EAI Endorsed Transactions
on Mobile Communications and Applications

Research Article ICST.ORG

Load Balancing Techniques for Dynamic Gateway Selection in LTE Wireless Networks

Sakshi Patni[1], Aditya Hegde[1] and Krishna M. Sivalingam[1,*]

[1]Dept. of CSE, Indian Institute of Technology Madras, Chennai 600036, INDIA
India-UK Advanced Technology Centre of Excellence in Next Generation Networks, Systems and Services (IU-ATC), Chennai, INDIA

Abstract

In this paper, we consider a Long Term Evolution (LTE) network. The user equipment (UE) nodes connect to the Internet through packet gateway nodes. Multiple P-GWs in the LTE core network forward these packets. Thus, based on how the sessions are established, different gateways may handle different levels of network traffic. If proper load balancing algorithms are not used, the load on some gateways may be higher than others leading to Quality of Service (QoS) degradation. We propose six different heuristics for load balancing, including one based on entropy called EBA. The entropy function provides an accurate measure of the difference in the load on the gateways. The performance studies show that the EBA algorithm provides good performance improvement for networks consisting of 1,00,000 users, 12 APNs and 50 P-GWs.

Received on 14 December 2015; accepted on 07 April 2016; published on 20 June 2016

Keywords: LTE networks, Packet Gateway Selection, Mobility Management Entity, Load Balancing

doi:10.4108/eai.20-6-2016.151518

1. Introduction

Long Term Evolution (LTE) is a fourth generation mobile technology that can provide data rates up to 100 Mbps on the downlink and 50 Mbps on the uplink [1, 2]. In an LTE , several user equipment (UE) mobile devices connect to the Internet through *enodeB* (base-station) and Packet Gateway (P-GW) nodes. The Access Point Name (APN) is a configurable identifier used to specify the network that the UE wants to connect to. Each APN is associated with a number of P-GWs. Hence, when a connection request arrives from a UE using a given APN, the connection is established on one of the P-GWs that the APN is configured to use.

The LTE network has several such P-GW nodes serving all the UE nodes. Each P-GW has limited capacity in terms of link capacity, memory, CPU speed, buffer space and IP address pool. When any of these resources is used completely, no more connections can be set up through that P-GW. The load on the P-GW directly impacts the response time of the packet at the P-GW. Thus, load balancing techniques are required to ensure that all the P-GWs are balanced and have approximately similar load at all times. *The focus of this paper is to develop efficient load balancing algorithms that assign UEs and APNs to the most suitable P-GW.*

Existing load balancing algorithms use static weights for the P-GWs and try to balance the load using weights based on capacities relative to other P-GWs. This kind of load balancing is usually done for traffic associated with bearers of QCI value of 9 which have delay budget of 300 ms [3]. Since different P-GWs will be associated with different number of APNs and different APNs may have different number of active users, this measure may not always lead to optimal load balancing. Thus, dynamic load balancing algorithms that provide a better distribution of the load on the gateways. are studied in this paper.

In the proposed algorithms, the gateways are assigned weights that depend on different metrics that

[★] Part of this paper was presented at International Conference on Distributed Computing and Networking (ICDCN), Coimbatore, India, Jan. 2014.

[*] Corresponding authors. Email: krishna.sivalingam@gmail.com

Table 1. Table of Abbreviations.

Abbreviation	*Expansion*
AMBR	Aggregate Maximum Bit Rate
APN	Access Point Name
DW	Dynamic Weights
E-UTRAN	Evolved Universal Terrestrial Radio Access Network
EBA	Entropy Based Assignment
EPC	Evolved Packet Core
GBR	Guaranteed Bit Rate
HSS	Home Subscriber Server
LBT	Load Balancing Threshold
LTE	Long-Term Evolution
MAG	Mobile Access Gateway
MME	Mobility Management Entity
NAS	Non-Access Stratum
P-GW	Primary Gateway
PCRF	Policy and Charging Rules Function
RR	Round Robin
S-GW	Serving Gateway
SAAW	Static APN Associated Weights
SNMP	Simple Network Management Protocol
UE	User Equipment
UMTS	Universal Mobile Telecommunication System
eNB	evolved NodeB

are either static or dynamic. In case of dynamic weights, the weight changes based on the load conditions of the P-GWs. The changes in the load are informed to the Mobility Management Entity (MME) by the P-GWs. There exists a trade-off between load balancing and the overhead in sending the updates.

Six algorithms have been proposed for balancing the load among the P-GWs. These are: Load Balancing Threshold (LBT), Static APN Associated Weights (SAAW), Dynamic Weights (DW), Entropy Based Assignment (EBA), Round-Robin (RR) and Randomized Load Balancing. The LBT algorithm uses different threshold intervals for sending the load updates from the P-GWs to the MME. The SAAW algorithm uses static weights for the P-GWs. Only certain P-GWs can be used for connections from a particular APN. The EBA algorithm uses entropy function to see which P-GW to use for a connection. RR based assignment assigns the P-GWs in a circular fashion. The randomized algorithm assigns P-GWs randomly from the available pool.

The performance of the algorithms has been studied using Java and ns3 (a network simulator) based simulation models. It is seen that the SAAW algorithm improves the performance of the static algorithm by 13% and requires no overhead for sending load updates from the P-GW to the MME. If the number of active users keep changing frequently, then the DW algorithm gives an exact measure of the load on the P-GW. The EBA algorithm is shown to improve performance as much as 18% for large topologies with 1,00,000 users, 12 APNs and 12 P-GWs.

2. Background and Related Work

This section provides a brief overview of LTE wireless networks and the existing load balancing techniques for packet gateway assignment. The set of abbreviations used in the paper is summarized in Table 1.

