Efficient Downlink Video Transmission in LTE Using Mobility Predictions

Mohamed Shoeb
Department of Electrical and Computer Engineering
Queen's University
Kingston, Canada
mohamed.shoeb@queensu.ca

Ashraf Khresheh
School of Computing
Queen's University
Kingston, Canada
khashraf@cs.queensu.ca

***Abstract*—In this paper, we develop an implementation of the distributed heuristic predictive Medium Access Control (MAC) scheduler proposed in [2]. The predictive scheme is implemented using the Network Simulator (NS-3). To realize the distributed scheme, we devised and utilized a specialized network element which we denote as the Global Persistence Module (GPM). The GPM is, in essence, a bookkeeping database module used to carry and maintain up-to-date information about active User Equipment (UE) residing within the network. Information, such as user channel conditions and video streaming buffer statuses, is stored, updated and retrieved when necessary by various network entities. The GPM module, therefore, facilities independent scheduling decisions to be independently rendered by network elements, such as Base Stations (BSs), thus paving way for fully distributed designs and implementations. We modify the NS-3 LENA framework to incorporate the GPM, and implement the predictive scheduler. Performance of the heuristic scheme is evaluated for a stored video transmission application through TCP/IP in Long Term Evolution (LTE) networks with Evolved Packet Core (EPC) functionality. Metrics such as cumulative allocations, air-time and playback stalls are used to examine and compare performance of the predictive scheduler, as well as the Proportionally Fair (PF) scheduler, under different cell load scenarios and user speeds, assuming a highway mobility model.**

***Index Terms*—Mobility Management, Predictive Scheduling, Energy Efficiency, Video Streaming, Resource Management**

I. Introduction

Video streaming is experiencing unprecedented growth with recent forecasts indicating that it will soon account for 66 percent of the total mobile traffic [1]. Meanwhile, the plethora of navigation and context information available in today's smart phones can facilitate efficient transmission schemes that exploit mobility information. For example, if a user is moving away from a Base Station (BS) or Access Point (AP), the network can temporarily increase the allocated wireless resources to allow users to buffer more video content before poor channel conditions prevail. Doing so allows the BSs to transmit more content in less time, thereby saving energy and transmission resources. Furthermore, the scheme improves user Quality of Experience (QoS) as the requested content is pre-buffered and streaming interruptions are effectively reduced.

In [2], conceptual details and potential gains of such predictive transmission approach are discussed, and a simple distributed allocation heuristic is presented. The heuristic only requires knowledge of the user's streaming buffer and whether channel conditions are improving or deteriorating, and was demonstrated to achieve significant transmission efficiency gains.

In this paper, we implement the distributed heuristic in [2] using the NS-3 network evaluation tool. We specifically implement the proposed MAC scheduler within the LTE LENA framework by modifying multiple network components such to enable information exchange, storage and retrieval of current UE state information from the GPM database module in an on-demand fashion. The persistence module solution was elected for this project since it is a currently utilized within emerging 5G systems, such as in [3], which continuously require up-to-date information about user requests, resource availability and demands, user mobility patterns and day-to-day behavior predictions.

Performance of the implemented predictive scheduling system is evaluated assuming a highway mobility model and active network subscribers are running video streaming applications, which are transmitted over TCP/IP. We analyze the performance of the heuristic scheme under various cell load scenarios with comparisons to the traditional PF scheduler.

The organization of the rest of this paper is as follows. In section II we present and discuss the system model. In section III, we describe our implementation in detail. In section IV, simulation settings are presented, followed by results in section V. Discussion and ideas for future work can be found in sections VI and VII respectively.

II. System Model

We consider a highway mobility scenario where LTE-EPC users are traversing at a steady speed and streaming stored videos of different rates and durations from a remote host. The LTE network consists of two single sector cells. One isotropic antenna is placed on each BS tower. UEs are initially attached to serving BSs based on distance, and then the A2-A4-RSRQ algorithm is used. Figure 1 illustrates the network deployment topology.

Videos are transported over TCP/IP protocol stacks based the Tahoe congestion avoidance scheme. We disable Hybrid-Automatic Repeat Request (HARQ) mechanism. Further, the Radio Link Control (RLC) layer operates in Unacknowledged Mode (UM). The remote host transports data to the Packet Gateway (P-GW) which in turn is transported via an appropriate BS to the respective UE. A single dedicated bearer per UE is established between the UE and P-GW, which is a Non-Guaranteed Bit Rate (NGBR) bearer.

