Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.18576/amis/13S108

LWDF-DiscMOPSO-EDF Hybrid OFDM Downlink Schedulers for Improved Performance in 4G Networks

T. Sentthil Vinayakam [1,*] *and A. Rajeswari* [2]

Department of Electronics and Communication Engg, Coimbatore Institute of Technology, Coimbatore, India

Received: 12 May 2019, Revised: 27 Jul. 2019, Accepted: 28 Jul. 2019
Published online: 1 Aug. 2019

Abstract: A hybrid scheduling scheme for the downlink of 4G wireless network is proposed herewith and it provides improved and consistent performance across all performance metrics for various overloaded traffic conditions with various heterogeneous traffic mix ratios and intensities. The proposed hybrid algorithm, namely "LWDF-DiscMOPSO-EDF Hybrid (LDEH)" OFDM scheduling scheme is an innovative and improvised combination of the best logic and capabilities of multiple scheduling algorithms i.e. PSO, LWDF and EDF. The PSO-based heuristic scheduler has been improvised much into discrete-multi-objective PSO that is designed for achieving multiple objectives, namely best goodput, reduced starvation, deadline and PLR compliances, all targeted simultaneously. The proposed discrete-PSO overcomes the high spectral inefficiency of a conventional PSO. The well-considered design of the bandwidth-reservation policy in this LDEH scheme enables compliance with deadline & other QoS requirements even under high-intensity traffic conditions. Two versions of the LDEH hybrid scheme are proposed based on two different bandwidth reservation policies. These hybrid schemes' performance is compared with various contemporary algorithms and results prove that both the proposed LDEH schemes outperform existing schemes by producing the best overall and consistent rankings across various metrics and classes of service. The proposed schedulers are also fair, i.e. they do not compromise on performance for one metric or class to improve in another metric or class. Demanding traffic scenarios are used for simulation so that they tested the schedulers in various over-loaded traffic conditions and with varied traffic mix ratios.

Keywords: multimedia, heterogeneous traffic, real-time services, non-real-time services, radio resources, OFDM, 4G, channel-aware, QoS-aware, traffic-aware, multiple objectives, PLR, goodput, packet-delay, starvation, spectral efficiency, PSO, M-LWDF, EDF, hybrid scheduler

1 Introduction

The demand for network capacity in wireless networks continues to expand exponentially due to the increase of mobile penetration and its applications. The convergence of various services and mobile applications have resulted in this exponential growth in bandwidth demand and high network utilization. Evolution of 4G and higher-order networks are based on an integrated IP platform that can deliver multiple services, converged in a single wireless pipe such as voice, audio-conferencing, live-video, recorded audio, file transfer and web applications. The performance of the downlink part of 4G or 5G mobile network is critically dependent on its capability to carry heterogeneous traffic in an efficient manner. The services in 4G networks can be broadly classified into two types, namely real-time and non-real-time services.

IEEE 802.16m and LTE are competing 4G wireless network standards. IEEE 802.16 standard is called as WiMAX which is an acronym for Worldwide Interoperability for Microwave Access. The processes of packet scheduling and frame packing are mostly similar between 802.16m and LTE standards except for few specifications. The data rate of LTE is higher due to higher modulation levels used in its RF Transmitter.

IEEE 802.16m network is considered as the reference in this paper. This network allows five classes of services such as Unsolicited Grant Service (UGS) , Extended Real Time Polling Service (ERTPS), Real-Time Polling Service (RTPS), Non-Real-Time Polling Service (NRTPS) and Best Effort service(BE). UGS, ERTPS and RTPS are real-time services which provide real-time experience to viewers or end customers by means of streaming data. NRTPS and BE are non-real-time services

* Corresponding author e-mail: ts.vinayakam1@gmail.com

which typically transfer stored data or support applications that are not affected by delays of one or two seconds. Typical applications for these classes are:

1. **UGS**: Real-Time: VoIP streaming without silence suppression.
2. **ERTPS**: Real-Time: Silence-suppressed VoIP streaming
3. **RTPS**: Real-Time: MPEG4 in real-time or live video streaming
4. **NRTPS**: Non-Real-Time: File Transfer (FTP)
5. **BE**: Non-Real-Time: HTTP i.e. Browsing

Each of these applications have their own end-to-end delay, packet dimensions , inter-packet delays and other specifications which are specified by various standards, namely 3GPP [12] standards, ITU-T Recommendations[14], CISCO[13] reference guides and WiMAX Forum[15]. These multitudes of standards for end-to-end services are mostly consistent, met in many cases and are non-overlapping. Thus in this paper, references are considered from all these standards.