2.1. Long Term Evolution (LTE)

Long Term Evolution (LTE) [1, 2], a 4G technology, was developed by Universal Mobile Telecommunication System (UMTS). The technology supports higher data rates as compared to 3G and other mobile networks. The control and user plane latencies are also comparatively low and thus improve the overall performance of the system. The system architecture of LTE is presented in Fig. 1. Here, evolved NodeBs (eNBs) are the base stations to which the User Equipments (UEs) connect to. The Core Network (CN) of LTE consists of Mobility Management Entity (MME), Serving Gateway (S-GW) and Packet Data Network Gateway (PDN-GW or P-GW). The MME takes care of user mobility, intra-LTE handover and user authentication. The S-GW is responsible for supporting handover between the LTE nodes and the other 3GPP technologies. The P-GW is used to connect to the external packet data networks. The data path between

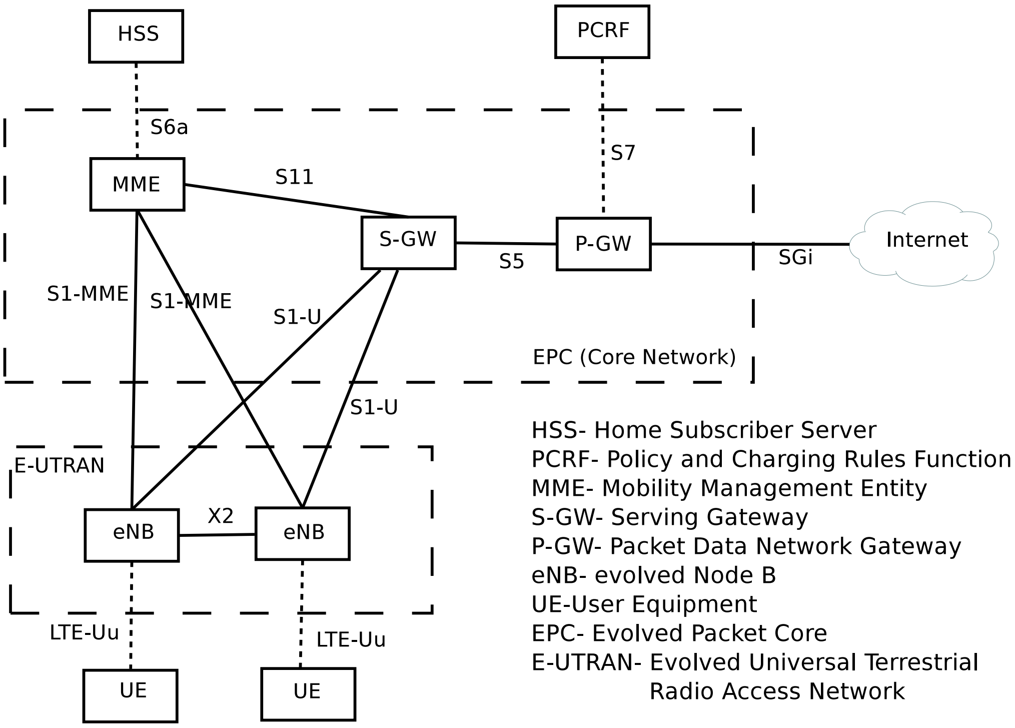

HSS
PCRF
S6a
S7
MME
S11
S-GW
P-GW
S5
SGi
Internet
S1-MME
S1-MME
S1-U
S1-U
EPC (Core Network)
E-UTRAN
eNB
X2
eNB
LTE-Uu
LTE-Uu
UE
UE
HSS- Home Subscriber Server
PCRF- Policy and Charging Rules Function
MME- Mobility Management Entity
S-GW- Serving Gateway
P-GW- Packet Data Network Gateway
eNB- evolved Node B
UE-User Equipment
EPC- Evolved Packet Core
E-UTRAN- Evolved Universal Terrestrial Radio Access Network

Figure 1. LTE System Architecture.

the UE and the P-GW is established through S-GW and the control path is established through MME. The Home Subscriber Server (HSS) maintains all the user information which is used by MME for authentication and authorization. The Policy Charging and Rules Function (PCRF) is used to charge the users according to the services used by them.

The P-GW or PDN-GW is the network entity that provides access to external packet data networks such as Internet or IP Multimedia Subsystem (IMS) networks. These entities provide the functions such as packet filtering, providing Quality of service (QoS), IP address allocation, charging and policy enforcement via the PCRF. For each UE, there is one or more P-GW serving its requests.

The Mobility Management Entity (MME) exchanges the control signals with the UE through Non-Access Stratum (NAS) signaling. It is also responsible for the selection of S-GW and P-GW. The MME keeps track of the location of the UE by Tracking Area Update (TAU) of the user. It also selects other MMEs for handover.

2.2. Bearer Establishment

A bearer uniquely identifies the flows which receive a common treatment between the UE and the P-GW. A unique bearer exists for each combination of a particular QoS and IP address of the terminal. A Guaranteed Bit Rate (GBR) bearer [2] is associated with a dedicated EPS bearer and provides a guaranteed minimum transmission rate in order to offer constant bit rate services for applications such as interactive voice that require deterministic low delay service treatment. For a group of non-GBR bearers, Aggregate Maximum Bit Rate (AMBR) [2] denotes the bit rate of traffic for a group of bearers destined for a particular PDN. The Aggregate Maximum Bit Rate is typically assigned to a group of Best Effort service data flows over the Default EPS bearer.

When a UE initially attaches to the network, a *default* bearer is established between the UE and the P-GW. This bearer remains established to provide always-on connectivity to the UE. The default bearer will be in non-GBR mode since it will remain established for long periods. Any additional EPS bearer that is established to the same PDN is referred to as a *dedicated* bearer.

A UE needs to register with the network to receive certain services. This registration is described as Network Attachment. The always-on IP connectivity for UE/users of the EPS is enabled by establishing a default EPS bearer during Network Attachment. The PCC rules applied to the default EPS bearer may be predefined in the PDN-GW and activated in the attachment by the PDN-GW itself. The Attach procedure may trigger one or more Dedicated Bearer Establishment procedures to establish dedicated EPS bearer(s) for that UE. During the attach procedure, the UE will request for an IP address allocation.

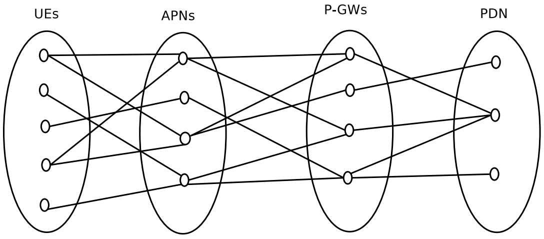


Figure 2. UE, APN and P-GW Association.