A PGM database module is added to the LTE-EPC network which is responsible of maintaining up-to-date information about each UE. Examples of types of information maintained by this module include sizes of requested videos, amounts of video bytes buffered, IMSI and RNTI numbers, Wideband-Channel Quality Indicators (W-CQIs), associated cell IDs, and others. Each BS runs an instance of the predictive MAC scheduler, which can access the PGM module at any given time. Each MAC scheduler will examine UEs in corresponding cells and decide based on W-CQI changes whether a UE is classified as 'normal' or 'green'. Users entering the cell are initially declared as normal as soon as they establish a connection with the BS. A normal user can be classified later as green when the W-CQI begins to decrease, indicating motion away from a BS. The MAC scheduler maintains the optimization constraints in [2]. For example, normal users receive segments of video with a minimum amount of data which guarantees no video interruptions, degradations or stalls to occur due to an empty receive buffer.

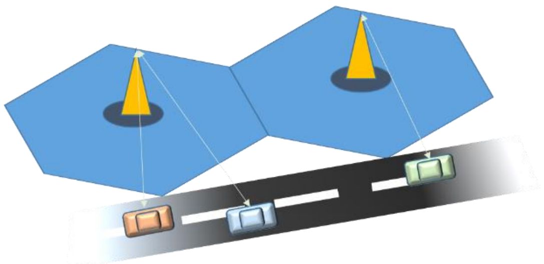

Fig. 1. Network topology and road mobility model

To model typical streaming rates, average encoder rates were obtained from YouTube [4] for different supported quality formats, including 240p, 360p, 480p, 720p, and 1080p. We assume a streaming rate of 1.25 times higher than the video playback rate to provide a margin of clearance to ensure no artificial stalls happen at the end of each video segment received. This selection is determined based on measurements conducted in [5]. The remote host runs an application based on bulk transmission, as in a File Transfer Protocol (FTP) which sends data at the fastest supported rate on the point-to-point connection between the remote host and the P-GW. A sink application is installed on each UE.

III. Implementation

A. PGM

The first module developed was the PGM module. The implementation was done first by creating an NS-3 module PGM with global definitions of the PGM structure denoted Table to contain several elements which are described in table 1. The global database table is declared as extern in **gpm.h** which is a vector of type <*GlobalPersistenceModule*>, as follows

extern std::vector <GlobalPersistenceModule>Table;

The PGM module also contains the following global variables

extern std::vector <double>Stored_Video_Playback_Rates;
extern std::vector <double>Stored_Video_Playback_Sizes;
extern std::vector <double>Stored_Video_Playback_Times;

which carry the video properties of each user, and also,

extern bool cycle_sync;
extern double cycle_slip_guard;
extern uint16_t max_no_of_enbs;

TABLE I. PGM Module Elements

Element	*Function*
videonodeid	Links UE NS-3 node ID with IMSI
imsi	International Mobile Subscriber Identity of the UE
rnti	Radio Network Temporary Identifier of the UE
wcqi_t	W-CQI at current Transmission Time Interval (TTI) index of the UE
wcqi_t_minus1	W-CQI at previous TTI index of the UE
cellid_t	Current Cell ID of the UE
cellid_tminus1	Previous Cell ID of the UE
V	Video streaming rate of this user (bytes/s)
D	Video playback demand in bytes
R_t	Downlink achievable rate for this UE at the TTI index
R_tminus1	Downlink achievable rate for the UE at the previous TTI index
video_length	Length of requested video in bytes
video_remaining	Remaining video bytes for the UE
video_duration	Total duration of the video (seconds)
user_type	0 for Normal, 1 for Green
sched_start	0 for TTI computations not ready this 1 second cycle, 1 when computations are ready
sched_stop	0 when UE has not received all scheduled Tx subframes this 1 second cycle, 1 when all subframes have been scheduled for Tx
sched_priority_index	Priority index for scheduling the user (lower is higher priority)

Element	*Function*
tti_count	No. of subframes scheduled for Tx for the UE since beginning of 1 second cycle
tti_total	Total no. of subrames the UE should receive this 1 second cycle
handover_occured	Flag: True when handover has recently occurred

which are indicated for the following purposes:

***cycle_sync**:* for synchronizing the internal timers in the MAC schedule timer with the beginning of the mobility position update.
***cycle_slip_guard**:* Number of unreserved subframes at the end of the 1 second cycle to prevent loss of sync between the scheduler's internal 1 second timer and the mobility position update timer when the number of assigned subframes is close to the maximum (1000).
***max_no_of_enbs**:* Maximum number of BSs in the network.

When the simulation starts, the GPM is uninitialized by default. The Remote Radio Controller (RRC) at the BSs adds records in the table after users successfully establish a connection. Figure 2 illustrates the general interaction between the GPM and the BS MAC instances.