The downlink part of a 4G or any higher generation wireless network's base station is a single node point that carries high volume and dense mix of heterogeneous traffic. Scheduling for a downlink in high speed networks is a critical real-time process wherein a scheduler resolves contention for radio resources from various users & flows and this determines the order and quantum of radio resource allocation for each of them. Scheduling in such high-speed multi-class systems can be classified as an NP-Hard problem, where the best solution cannot be found in bound time, but a near optimal solution can be found by means of heuristic search algorithms. The downlink schedulers largely determine the ability of the 4G wireless network to meet the QoS requirements of various services carried by it. Akashdeep et al., [3] have well analyzed the classification, characteristics and issues of various scheduling techniques for all WiMAX standards based on their fundamental working principles.

In this paper, OFDM downlink schedulers working on MAC & physical layers and meeting the IEEE 802.16m radio interface specifications are proposed and simulated to schedule heterogeneous traffic. The QoS specifications specified in 802.16m standard [12, 13, 14, 15] for the performance metrics, namely packet delay, goodput, packet-loss ratio and spectral efficiency vary a lot between the five classes of services. Meeting these varied QoS specifications for such a mix of service flows is a complex multi-dimensional task. The scheduling algorithm has to dynamically balance the performance across service flows in dynamic traffic conditions. Typically real-time services have low volume but stringent latency constraints and deadlines, while non-real time services occupy high volumes and have relaxed latency constraints. In dynamic traffic scenarios, if real-time flow volume increases and the scheduler is not dynamic enough in priorities, it will easily result in the starvation of non-real time services over long periods resulting in unfairness and packet losses going beyond limits. If scheduler is optimized more on non-real time service flows, then service quality, packet losses and delays in real-time flows get affected. Effective and dynamic scheduling of such heterogeneous flows with high-traffic-intensity, requires careful design of the scheduler.
Good heterogeneous schedulers should meet the basic principles listed below:

1. Differentiation: by the ability to differentiate between classes of service.
2. Prioritization: by the ability to prioritize certain flows and packets based on its QoS classes whenever there is contention for bandwidth.
3. Fairness: by the ability to avoid starvation of non-prioritized flows in-spite of high load from prioritized services.
4. Opportunistic scheduling: by the ability to utilize good channel conditions at appropriate time to improve throughput and other parameters.
5. Adaptive scheduling: by the ability to adapt based on QoS requirements and load conditions.

All contemporary schedulers of heterogeneous traffic have differentiation property. All Hierarchical schedulers adopt the prioritization principle since they first prioritize classes with least deadlines. Channel-aware schedulers have cross-layer approach and are opportunistic in nature, since they allocate radio resources demanded by users/flows with good RF channel conditions. Good RF Channels have high MCS value i.e. Modulation Control Scheme value. Thus channel-aware schedulers use physical layer information on channel quality to do MAC level scheduling. Load- or traffic-aware schedulers are able to meet fairness principle, since they analyze the load offered by each flow. In [26] Niyato et al., used an adaptive queue-aware algorithm for uplink bandwidth allocation and rate-control mechanisms for polling services in mobile stations. QoS-aware schedulers are able to prioritize flows that have stringent deadlines and priorities to be met. The QoS-aware schedulers also have cross-layer approach, since the performance requirements are higher layer information. J. Lu et al., [24] have studied the effect of cross-layer information in drastic improvement in channel-aware and QoS-aware schedulers' abilities to meet performance requirements. Scheduler schemes [27] that are based on call admission and flow control, resolve contention from a lower class connection in case of traffic congestion and allow that call only if it does not overload the network. These are constraint-based schemes that require co-ordinated functioning of the Base-station node as well as the gateway or application layer function that has control over call admission.

The "LWDF-DiscMOPSO-EDF Hybrid (LDEH)" OFDM scheduling scheme proposed in this paper is designed to meet all the five principles from differentiation to adaptive scheduling. LDEH scheduler does not require call admission or flow control features. The proposed LDEH scheduler's performance proves that it is comprehensively superior under high-intensity traffic when compared with recent channel-aware, QoS-aware and traffic-aware schedulers, namely LOG, EXP-W and M-LWDF, which are referred in recent papers [1,2].