2.3. APN and P-GW Association

The Access Point Name (APN) is a configurable identifier that identifies the P-GW and thus the packet data network (PDN) to which a UE wants to connect to. An example APN would look like: `Fast.T-Mobile.com`. If the APN is not present in the request, the MME gets the default APN from the HSS and uses it for the connection. The association between the UEs, APNs and P-GWs is shown in Fig. 2. Also, each UE can have multiple sessions over each APN. A UE can also use multiple APNs to connect to different PDNs for different services including IMS and Internet. It is assumed that the UE is provided with the APN selection – this aspect is not considered in this study.

The Mobility Anchor Gateway (MAG) function on the S-GW can maintain multiple PDN connections for the same user session. Each APN is associated with a number of P-GWs to connect the UE to the PDN. The MME while selecting the P-GW using "PDN-GW selection function" uses the APN to get the list of allowed P-GWs for that APN.

2.4. Load Balancing among P-GWs

There are two existing commonly-used methods discussed for load balancing among P-GWs: static load balancing and inter-GW load balancing as explained below.

Static Load Balancing. The basic method [2] uses static weights for the P-GW based on its capacity relative to other P-GWs. These weights are given to the P-GWs so that the traffic can be distributed according to the capacities of the P-GWs. Thus, if the capacities of the P-GWs 1, 2, 3 and 4 are 1 Gbps, 2 Gbps, 750 Mbps and 900 Mbps, the weights assigned to them will be 1, 2, 0.75 and 0.9.

Inter-GW Load Balancing. In [4], the author has proposed inter-Gateway load balancing approaches. The P-GW transfers the connection to other P-GWs if the load on it is more than other P-GWs. Two types of load balancing approaches are discussed: centralized and distributed. In centralized inter-GW load balancing, all the GWs report their load conditions to a control plane gateway (CP_GW). The CP_GW takes decision about transferring connections from one GW to another depending on their load conditions. In distributed inter-GW load balancing, all the GWs exchange load information with their neighbors, so that they all know about the load conditions on each other. If one of the GWs experiences heavy load, they ask their less loaded neighbors if they can accept some handovers from it. After getting a positive acknowledgement from its neighbor, the GW can transfer some connections to the other GW.

We next present some basic definitions for entropy that are used in the rest of this paper.

2.5. Entropy

Given a probability distribution, there is some uncertainty associated with it. According to the maximum-entropy principle, given some partial information about a random variable, we should choose that probability distribution for it, which is consistent with the given information, but has otherwise maximum uncertainty associated with it. Let the probabilities of n possible outcomes $A_1, A_2, ..., A_n$ of an experiment be $p_1, p_2, ..., p_n$ respectively giving rise to the probability distribution:

$$P = (p_1, p_2, ..., p_n); \sum_{i=1}^{n} p_i = 1; p_i \geq 0 \tag{1}$$

Shannon suggested the following function to measure entropy [5]:

$$H_n(p_1, p_2, ..., p_n) = -\sum_{i=1}^{n} p_i \ln p_i \tag{2}$$

The measure suggested by Shannon can be clearly seen as a function of all the probabilities. It is both a continuous and a symmetric function when $0 \ln 0$ is replaced by 0. The maximum value is reached when all the outcomes are equally likely i.e. when $p_i = \frac{1}{n} \forall i = 1, 2, ..., n$, and the maximum value is $\ln n$. Since $\ln n$ is a monotonically increasing function, the value of H_n increases as n increases.

Table 2. Table of Symbols.

Symbol	*Definition*
a_i	APN Identifier
u_j	UE Identifier
b_k	Bearer Identifier
L	PGW Load
$\mathcal{B}$	PGW Capacity
δ	PGW Bandwidth Utilization
B	Connection Bandwidth Request
v_j	Weight of APN j at a PGW
W_ij	Weight for PGW i at APN j
n	Number of PGWs selectable for an APN

3. Proposed Load Balancing Algorithms

This section presents the proposed load balancing algorithms. The network architecture presented in Fig. 1 is considered. The network consists of a set of user nodes (UEs) that communicate through the serving gateway and the packet gateway to the Internet. As described earlier, each UE is assigned one of the several available packet gateways (P-GW) in the network. The load on the P-GW has a direct impact of the response time of the packet traversing the P-GW. Thus, load balancing techniques are used to ensure that all the P-GWs are balanced and have approximately similar load at all times. This will enable better support of Quality-of-Service to the UE's traffic.

Thus, the problem considered is: how do we efficiently balance the load on the packet gateways (P-GWs) by assigning a UE to the most appropriate packet-gateway?

The following sections present the details of the proposed algorithms. Table 2 presents the list of important symbols used in this section.

3.1. Load definition

We first define the notion of "load" as measured at a P-GW. The metric used to quantify load balancing is a P-GW's carried load relative to its bandwidth. The bandwidth of the P-GW is defined as the amount of data it can transfer or handle per unit time. Different P-GWs may have different bandwidths depending on their configuration. The load on a P-GW is defined as the fraction of the bandwidth that is being used. More specifically, the used bandwidth can be measured as a combination of GBR for the dedicated bearers and AMBR for the default bearers. The amount of bandwidth reserved for the dedicated bearers are based on the number of APNs that share that P-GW, the number of UEs using that P-GW and the number of sessions or bearers each UE has on that P-GW. Similarly, Aggregate Maximum Bit Rate (AMBR) is defined as the bit rate allocated to a group of default bearers for a UE using an APN. Thus, the load is defined as:

$$\begin{aligned} L \;=\; & \sum_{a_i \in \mathrm{APN}} \sum_{u_j \in \mathrm{UE}} \sum_{b_k \in \mathrm{bearer}} \mathrm{GBR}(a_i, u_j, b_k) + \\ & \sum_{a_i \in \mathrm{APN}} \sum_{u_j \in \mathrm{UE}} \mathrm{AMBR}(a_i, u_j) \end{aligned} \tag{3}$$

Let B denote the capacity or the bandwidth of the P-GW. Then, the fraction of the bandwidth utilization of the P-GW is defined as:

$$\delta = \frac{L}{\mathcal{B}} \tag{4}$$

In the algorithms proposed, the major factor used for measuring load will be the bandwidth utilization of the P-GW, i.e. δ; the term *load* may be used interchangeably with δ.