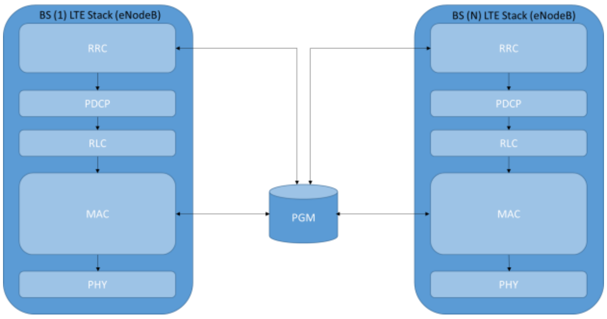

Fig. 2. Interaction between LTE stack layers and PGM

B. Mobility Models

Mobility of the users is based on the Waypoint mobility model. Users remain fixed in their respective positions for almost 1 second and are displaced by a predefined distance at the beginning of the second. In this model, the user 'hops' and remains still for approximately 1 second. This model is used to emulate a fixed path-loss throughout the second such that calculations of resource allocations for each user remain valid throughout the simulation. BS positions are fixed. To implement the hops, the Waypoint model was called twice at the same position at the beginning of each second, and at 1 nanosecond before the end of the second.

C. Green MAC Scheduler

The predictive scheduler in [2] is implemented by modifying the GSOC 2012 Time Domain Maximum Throughput (TDMT) scheduler. The TDMT calculates achievable rates for each UE in the cell and schedules transmission to the UE with highest achievable rate. The scheduled UE receives all available resource blocks for the current TTI index. We started by modifying the scheduler to schedule the UE with highest priority instead of rate at any given TTI (if there were UEs to schedule). The heuristic described in [2] is a Time Domain (TD) approach which divides air-time amongst active UEs according to the following rules:

- **Rule #1**: Deliver the minimum required amount of bits to satisfy UEs classified as '**normal**'
- **Rule #2**: Deliver the maximum amount of bits to UEs classified as '**green**'.

Implementation of the heuristic requires calculating the number of subframes required by each UE at the beginning of each 1 second, and serving the UEs until the end of the 1 second cycle. Since priority of scheduling was not very specific in [2], we elected to use a First-In First-Out (FIFO) approach after classifying users. Air time is given to the highest priority scheduled UE in the 1 second cycle until total number of subframes is reached for this UE is reached, then the next UE in the queue is served, and so on. Normal UEs are always served prior to green UEs. We created an NS-3 module called greenffmacscheduler, which contains the c++ and header files for the Green scheduler. Figure 3 illustrates the main components of the Green scheduler.

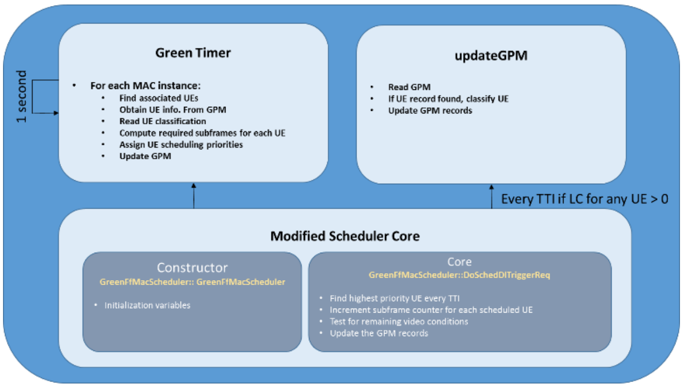

Fig. 3. Interaction between submodules within the Green scheduler

The scheduler is comprised of three main submodules, which are described as follows:

1) **Green Timer**: This submodule is a timer module which executes every 1 second. It is responsible for providing the calculations for the amount of subframes each user receives each second. The number of subframes assigned to each user is calculated according to the classification, according to the following code

In current cell, for N users, for each UE i

- If UE is 'Normal': $$A_{normal,i} = \left\lceil 1.25 \frac{D_i}{R_i} \right\rceil$$
- If UE is 'Green': $$A_{green,i} = \min\left(\left\lceil \frac{L_i}{R_i} \right\rceil, 1000 - \sum_{k=1}^{N} A_{normal,i} - cycle\_slip\_guard \right)$$

where A is the number of allocated subframes, D is the number of bytes per streaming Video segment, R is the achievable rate and L is the remaining video in bytes, and ⌈ ⌉ denotes the ceiling rounding operation

2) **updateGPM**: This submodule is called by the modified core module every TTI, only when a Logical Channel (LC) is assigned, to read the GPM module, reclassify the users and update the GPM module.

3) **The modified core**: This is the main component which calculates the achievable rates for each user, and a user is then selected for Tx scheduling according to set priority.

D. Modified RRC

Figure 4 illustrates the modified RRC layer. The submodule **updateGPM** in this layer is not the same as the MAC scheduler version. This submodule is responsible of updating the GPM on several RRC events, which include establishing a connection between a BS and UE, handover start and end, connection reconfiguration, and other events. Since such events trigger the submodule, changes in UE state, such as moving into a different cell, or association with a new BS will be caught and registered in the GPM early.

