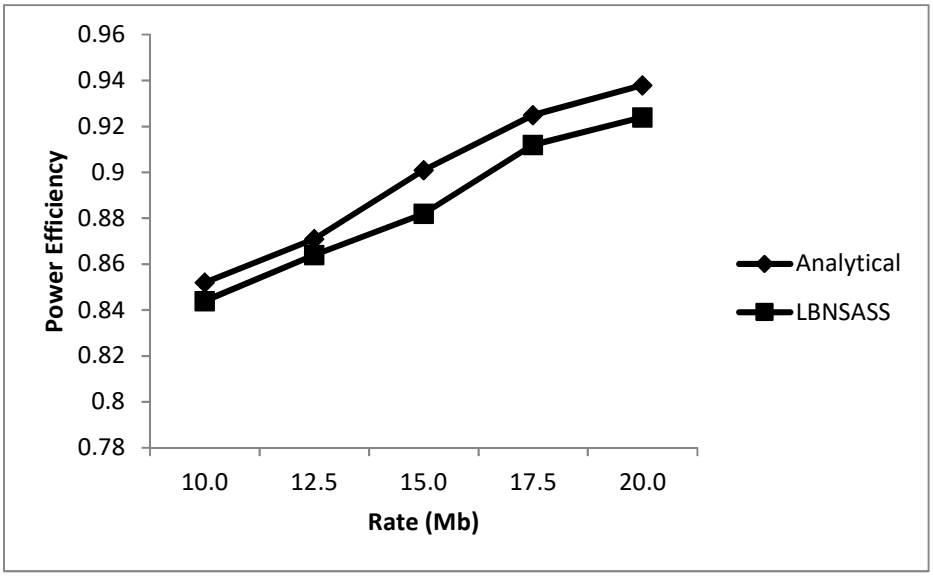

Figure10 Power Efficiency for varying Rate

Figure 10 shows the comparison of power efficiency values for simulation and analytical results. It can be observed that difference between both types is nearly 1.3% only.

5. Conclusion

In this paper, we propose to design a load balancing technique based on network segmentation and adaptive sleep scheduling for 5G-IoT networks.In this technique, for each network segment,sub-segments are formed and grouped to process IoT applications having different QoS requirements. The improved DBSCAN algorithm is used for sub-segmentation and grouping. based on a combined rank of QoS factors. In the next phase, each SBS executes adaptive dynamic sleep scheduling based on its load level. In the load balancing policy of SBS, when the average load of any SBS increases beyond the load of the Macrocell, the overloaded traffic can be progressed to the Macrocell.

The proposed LBNSASS technique is implemented in NS2 and compared with the MLBNSFtechnique. In simulation experiments, the traffic load is varied to measure the energy consumption, success probability, computational overhead, packet drop and power efficiency metrics. Simulation results are validated against the analytical results and hence prove that the proposed LBNSASS technique achieves higher success probability and power efficiency with reduced energy consumption and packet drops of SBSs when compared to MLBNSF technique.

Declaration:

This manuscript was originally prepared by me and my co-authors. This is not submitted to any-where for Publication.

Authors Contribution – All the authors are equally contrbuted

Funding – Not Applicable

Competing Interests – The authors don’t have any Competing Interests

Data Availability - The authors didn't use any third party data on their manuscript.

Code Availability – The authors have the implemented code.

References

1. MostafaMouawad , ZbigniewDziong and Ahmed El-Ashmawy, "Load Balancing in 5G C-RAN based on dynamic BBU-RRH Mapping Supporting IoT Communications", *IEEE Global Conference on Internet of Things (GCIoT)*, 2018.

2. Xiang Sun and Nirwan Ansari, "Traffic Load Balancing among Brokers at the IoT Application Layer", *IEEE Transactions on Network and Service Management*, vol. 15, no. 1,,pp. 489-502, March 2018.

3. Pol Serra i Lid´on,GiuseppeCaso, "Two-tier Architecture for NB-IoT: Improving Coverage and Load Balancing", *IEEE*, 2019.

4. TrungKienVu,MehdiBennis,SumuduSamarakoon,andMattiLatva-aho, "Joint Load Balancing and Interference Mitigation in 5G Heterogeneous Networks", *IEEE*, 2017.

5. Christos Tsirakis, PanagiotisMatzoros, PetrosSioutis and George Agapiou, "Load balancing in 5G Networks", *MATEC Web of Conferences*, Vol-125,2017.

6. Rashad.T.S.and A.Ch. Sudhir," Fuzzy-Neural based Cost Effective Handover Prediction Technique for 5G-IoT networks", *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, Volume-9 Issue-2S3, December 2019, ISSN: 2278-3075

7. Sushil Kumar Singh ,Mikail Mohammed Salim, Jeonghun Cha , Yi Pan and Jong Hyuk Park, "Machine Learning-Based Network Sub-Slicing Framework in a Sustainable 5G Environment", *Sustainability*,Vol-12, 2020.

8. KaigeQu,WeihuaZhuang,QiangYe,Xu Li, and Jaya Rao, "Traffic Engineering for Service-Oriented 5G Networks with SDN-NFV Integration", *IEEE,*2018.

9. Zhu Xiao, Shuangchun Li, XiaochunChen,Dong Wang and Wenjie Chen, "A Load-Balancing Energy Consumption Minimization Scheme in 5G Heterogeneous Small Cell Wireless Networks Under Coverage Probability Analysis", *International Journal of Pattern Recognition and Artificial Intelligence*,Vol. 31, No. 7,2017.

10. MisikirEyobGebrehiwot, PasiLassila and Samuli Aalto, "Dynamic load balancing in 5G HetNets for optimal performance-energy tradeoff", *IEEE,IFIP*, 2018.

11. Martin Ester, Hans-Peter Kriegel, Jiirg Sander, XiaoweiXu,"A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise", *KDD-96 Proceedings*, 1996

12. Xiaolu Li , Peng Zhang and GuangyuZhu,"DBSCAN Clustering Algorithms for Non-UniformDensity Data and Its Application in Urban Rail Passenger Aggregation Distribution", Energies, *MDPI*, September 2019