3.2. Load Balancing Threshold Algorithm (LBT)

In this algorithm, a min-heap is maintained for each of the APN at the MME or HSS. The heap will contain all the P-GWs that can be used for that APN. The weights of all the P-GWs are initialized to 1. When a connection request with bit-rate B arrives at the MME from a UE, the MME accesses the heap depending on the APN used by the UE for that connection. The MME then accesses the root of the heap and assigns the connection to the P-GW at the root of the heap. The load on the P-GW increases with the increase in the number of connections. The P-GW keeps monitoring its load and sends an update to the MME as the load crosses a threshold value. Based on the threshold value crossed, the MME updates the weight of the P-GW in all the APNs. The threshold values are defined as intervals having a lower threshold and an upper threshold. If the load on the P-GW crosses an upper threshold of one interval and goes into another interval, the weight at the MME for that P-GW in all APNs is increased.

A sample set of threshold values and their corresponding weights are defined in Table 3. Thus, initially the weights of all the P-GWs will be 1. When the load on any P-GW becomes greater than 1/3, then the weight of that P-GW increases to 2. If it becomes greater than 2/3, the weight becomes 3 and so on. If the load after being greater than 2/3 becomes less than 1/3 for a P-GW, then the weight of that P-GW also decreases and becomes 2.

Table 3. Sample weight factors based on load values

Load	Indicative Weight
$< 1/3$	1
$\geq 1/3$	2
$\geq 2/3$	3
$\geq 7/9$	4
$\geq 8/9$	5

The thresholds used can change depending on the amount of load balancing needed at the cost of increased signalling cost. The cost and load balancing trade-off is examined later.

For the implementation, instead of a min heap, we are maintaining a max heap. It will contain all the P-GWs of the APN. The P-GWs are heapified based on the current available bandwidth. That is, at any given instant, the P-GW with the maximum current available bandwidth will be at the top of the heap. The implementation of the algorithm is as follows:

```
procedure ALGORITHM 1: LOAD BALANCING BASED ON CURRENT AVAILABLE BANDWIDTH
    Description: Assign the optimal P-GW node to the UE
    Let B be the current UE bandwidth requirement
    Maintain a max-heap H at the Load Balancer. Heapify based on the current available bandwidth. Initially the current available bandwidth of each P-GW node is equal to their max capacity.
    optimalsgwPgw = H.top();
    H.pop();
    optimalsgwPgw.currentCapacity = optimalsgwPgw.currentCapacity - B
    H.push(optimalsgwPgw)
    return optimalsgwPgw
end procedure
```

3.3. Static APN Associated Weights Algorithm (SAAW)

This algorithm considers the different APNs a P-GW is associated with and also considers their weight defined by their priority. Suppose each APN at GW_i has weights associated with them, denoted by v_1, v_2, ..., v_k. Assume that the bandwidth of GW_i is $\mathcal{B}_i$. For the implementation purpose, it is assumed that the weights are equal for all APNs. For each APN j with weight v_j at GW_i, the weight W_{ij} for the P-GW is given as:

$$W_{ij} = \frac{v_j}{\sum_k v_k} \times \mathcal{B}_i \tag{5}$$

Based on the above equation, the P-GWs are assigned weights in the APNs. The weights are based on the bandwidth of the gateway and the number of APNs a P-GW is associated with. If the number of active users in each APN is different, then the weights for the P-GWs are slightly modified to include the information related to the number of UEs. Assume that a P-GW is shared by k different APNs with number of active users U_1, U_2, ..., U_k. Then, the weight of the Eq. 5 is modified to:

$$W_{ij} = \frac{v_j}{\sum_k v_k} \times \frac{U_j}{\sum_k U_k} \times \mathcal{B}_i \tag{6}$$

3.4. Dynamic Weights Algorithm (DW)

The initial weights are given to the P-GWs as shown in Eq. 5. The SAAW algorithm will work well if the connections established are not terminated or terminate according to some pattern. The connections terminate randomly and thus the load on the P-GWs changes in a non-deterministic manner. Hence, if we change the weight of the P-GWs dynamically according to the load, then the load distribution will be uniform. The dynamic weights for the P-GW is given as:

$$w_{ij} = W_{ij} \times (1 - \delta) \tag{7}$$

The weight of the P-GW changes based on the load on the P-GW. When the load increases, the factor $1 - \delta$ decreases and the weight of the P-GW decreases than the initial weight and other P-GWs having lesser load will have more chances of getting selected. As the load increases and approaches 1, the weight factor approaches 0 and the P-GW will have less chances of getting selected.

3.5. Entropy Based Assignment (EBA)

The performance of the algorithms is measured with the help of an entropy function, with p_i defined as

$$p_i = \frac{\delta_i}{\sum_j \delta_j} \tag{8}$$

where $p_1 + p_2 + ... + p_n = 1$. Here, the load, i.e. utilization, on each P-GW (δj) is normalized with respect to the total load of all P-GWs. Also, if the load on all the P-GWs is the same, then $p_1 = p_2 = ... = p_n = \frac{1}{n}$ and

$$H_n(p_1, p_2, ..., p_n) = \sum_i -p_i \ln p_i = \ln n \tag{9}$$

This is the maximum value for the Shannon's Entropy function. Thus, if all the P-GWs are equally balanced or have the same amount of load, the value of the Entropy function will be maximum and will be equal to $\ln n$. If one of the $p_i = 1$, then

$$H_n(p_1, p_2, ..., p_n) = \sum_i -p_i \ln p_i = 0 \tag{10}$$

This is the minimum value of the entropy function. Thus, if all the connections are on the same P-GW, then the entropy function will be zero. Therefore, in the load balancing algorithms, the performance of the algorithms can be measured as the value of the entropy function changes. The use of entropy for load balancing has also been considered in other work [6].