The results in this paper are analyzed in two traffic conditions, namely low RT/NRT ratio (< 1.0) and high RT/NRT Ratio (> 1.0). This paper proves that the proposed first version of the hybrid scheduler, i.e. LDEH-1 (HYB-1) gives the best results in all Real-Time (RT) metrics, namely RT-goodput, RT-delay, RT-starvation and RT-PLR under various highly-overloaded traffic conditions. For Non-Real-Time (NRT) metrics such as NRT-goodput and NRT-SI as well as the overall spectral efficiency metric, HYB-1 gives top or close to top performance particularly under low RT/NRT traffic. The NRT-packet delay metric is the only metric in which HYB-1 results are not competitive. But it should be noted that HYB-1 is able to provide zero NRT-starvation for low RT/NRT in spite of higher delays.

The proposed second version, namely LDEH-2 (HYB-2) results in the top rank performance in NRT-goodput, NRT-starvation-index and overall spectral efficiency among all traffic conditions. HYB-2 also performs close to the top rank in all other metrics, namely RT-goodput, RT-delay, NRT-delay, and RT-SI. HYB-2 is also the most consistent scheduler that comes in top rank or close to that in all metrics and classes even under highly-overloaded traffic. This is one quality that is set out as objective and is achieved in this research work.

The results prove that the design of HYB-2 is robust enough such that even under overloaded traffic, it controls packet delays within deadline limits for both high- and low-priority services such that the packets do not get dropped as per HARQ specifications of IEEE802.16m, thus meeting the critical end-to-end delivery requirements. This quality is not observed in other competing schedulers such as LOG which compromises on critical QoS requirements such as RT-starvation and RT-Packet Loss Ratio (PLR), but LOG is able to produce very good results in other metrics & class such as NRT-packet delay. Starvation in real-time streaming services affects end-to-end QoS severely. RT-starvation index is one of the parameters in which both the proposed LDEH schemes produce outstanding results. Other competing recently referenced scheduler, namely M-LWDF, performs relatively poorer in delay metrics and thus ranks relatively lower. EXP-W ranks the lowest due to its exponential objective function that is sensitive to load. The testing methods and traffic scenarios adopted in this work brings out the comprehensive behavior of these competing schedulers.

Thus this paper shows that even under highly-overloaded traffic, the proposed LDEH scheduler ensures good compliance in all QoS requirements, without compromising on one class or metric in order to improve results for another class or metric, which dynamically schedules highly-overloaded heterogeneous traffic and thus provides good results. These qualities are primarily achieved by the well-considered overall design of the hybrid, discrete-multi-objective PSO and the dynamic bandwidth reservation schemes proposed in this paper.

Even when compared with contemporary advanced hybrid schedulers that use heuristic algorithms, namely FLS [1],[7] & HWEL-MT [2] schedulers, the proposed LDEH scheduler is superior in handling high-traffic intensities up to 1.6, while FLS and HWEL-MT are tested for an estimated traffic intensity up to 0.9 only.

This paper is organized as follows: Section 2 overviews the complete OFDM Scheduler, Section 3 explains existing flow scheduling schemes, Section 4 explains the design of the LWDF-DiscMOPSO-EDF HYBRID (LDEH) scheduler, Section 5 proposes the frame packing algorithm, Section 6 explains simulation specifications, Section 7 discusses the obtained results and Section 8 concludes the main aspects of the paper.

2 OFDM SCHEDULER - AN OVERVIEW

The overall system structure for a two-dimensional OFDM downlink scheduler is shown in the block diagram Fig (1). It consists of packet / flow scheduler, OFDM frame packing algorithm and the two-dimensional OFDM frame structure in 802.16m network. The first stage is the flow (packet) scheduler or MAC scheduler that takes packet flows as inputs from connection queues of various classes of service. Then it schedules and enlists packets into a pre-pack queue. This pre-pack queue in turn forms the basis for resource allocation by the frame packing algorithm to form the OFDM downlink frame that can be transmitted. The frame packing algorithm proposed herewith is a modified-OBBP (Orientation-Based Burst Packing) algorithm. The OBBP algorithm [10] processes the entire pre-pack queue in a comprehensive manner involving an exhaustive way of classifying packets based on the rectangular burst sizes and then decides on the exact location, size & orientation of resource allocation in the current OFDM frame for each burst. A relative merit of OBBP is that it takes less processing time compared to other frame packing algorithms. A sample frame packed with bursts of various sizes and orientations is illustrated in Fig(1). In this paper, modified-OBBP is proposed to achieve high spectral efficiency of 0.9 bps/Hz. The section titled *Frame Packing Algorithm* may be referred for details about the proposed modified-OBBP.