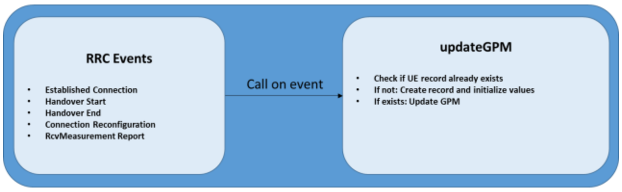

Fig. 4. Functionality of the modified RRC layer

E. Other modifications

Some modifications have been applied to other modules to facilitate the implementation of the MAC scheduler. The following modifications have been conducted:

- We define a variable ***mycellid*** inside the structure ***CschedCellConfigReqParameters*** in ***ff-mac-csched-sap.h***, which is passed to the Green scheduler through the structure **params** in member function **doCschedCellConfigReq()** within the Green scheduler class such to allow each MAC instance to know its own Cell ID.
- A new member function which we called ***PassMyID()*** inside ***lte-enb-mac.cc*** was introduced, which is called from the NS-3 ***main*** function as follows:

mac->PassMyID(cellID);

IV. Simulations

Simulations are conducted to evaluate the performance of the implemented distributed heuristic system. Table 2 shows the NS-3 parameter settings in the project main function. UEs start from the initial coordinates (-500, 50, 0) and traverse in both 5 and 25 meter steps every second to (+500, 50, 0) on the x-axis using the previously described hop behavior. The step size in meters is fixed per simulation run. Bulk send applications are installed on the remote host, and packet sinks at the UEs. The bulk send application attempts to transmit at the highest point-to-point data rate, which was set to 1 GB/s.

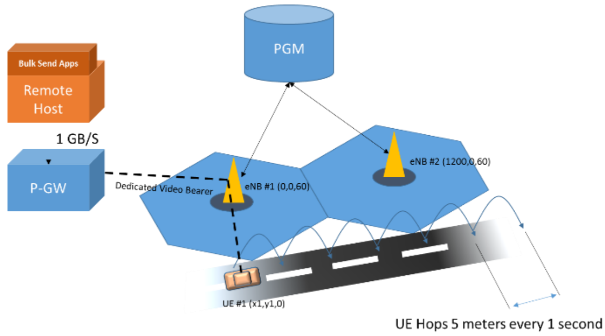

Fig. 5. Functionality of the modified RRC layer

TABLE II. Simulation Settings

Element	***Function***
Path-loss Model	Friis Free Space Propagation Model
BS Tx Power	0 dBm
Antenna type	Isotropic
Antenna Height	60 m
Sectors/Cell	1
Inter Site Distance (Km)	1.2
No. of cells	2
cellid_tminus1	Previous Cell ID of the UE
Bearer type	Dedicated, NGBR_VIDEO_TCP_DEFAULT
TCP congestion control	Tahoe
TCP Socket Rx Buffer	1 GB
TCP Socket Tx Buffer	1 GB
Remote Host Application	Bulk Send
Fading	None
Initial cell association	Attach to Nearest
Handover Algorithm	A2-A4-RSRQ
HARQ	Disabled
Bearer to RLC Mapping	Unacknowledged Mode (UM)
Point-to-Point Data Rate	1 GB/s
Maximum Transmit Unit (MTU)	512

Element	Function
Point-to-Point Channel Delay	10 ms
RRC Type	Ideal
UE speed	Either 5 or 25 meter/s

Applications begin to transmit shortly after 1 second, which ensures data transmission begins after giving the network enough time to establish connections with UEs. Application start times have been randomized time in order to prevent transmission problems in TCP. All transmit and receive buffers are set to a very high length such to eliminate the possibility of buffer overflows at such high data rates. Although we have set RRC attribute ***EpsBearerToRlcMapping*** to ***RLC_AM_ALWAYS***, the ***RLC_UM*** mode is run by NS-3 instead. Thus, performance evaluations will be over UM. The videos requested by the UEs are picked randomly at the beginning of the simulations and thus video attributes vary both in playback rate and duration. However, for the results below, we have fixed the initial seed. Table III shows the list of possible video playback rate based on recommended YouTube data rates in [7]. Video durations range between 4 and 10 seconds.

We simulate several scenarios with different cell loads and compare against the traditional PF scheduler. Table III shows the simulation scenarios. Performance evaluation based on metrics in [2] are used analyze the results presented in the next section. The following evaluation metrics are utilized

1) ***Cumulative allocations****:* This metric calculates the accumulated number of bytes transmitted from the BSs and received at the UE, versus the UE demand rate.
2) ***Air-Time***: This metric calculates the bytes transmitted to and received at the UE every one second for the entire session.
3) ***Instantaneous video stalls***: This metric calculates the difference between the instantaneous demand and fulfillments which indicates video stalls when greater than zero.
4) ***Cumulative video stalls***: This metric calculates the cumulative video stalls.