The earlier three heuristics consider various factors and then calculate the weight of the P-GW in an APN to get selected. The entropy function is used to see how balanced the P-GWs are with respect to each other. The P-GWs in the previous algorithms were not equally balanced because of the different weights of the APNs. Hence, instead of using other methods, entropy function can itself be used to select a P-GW for a connection. Thus, when a new connection or a session is to be established with a particular APN, the MME selects the P-GW that increases the entropy to the maximum extent. The time required to select a P-GW in this case is $O(n)$, where n is the number of P-GWs that can be selected for that APN. Therefore, the entropy function gives an accurate measure of the difference in loads on different P-GWs; thus, the most appropriate P-GW for a given session is selected.

```
procedure Algorithm 2: Entropy based Load Balancing
    Description: Assign the optimal P-GW node to the UE
    Let B be the current UE bandwidth requirement
    let sgwPgwList be the list of all P-GW nodes.
    Initially the current available bandwidth of each P-GW node is equal to their max capacity.
    optimalsgwPgw = NULL;
    double max = 0;

    for ( do sgwPgw in sgwPgwList)
        sgwPgw.currentCapacity = sgwPgw.currentCapacity - B;
        double entropy = calculateEntropy();
        if entropy > max then
            max = entropy;
            optimalsgwPgw = sgwPgw;
        end if
        sgwPgw.currentCapacity = sgwPgw.currentCapacity + B;
    end for
    optimalsgwPgw.currentCapacity = optimalsgwPgw.currentCapacity - B;
    return optimalsgwPgw;
end procedure
```

Additional details and examples for the above heuristics are available in [7]. We next present two simple heuristics, used mainly for comparison purposes.

3.6. Round-Robin based Load Balancing (RR)

In this algorithm, P-GWs are assigned in a circular fashion starting from the first node in a round-robin manner.

```
procedure Algorithm 3: RR based Loadbalancing
    Description: Assign the P-GW node indexed by modeCounter to the UE
    Let B be the current UE bandwidth requirement and m_sgwPgwCount represents the total number of P-GW nodes in the network
    modAlgoCounter++;
    modAlgoCounter = modAlgoCounter % m_sgwPgwCount;
    sgpgw = m_sgwPgwList[modAlgoCounter];
    sgpgw.m_sgwPgwApp.currentCapacity = sgpgw.m_sgwPgwApp.currentCapacity - B;
    return sgpgw;
end procedure
```

3.7. Randomized Load Balancing

In this algorithm the P-GWs are assigned to the UEs in a random fashion.

```
procedure Algorithm 4: Randomized Load Balancing
    Description: Assign the P-GW node to the UE in a random fashion
    Let B be the current UE bandwidth requirement
    int randomIndex = rand() % m_sgwPgwCount;
    sgpgw = m_sgwPgwList[randomIndex];
    sgpgw.m_sgwPgwApp.currentCapacity = sgpgw.m_sgwPgwApp.currentCapacity - B;
    return sgpgw;
end procedure
```

4. Performance Analysis

To conduct simulation modeling based analysis, a Java based model (without discrete-event simulation facility) of the system was initially written. Subsequently, a subset of the mechanisms was implemented in the ns3 simulator.

4.1. Java Simulator Based Study

The simulation environment used for the modeling of the algorithms consisted of multiple UEs, multiple APNs and multiple P-GWs. Each UE can use only those APNs that it is configured to use. The APNs will be associated with some P-GWs. Only these P-GWs can be selected by the MME for the connection coming from that APN. Each P-GW can have incoming traffic from the UEs of multiple APNs, i.e. each P-GW is also associated with multiple APNs and serves users from each one of them. The simulation is run for different times with varying number of users, APNs and P-GWs. Two kind of scenarios are simulated: one in which no connection is terminated and one in which connections are terminated within the simulation time. The P-GWs are selected for each connection instead of each user. Thus, the granularity for selecting a P-GW is session or connection based.

The number of connections in each second is approximately the ratio of the total number of sessions (average number of sessions × total number of users) to the simulation time. Each time a connection is set-up on a P-GW, the load is updated and the entropy is calculated.

4.2. Parameters and Metrics

The parameters used for simulating the analysis are presented in Table 4. The number of UEs, number of APNs and number of P-GWs are varied. The bandwidth of the P-GWs is set to either 1 Gbps, 2 Gbps or 750 Mbps. Different termination time is set for connections in the scenarios where connections are terminated. Each simulation experiment is run for 500 seconds.

The metrics analyzed are the entropy function value that captures the load balancing; and the costs involved

Table 4. Parameters

Parameter	Description
P-GW bandwidth	0.75, 1 or 2 Gbps
Avg. sessions per user	3
Simulation time	500 seconds
Bit rate for video traffic	64 Kbps
Bit rate for VoIP call	17 Kbps
Bit rate for video streaming	12 Kbps
Bit rate for live streaming	64 Kbps
Bit rate for gaming applications	100 Kbps

in some of the algorithms for dynamic weight update. The cost is based on the number of times the update has to be sent to the MME by the P-GW.

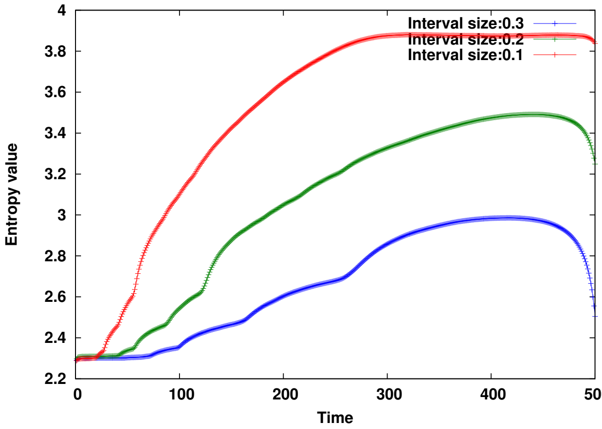


Figure 3. Performance of LBT algorithm with 1,00,000 users, 12 APNs and 50 P-GWs.

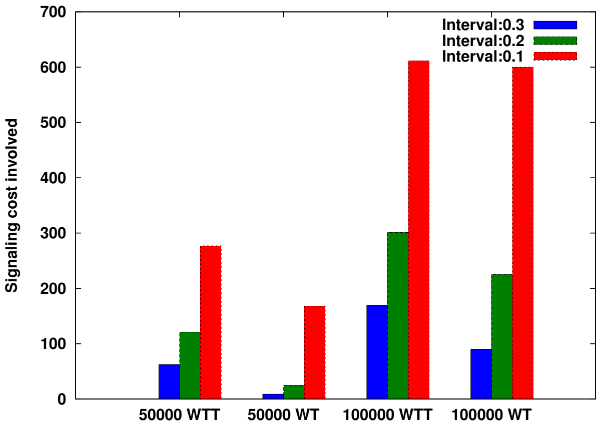


Figure 4. Signaling cost for sending updates in LBT.