OFDM frame structure in 802.16m: As per IEEE 802.16m radio interface specifications [11,16] and the overall system illustrated in Fig (1), a typical OFDM frame's dimensions of 10 MHz channel bandwidth as height and 5 ms frame duration as width is considered in this paper. The dimension of the vertical frequency axis is of 10 MHz bandwidth and it is made up of 1024

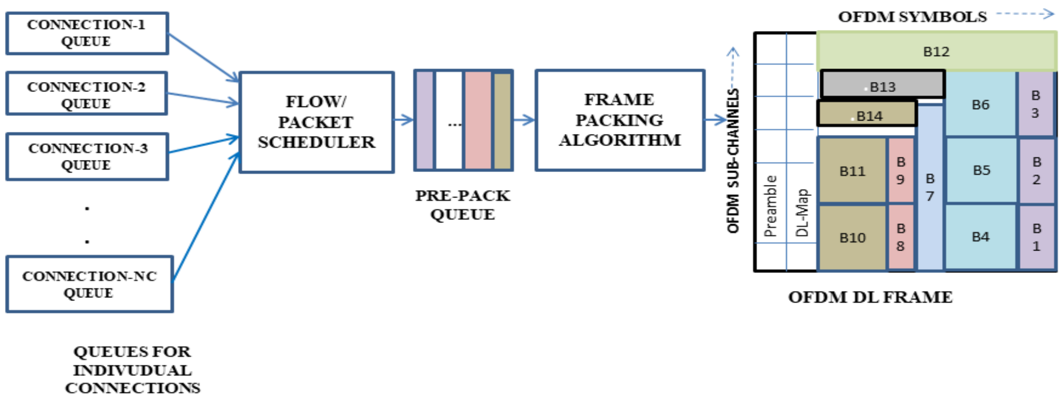


Fig. 1: Overall block diagram for packet scheduler, OFDM frame packer and OFDM frame structure in 802.16m

sub-carriers. Due to frequency margins and overheads, the usable RF bandwidth is only 840 sub-carriers consisting of 30 sub-channels of 28 data sub-carriers each.

The horizontal dimension of 5 ms in time axis is broken into 48 symbol duration. The smallest resource block is a slot which is a rectangular radio block of 28 sub-carriers in the frequency axis and two-symbol duration in the time axis. In time axis, each frame is made of 48 symbol duration and it consists of 4 sub-frames occupying 12 symbols each. Frame overheads such as Preamble and Downlink-Map (DL-Map) typically occupy first two to four slot-columns of the frame depending on whether it is a normal frame or a super-frame. In a normal-frame, first two slot-column widths are frame overheads resulting in an usable resource of 30x22 = 660 slots per frame. One $Super-frame$ occurs for every four frames and additional 2-slots wide column is utilized for overheads resulting in an usable resource of 30x20 = 600 slots per super-frame.

The proposed scheduler is configured for downlink scheduling in the FDD systems occupying 10 MHz spectrum for downlink with a frame duration of 5 ms each. Downlink is considered in this paper, since only downlink schedulers face the complex task of scheduling packets among multi-class and multi-flow conditions.

The flow scheduler and OFDM burst constructor are very much dependent on each other's specifications and performances. Both of them together decide the performance of the entire downlink scheduler. Thus this paper proposes a comprehensive system that takes in queues of multi-class packets and produces two-dimensional OFDM frames that can be fed to RF transmitter.

3 EXISTING SCHEDULING SCHEMES

Various scheduling algorithms are proposed and used for downlink in 4G wireless networks. Recently-developed conventional algorithms that provide competent performances are LOG, EXP-W and M-LWDF [1,2], and all these three are channel-aware, traffic-aware and QoS-aware schedulers which are desirable characteristics. They perform better than all other conventional algorithms even in high-traffic conditions. Thus LOG, EXP-W and M-LWDF are considered for reference and comparison in this paper. Conventional scheduling algorithms are less-process intensive with low-execution time, but they are sub-optimal in high-traffic conditions as proved in this paper. LOG, MLWDF and EXP-W algorithms compute and evaluate their respective objective function values for the HOL packets of all connections, based on various parameters as stated in below equations (1), (2) & (3). Then in each iteration, these schedulers choose a HOL packet that has the best function value (in which the best can be a maxima or minima) and enlists it. This iterative process is repeated until all the queues are exhausted or enlistment attains frame capacity.