TABLE III. SIMULATION SCENARIOS

Scenario	*Number of UEs*	*UE Speed*	*Scheduler*
1	1	25	Green Predictive
2	2	25	Green Predictive
3	3	25	Green Predictive
4	1	25	PF
5	1	5	Green

TABLE IV. VIDEO DATA RATES USED

Video Setting	*Data Rate (Bits/s)*
240p	409600
360p	768000
480p	1240000
720p	2560000
1080p	3072000

V. RESULTS

Results for the five scenarios are presented below. Figure 6(a) shows the accumulative allocations for a single UE with constant speed equal to 25 m/s with the distributed heuristic green scheduler. Figure 6(b) shows the air-time for the UE. Figure 6(c) the instantaneous video stalls for this UE, and figure 6(d) shows the cumulative video stalls for this UE.

Figure 7(a) shows the accumulative allocations for the first UE with constant speed equal to 25 m/s with the distributed heuristic green scheduler. Figure 7(b) shows the air-time for the first UE. Figure 7(c) shows the instantaneous video stalls for the first UE, and figure 7(d) shows the cumulative video stalls for the first UE. Figure 7(e) shows the accumulative allocations for the second UE with constant speed equal to 25 m/s with the distributed heuristic green scheduler. Figure 7(f) shows the air-time for the second UE. Figure 7(g) shows the instantaneous video stalls for the second UE, and figure 7(h) shows the cumulative video stalls for the second UE.

Figure 8(a) shows the accumulative allocations for the first UE with constant speed equal to 25 m/s with the distributed heuristic green scheduler. Figure 8(b) shows the air-time for the first UE. Figure 8(c) shows the instantaneous video stalls for the first UE, and figure 8(d) shows the cumulative video stalls for the first UE. Figure 8(e) shows the accumulative allocations for the second UE with constant speed equal to 25 m/s with the distributed heuristic green scheduler. Figure 8(f) shows the air-time for the second UE. Figure 8(g) shows the instantaneous video stalls for the second UE, and figure 8(h) shows the cumulative video stalls for the second UE. Figure 8(i) shows the accumulative allocations for the third UE with constant speed equal to 25 m/s with the distributed heuristic green scheduler. Figure 8(j) shows the air-time for the third UE. Figure 8(k) shows the instantaneous video stalls for the third UE, and figure 8(l) shows the cumulative video stalls for the third UE.

Figure 9(a) shows the accumulative allocations for a single UE with constant speed equal to 25 m/s with the PF scheduler. Figure 9(b) shows the air-time for the UE. Figure 9(c) the instantaneous video stalls for this UE, and figure 9(d) shows the cumulative video stalls for this UE.

Figure 10(a) shows the accumulative allocations for the first UE with constant speed equal to 5 m/s with the distributed heuristic green scheduler. Figure 10(b) shows the air-time for the first UE. Figure 10(c) shows the instantaneous video stalls for the first UE, and figure 10(d) shows the cumulative video stalls for the first UE. Figure 10(e) shows the accumulative allocations for the second UE with constant speed equal to 25 m/s with the PF scheduler. Figure 10(f) shows the air-time for the second UE. Figure 10(g) shows the instantaneous video stalls for the second UE, and figure 10(h) shows the cumulative video stalls for the second UE.

VI. Discussion

From the figures, it can be observed that the implementation of the distributed heuristic approach is in line with the results presented in [2]. For example for the first scenario, in figure 6(a), which is a low load scenario, the single UE, initially 'normal', receives the exact amount demanded per second until declared 'green' at approximately 23 seconds (moving away from the BS), where the UE receives large amounts of data every second. At time 50 seconds, the UE performs a handover, where the UE is declared as normal again. At about 73 seconds, the UE is declared 'green' again. The air-time in figure 6(b) also follows [2] before and after handover. As expected, no video stalls are observed. Interesting effects are seen in the third scenario, where the first UE (which is always ahead but close to the second UE on the highway), is declared 'green' first and receives a priority over the second UE, and continues to receive priority over the second for the entire simulation period as long as they are both declared 'green'. This shows that with FIFO priority queues, together with scheduling strategy of scheduling users down the queue only after finishing prior ones can lead to domination of certain UEs over others during 'green' periods. In scenario 4, it observed from the figures that the PF scheduler, while like the Green Predictive schedule, avoids stalls, its main disadvantage is in the high air-time. An anomaly is seen in scenario 5, where transmission air-time of the distributed heuristic green scheduler is correct, but the UE received bytes are close to zero although all buffers in the simulation were extremely large. This is believed to be due to congestion control.