4.3. Performance Results

The performance results for the different algorithms using the Java based simulator are presented in this section.

LBT algorithm. The LBT algorithm that uses thresholds for sending the updates to the MME is not compared to any other algorithm for performance since defining the thresholds means reducing the load balancing. Three threshold-based schemes are used for showing the signaling cost to entropy value trade-off. The first scheme uses unequal threshold interval sizes. The threshold intervals used are in the set {0.3, 0.6, 0.7, 0.8, 1.0}. The second scheme uses threshold intervals of size 0.2. Thus, the intervals are {0.2, 0.4, 0.6, 0.8, 1.0}. The third scheme uses threshold intervals of size 0.1. These three schemes are compared with each other for signaling cost and entropy values.

The experiments were run for a system with up to 1,00,000 users, 12 APNs and 50 P-GWs. The results are shown in Fig. 3 for the entropy difference in the values. Fig. 4 shows the signaling cost involved when different schemes are used. The trade-off can be observed from the figures. The entropy value is high for the intervals of size 0.1, but the signaling cost incurred is also high. The signaling cost specifies the number of times the update is to be sent to the MME or the HSS when the threshold interval is being crossed.

Fig. 4 shows four different cases where two cases are for 1,00,000 users and two are for 50,000 users, 12 APNs and 50 P-GWs. Each case is shown for two different types: one in which connections are terminated and one in which they are not. The legends WTT and WT respectively represent 'Without Termination' and 'With Termination'. Both the cases perform roughly the same in terms of entropy values; however, the entropy value starts decreasing at the end in case of terminating connections. In case of signaling cost, the signaling cost is less when connections are terminated since the load on the P-GWs rarely crosses the thresholds and typically remains in one threshold interval for a long time.

Performance Comparison. The next set of experiments are run for random topologies. The number of UEs is 1,00,000, number of P-GWs is 50, the number of APNs is 12 and the number of MME is 1. The simulation is run for 500 seconds. The average number of sessions that each UE can establish is 3. The simulations are run for two cases: one in which no connection is terminated and the other in which connections are terminated after some time. The time after which the connections terminate are selected randomly. The results are shown in Fig. 5. The average entropy values are 3.78, 3.83, 3.83 and 3.88 for static algorithm, SAAW algorithm, DW algorithm and EBA algorithm

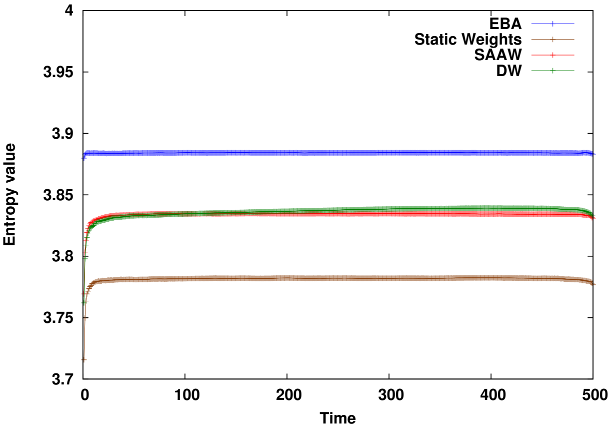


Figure 5. Comparative analysis for 1,00,000 users, 12 APNs and 50 P-GWs.

respectively when connections are terminated. Thus, the overall performance improvement from static algorithm to EBA algorithm is 10% whereas the SAAW and DW algorithm improves the static algorithm by 5%. The improvement calculations are done based on raising the entropy values to the respective powers of e. For instance, $(e^{3.88} - e^{3.78})/e^{3.78} \approx 10\%$.

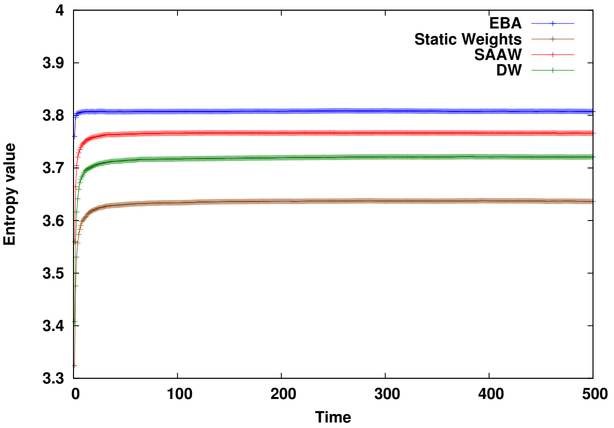


Figure 6. Comparative analysis with different UEs in each APN.

In a mobile network, the number of users associated with different APNs is usually different. Hence, the experiments are run for cases where different APNs have different number of UEs associated with them. Some APNs will be corporate APNs and will have less number of users associated with it than other mobile operators' APNs. The simulation here is run for 1,00,000 users, 12 APNs and 50 P-GWs. The entropy value varying with time is shown in Fig. 6. The graph shows the case where connections are terminated randomly since the algorithm performs the same in both the cases. The performance improvement achieved is 19% for EBA algorithm, 13% for SAAW algorithm and 8% for DW algorithm as compared to the static algorithm.