VII. Future Work

The distributed heuristic scheduler has been evaluated under idealized conditions, such as static channel conditions during the 1-second cycle computations. It was observed that the performance under TCP is degraded when the UE slows down. Thus, modifications to the TCP congestion control algorithms may alleviate degradations. Evaluations of the heuristic under various channel impairments and fluctuations is recommended. Finally, priority queuing can be modified such to ensure no 'green' UE is starved or experiences video stalls, possibly by computing decreases in delivery vs. demand curves from the GPM and serving 'green' UEs in a fair way by adaptively changing priorities.

References

[1] CISCO, "Cisco visual networking index: Global mobile data traffic forecast update, 2012-2017," 2013.

[2] H. Abou-zeid and H. S. Hassanein, "Predictive Green Wireless Access: Exploiting Mobility and Application Information," *IEEE Wireless Communications*, Oct. 2013.

[3] M Elrefaey, MH Shoeb, HH Elgebaly - US Patent App. 13/799,530, 2013.

[4] Google, "Live encoder settings, bitrates and resolutions - YouTube Help", Available: https://support.google.com/youtube/answer/2853702?hl=en (Accessed: 2015, April 29).

[5] Ameigeiras, P., Ramos-Munoz, J. J., Navarro-Ortiz, J. and Lopez-Soler, J.M. (2012), Analysis and modelling of YouTube traffic. Trans Emerging Tel Tech, 23: 360–3

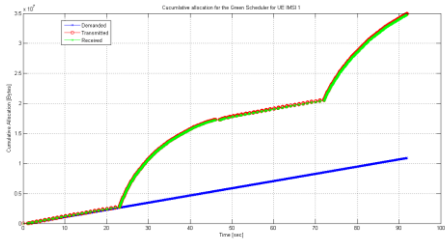

Fig 6(a) Accumulative Tx and Rx allocations for a single UE, moving at 25 m/s vs demand (Green scheduler)

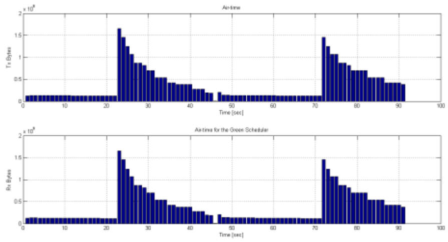

Fig 6(b) Air-time (Green scheduler)

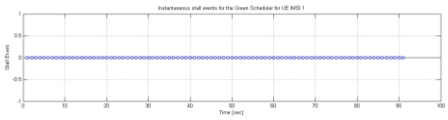

Fig 6(c) Instantaneous video stalls (Green scheduler)

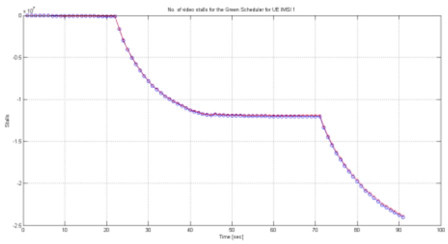

Fig 6(d) Cumulative video stalls (Green scheduler)

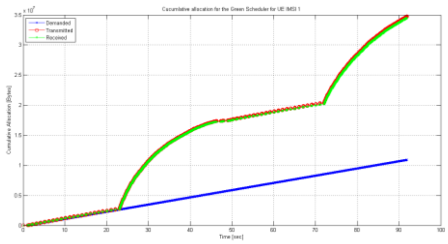

Fig 7(a) Accumulative Tx and Rx allocations for the first UE, moving at 25 m/s vs demand (Green scheduler)

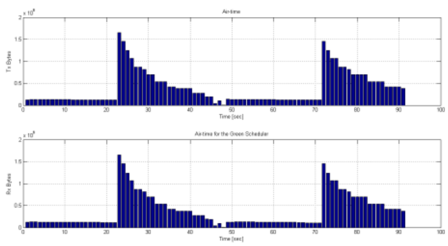

Fig 7(b) Air-time for first UE (Green scheduler)

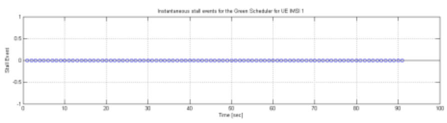

Fig 7(c) Instantaneous video stalls for first UE (Green scheduler)

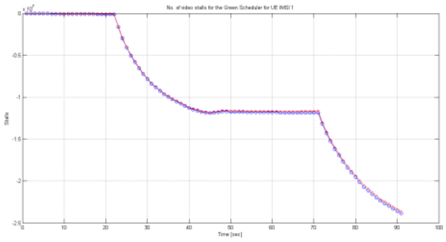

Fig 7(d) Cumulative video stalls (Green scheduler)