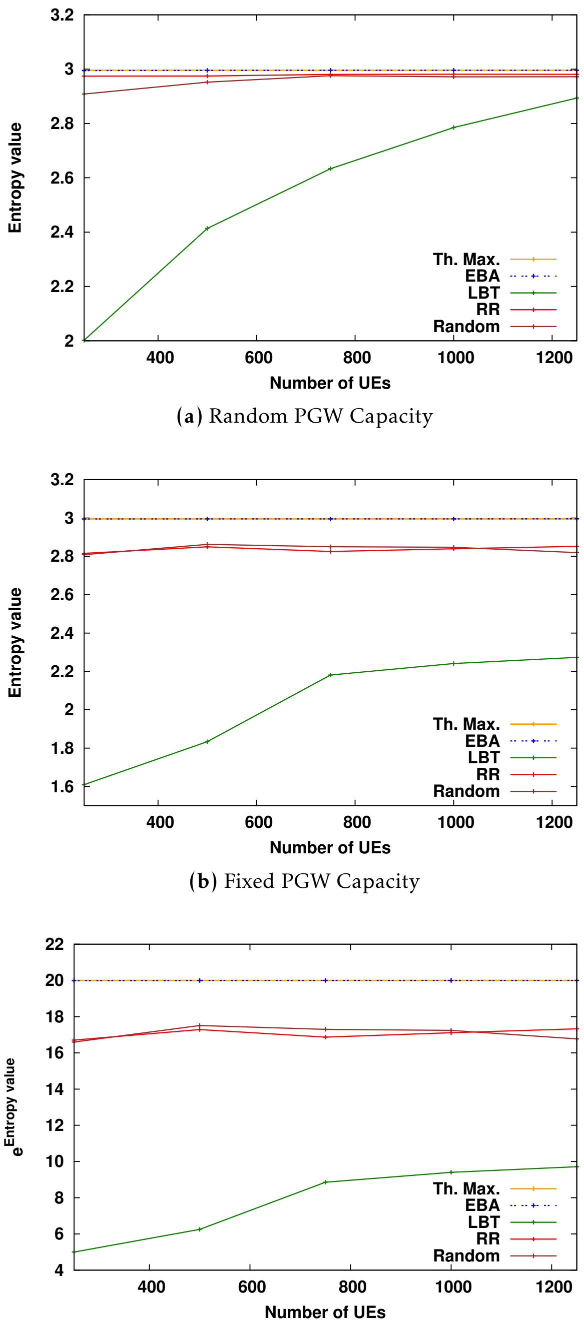
The DW algorithm in this case does not perform as well as SAAW algorithm. This is because, in the latter scheme, the weights are calculated on the basis of number of active UEs associated with an APN and the number of APNs associated with a P-GW. Thus, the algorithm performs well as long as the number of active users are actually what is being predicted. In case of DW algorithm, the weights change dynamically and thus, do not reflect the number of active UEs information anymore. Thus, the SAAW algorithm performs better in this case.

This section presented some of the key performance results. Additional details and performance studies are available in [7].

4.4. ns3 Simulator Based Study

A subset of the algorithms were studied by implementing them in a modified *ns3* LTE simulation framework [8]. Every eNB has a software running called "enbApplication" that connects to the P-GW and establishes a S1-U bearer based on UE requests. Since there are multiple P-GWs, the only choice is to replicate these applications and run on eNB. The number of enbApplications is equal to the number of P-GWs. Each enbApplication running on the eNB connects to the specified P-GW. Instead of a one-to-one mapping between an eNB and a P-GW, we have a one-to-one mapping between the enbApplication and P-GW and a given eNB hosts multiple such enbApplications. A hashmap is used to store the application *association* information. Whenever a UE enters a cell, the MME runs the Load Balancing algorithm and assigns the P-GW to the UE. Once the P-GW node is decided based on the cell identifier of the UE, the corresponding enbApplication could be chosen to establish the connection with the P-GW. The LBT, EBA, RR and Random algorithms have been implemented in ns3. SAAW could not be implemented since ns3 provides only one APN per P-GW; and DW could not be implemented since ns3 does not presently support dynamic topologies. Since the EBA mechanism was shown to have the best performance in the previous section, this mechanism is the main focus in the remainder of this paper. The algorithms have been tested under various scenarios and the results are described below.

Fig. 7(a) presents the performance for a scenario where, the number of eNBs is 25; the number of PGWs is 20; UEs per eNB is varied from 10 to 50; the randomly assigned capacity of the PGWs is: $\{976, 1115, 1211, 1059, 1399, 1421, 1008, 1133, 1489, 1222, 922, 1274, 1121, 1027, 1252, 1372, 1493, 919, 1419, 1269\}$ (in Mbps); the traffic generated by each UE is between 1 to 10 Mbps. As can be seen, the EBA algorithm performs best. The maximum and minimum performance gains of EBA over



(a) Random PGW Capacity

(b) Fixed PGW Capacity

(c) Entropy, raised to *e*

Figure 7. Performance comparison, varying the number of UEs for 25 enodeBs and 20 PGWs: (a) Randomly assigned PGW capacity; (b) Fixed capacity; (c) Entropy, raised to power of *e*.

LBT are 169% and 10.7% respectively; of EBA over RR are 2% and 1.4% respectively; and of EBA over RANDOM are 9% and 2.4%. In this scenario, the advantage of EBA over RR and RANDOM is not significant.

Fig. 7(b) presents the performance for a scenario where the capacity of PGWs is selected from the fixed set $\{500, 1000, 1500, 2000\}$ Mbps and not random as earlier. In this case, the maximum and minimum performance gains of EBA over LBT are 299.7% and 105.9% respectively; of EBA over RR are 19.6% and 15.4% respectively; and of EBA over RANDOM are 20.4% and 14.2% respectively. In this scenario, the improvement offered by EBA over the other schemes is seen to be significant. Fig. 7c presents the same entropy values, raised to power of e, to better demonstrate the magnitude of the improvement in load-balancing.

Thus, the results show that the EBA algorithm can provide the best performance among the algorithms studied, with a higher cost of signaling. Additional ns3-based results are available in [9].

5. Summary

In this paper, various P-GW load balancing algorithms for LTE networks were presented. The load balancing effectiveness of these algorithms was compared using an entropy measure. The cost including the signaling and updates from the P-GW to the MME was also analyzed.

The Entropy based algorithm (EBA) performed 19% better than the static weights algorithm when only one MME was used and different APNs have different number of active UEs associated with it. For the scenario with large max-min bandwidth value, the EBA algorithm performed a minimum of 106% better than LBT, 15% better than RR and 14% better than Random algorithm. The EBA algorithm requires higher signaling cost since the updates related to the load on the P-GWs have to be sent to the MME or the HSS. The algorithm also needs $O(n)$ time for selecting the optimal P-GW where n is the number of P-GWs associated with a particular APN.

The load balancing algorithms could be also applied to the S-GW, in case it becomes a bottle neck. In our simulation experiments, a single node was serving the roles of both S-GW and P-GW; hence, the simulation results ideally capture the balanced P-GW and S-GW case. The proposed schemes also can be implemented in a testbed for further practical studies.