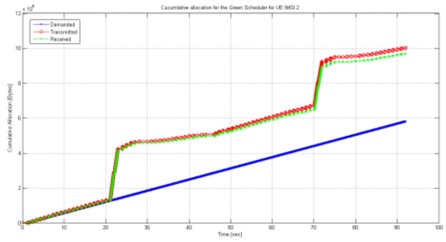

Fig 7(e) Accumulative Tx and Rx allocations for the second UE, moving at 25 m/s vs demand (Green scheduler)

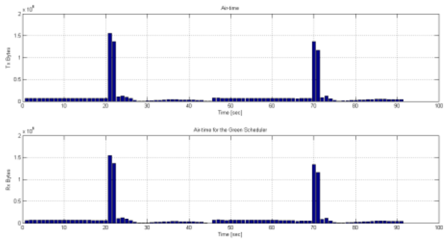

Fig 7(f) Air-time for second UE (Green scheduler)

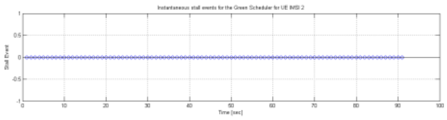

Fig 7(g) Instantaneous video stalls for second UE (Green scheduler)

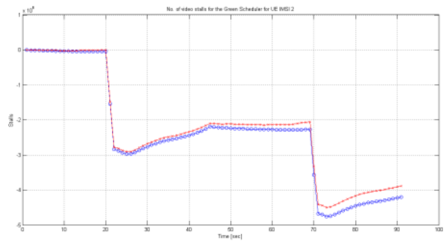

Fig 7(h) Cumulative video stalls (Green scheduler)

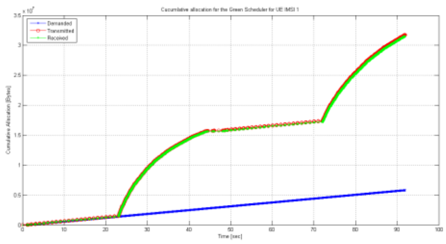

Fig 8(a) Accumulative Tx and Rx allocations for the first UE, moving at 25 m/s vs demand (Green scheduler)

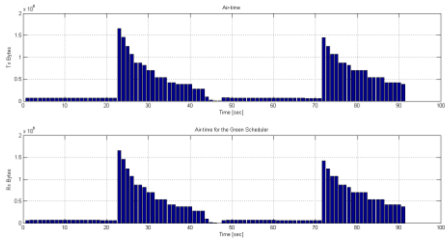

Fig 7(b) Air-time for first UE (Green scheduler)

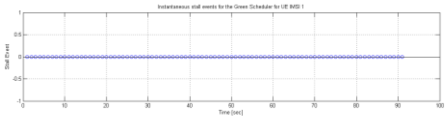

Fig 8(c) Instantaneous video stalls for first UE (Green scheduler)

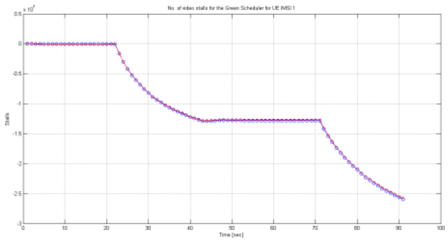

Fig 8(d) Cumulative video stalls (Green scheduler)

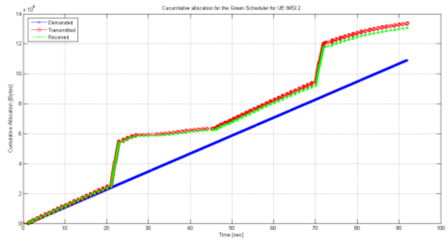

Fig 8(e) Accumulative Tx and Rx allocations for the second UE, moving at 25 m/s vs demand (Green scheduler)

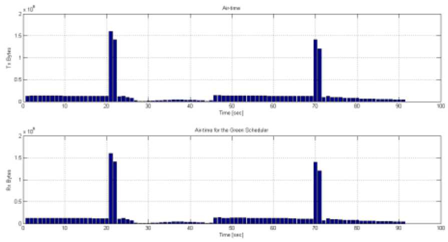

Fig 8(f) Air-time for second UE (Green scheduler)

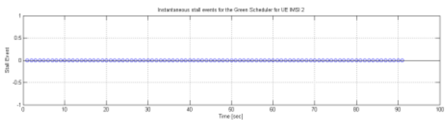

Fig 8(g) Instantaneous video stalls for second UE (Green scheduler)