Acknowledgements:

Part of this work was supported by India-UK Advanced Technology Centre of Excellence in Next Generation Networks, Systems and Services (IU-ATC).

References

[1] A. Ghosh, R. Ratasuk, Essentials of LTE and LTE-A, Cambridge University Press, 2011.

[2] 3GPP, General Packet Radio Service (GPRS) enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) access, 2011.

[3] 3GPP, Technical Specification, Policy and Charging Control Architecture (Release 8), number TS 23.203 in TS, 2009. URL: http://www.quintillion.co.jp/3GPP/Specs/23203-831.pdf, V8.3.1.

[4] C. Xue, J. Luo, R. Halfmann, E. Schulz, C. Hartmann, Inter GW Load Balancing for Next Generation Mobile Networks with Flat Architecture, in: Proc. IEEE Vehicular Technology Conference (VTC), 2009, pp. 1–5. doi:doi:10.1109/VETECS.2009.5073605.

[5] C. Shannon, A mathematical theory of communication, Bell System Technical Journal 27 (1948) 379–423.

[6] M. Tuba, Asymptotic Behavior of the Maximum Entropy Routing in Computer Networks, Entropy 15 (2013) 361–371.

[7] Sakshi Patni, Dynamic Gateway Selection For Load Balancing In LTE Networks, Master's thesis, Department of CSE, Indian Institute of Technology Madras, 2013.

[8] ns3 Team, ns3: a discrete-event network simulator for Internet systems, https://www.nsnam.org, 2016.

[9] Aditya Hegde, Network Centric and User Centric Load Balancing In Wireless LTE Networks, Master's thesis, Department of CSE, Indian Institute of Technology Madras, 2014.

6. Appendix

This section present the Linear Programming formulation of the problem.

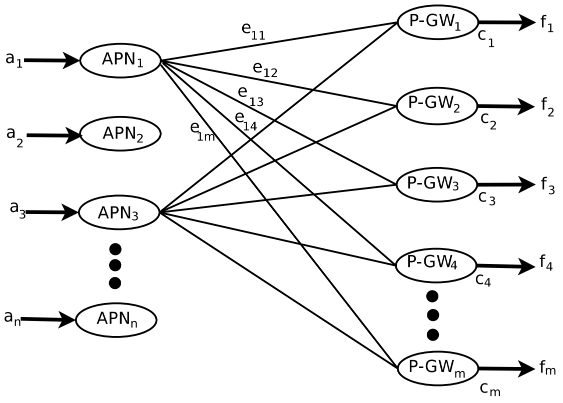


Figure 8. Linear Programming Formulation for Load Balancing.

The load balancing problem can be modeled as a linear programming (LP) problem. If the problem can be formulated as an LP problem, exact values for the loads on all the P-GWs can be given and all the P-GWs will have equal amount of load. The LP formulation is shown in Fig. 8. The figure shows the APNs generating and sending the traffic to the P-GWs. Assume there are m P-GWs and n APNs. The traffic coming from each APN is described as a_i which is calculated as:

$$a_i = A_i \sum_{j=1}^{k} S_{ij} \lambda_{ij} \tag{11}$$

Here, A_i is the number of active users using the APN$_i$, k denotes the total types of sessions, S_{ij} denotes the average number of sessions of type j at APN$_i$ and λ_{ij} denotes the arrival rate of session S_{ij} at APN$_i$. The total traffic at each APN can change as the number of active users and the average number of sessions change. This information has to be periodically updated based on the past history of the users.

The total inflow at the j^{th} P-GW is denoted as f_j and the capacity is denoted as c_j. The problem here is to distribute the load or incoming traffic from all the APNs to the P-GWs such that the load on all the P-GWs is approximately same. For distribution of the loads, a weight is associated from each APN to each P-GW. The APN$_1$ is connected to all the P-GWs as can be seen in Fig. 8 with weights $e_{11}, e_{12}, ..., e_{1m}$. If some P-GWs are not associated with a particular APN, then the weight of that P-GW can be zero. The weights actually denote the amount of traffic that should flow from the APN to the P-GWs. The incoming traffic on the P-GWs should be proportional to their capacities. This is measured through load on the P-GW which is the ratio of the total incoming flow to the total capacity of the P-GW. The load on the j^{th} P-GW is given as:

$$\delta_j = \frac{f_j}{c_j} \tag{12}$$

The formulation for the load balancing among the P-GWs is given as:

$$\text{minimize} \quad F = max\left(\frac{f_j}{c_j}\right) \quad 1 \le j \le m \tag{13a}$$

$$\text{subject to} \sum_{j=1}^{m} e_{ij} = a_i \qquad \forall 1 \le i \le n \tag{13b}$$

$$\sum_{i=1}^{n} e_{ij} = f_j \qquad \forall 1 \le j \le m \tag{13c}$$

$$f_j \le c_j \qquad \forall 1 \le j \le m \tag{13d}$$

$$e_{ij} \ge 0 \quad \forall 1 \le i \le n \quad \forall 1 \le j \le m \tag{13e}$$

The link utilization or the objective function will always have value between 0 and 1. The problem of Eq. 13 is not an LP since the objective function is non-linear. These kind of functions are known as piece-wise linear functions.

Example: Consider the following example with the number of APNs and P-GWs set to 3 each. Assume that the incoming traffic is calculated on the basis of number of active users, average number of sessions per user and the arrival rate of each type of session. Let the incoming traffic be 100 Mbps from APN$_1$, 50 Mbps from APN$_2$ and 200 Mbps from APN$_3$. Let the capacities of the P-GWs be 200 Mbps, 300 Mbps and 400 Mbps respectively. Using Matlab, the values of the variables e_{11} to e_{33} are found to be 52.59, 47.42, 0, 25.19, 0, 24.81, 0, 69.26 and 130.74 respectively. The values of f/c for the P-GWs 1, 2 and 3 are 0.38. Additional details and results for the above formulation are available in [7].

However, the above formulation does not scale well for large number of entities. Another problem with LP formulation is that the amount of total incoming traffic can only be predicted to a certain extent limiting the amount of load balancing that can be achieved. Hence, several heuristics that consider dynamic traffic patterns and can scale well for large number of nodes are proposed in this paper.