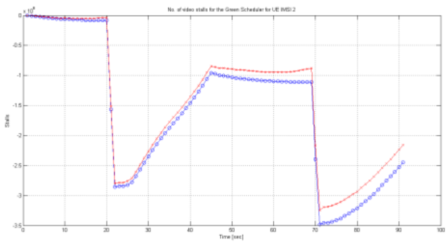

Fig 8(h) Cumulative video stalls (Green scheduler)

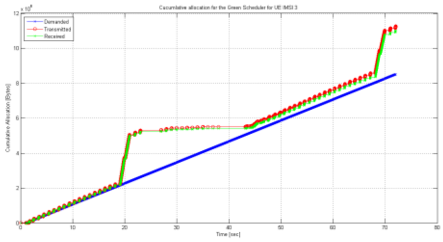

Fig 8(i) Accumulative Tx and Rx allocations for the third UE, moving at 25 m/s vs demand (Green scheduler)

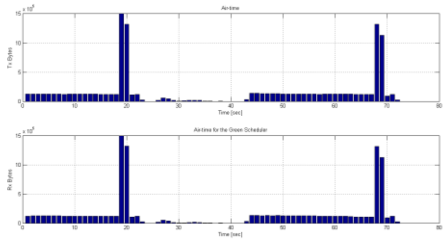

Fig 8(j) Air-time for third UE (Green scheduler)

Fig 8(g) Instantaneous video stalls for third UE (Green scheduler)

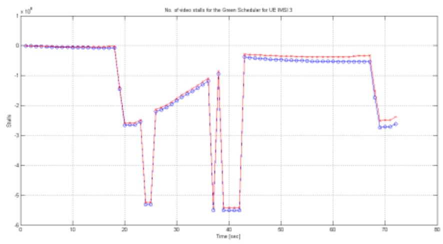

Fig 8(l) Cumulative video stalls (Green scheduler)

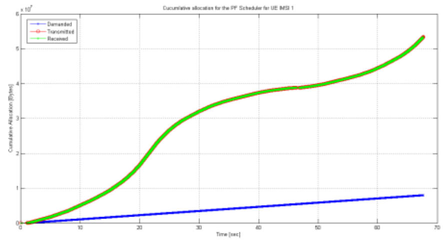

Fig 9(a) Accumulative Tx and Rx allocations for a single UE, moving at 25 m/s vs demand (PF scheduler)

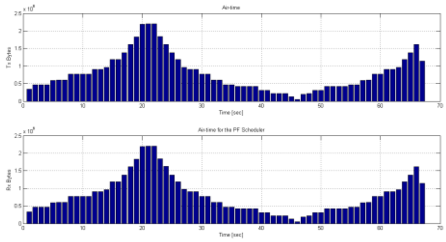

Fig 9(b) Air-time (Green scheduler)

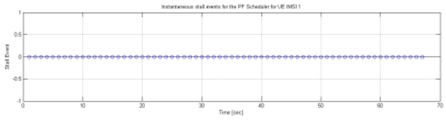

Fig 9(c) Instantaneous video stalls (Green scheduler)

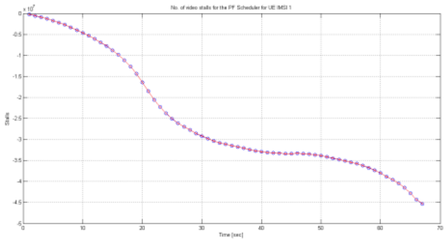

Fig 9(d) Cumulative video stalls (Green scheduler)

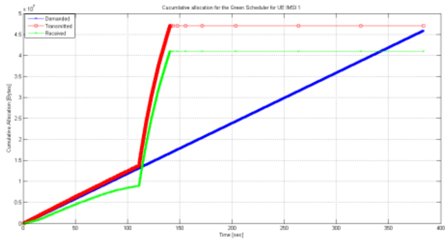

Fig 10(a) Accumulative Tx and Rx allocations, for UE moving at 5 m/s vs demand (Green scheduler)

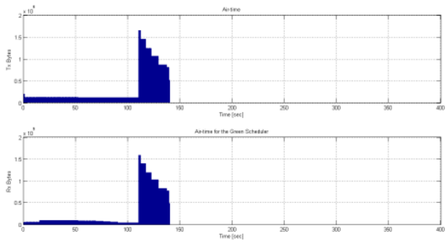

Fig 10(b) Air-time (Green scheduler)

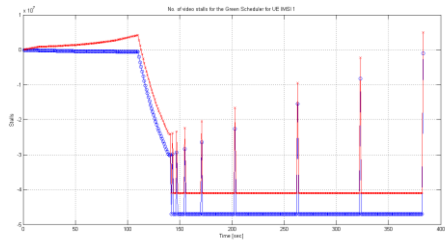

Fig 10(c) Cumulative video stalls (Green scheduler)