3.1 LOG

LOG was proposed by Sadiq et al., [18]. It is an opportunistic algorithm that optimizes delay and is a load-aware scheduler. Its objective function is given by equation (1)

$$F_{max} = max[b * \log(c + a_i Q_i) * K_i] \quad (1)$$

Where a_i = priority number of COS of the i^{th} connection. It takes one of the values of [20,20,15,2,1] for the 5 classes from UGS to BE in that order.
Q_i=Queue length in terms of slots-required for allocation of all packets for the i^{th} connection.

K_i = MCS value which represents the RF channel quality of i^{th} connection.
b constant is set at 1, and c constant set as Zero.

3.2 M-LWDF

M-LWDF was proposed by Andrews et al., [20] and later recently analyzed by Afroz et al., [1]. M-LWDF is a throughput-optimal channel-aware algorithm suited for overloaded traffic in a shared network, a scenario very similar to the scenarios considered in this paper. Simulation in this paper has also shown that M-LWDF is adaptive to traffic conditions, good at throughput and fairness metrics. Thus M-LWDF is incorporated as the first stage of the proposed LDEH scheme and is also used as reference. Its objective function is given in equation (2).

$$F_{max} = max[a_i * W_i * K_i] \tag{2}$$

Where W_i =Waiting time of HOL packet in the i^{th} connection,
a_i is the class priority number as described previously and K_i= MCS value representing its RF channel quality.

3.3 EXP

EXP-W and EXP-Q are two versions of exponential algorithms proposed by Shakkottai et al., [19]. EXP-Q uses the queue lengths into the objective function formula, while EXP-W uses the waiting time in it as in equation (3). Since EXP-W is found relatively better than EXP-Q in performance, EXP-W is used in this paper for reference and comparison. The objective function formula for EXP-W is equation (3):

$$F_{max} = max[\mu_i * \exp\frac{a_i * W_i}{E(W)^{\eta}}] \tag{3}$$

Where W_i =Waiting time of HOL packet of the i^{th} connection. i=1..N.
$E(W)$ = Average waiting time for all HOL packets in N connection.
μ_i = Number of packets of the i^{th} connection that can be scheduled if all available un-allotted radio resources of the current frame are used for servicing the queue of the i^{th} connection.
η is a constant set as 1. a_i is the priority number for the COS and it is similar to one used in LOG.

3.4 EDF

Earliest Deadline First (EDF) is one of the conventional schedulers used in various time constrained multi-flow applications and is based on the objective function (4).

$$F_{min} = min[t_i + dl_i] \tag{4}$$

Where t_i =time of entry of HOL packet in the i^{th} queue, dl_i is the deadline for its COS
Effectively, EDF chooses the HOL packet with the lowest function value i.e. nearest deadline time among the HOL packets of all the queues.
Sherry et al., [21] concludes that EDF is relatively efficient than WFQ and WRR in meeting deadlines.

3.5 Heuristic Algorithms

Heuristic algorithms iteratively search for optimal results by improving search in each iteration. They provide better results compared to conventional algorithms but require high processing resources and time. Various heuristic schedulers have been proposed based on various natural phenomena as well as theoretical approaches. Singh et al.,[23] have proposed linear programming-based and heuristic-based schedulers to achieve fairness for heterogeneous traffic in wireless downlink. But the objective and analysis in that research work [23] was restricted only to fairness and data loss metrics. Among the various heuristic methods proposed earlier, Particle Swarm optimization (PSO) [6,5,17] is a heuristic scheduling algorithm with a good amount of flexibility and fits effectively very well for the requirement of multi-objective function. PSO has been proposed earlier for usage in wireless networks [17]. Hence PSO is adopted and integrated into the proposed scheduler in this paper.

3.6 Particle Swarm Optimisation

Particle Swarm Optimization is a Genetic optimization algorithm inspired from nature and it was originally proposed by Kennedy & Eberhart [29]. PSO is a heuristic search algorithm which is able to overcome local minima in a much better way compared to conventional algorithms. The PSO algorithm is inspired from the natural phenomenon of birds flocking towards the best prey position using personal knowledge and the swarm's collective intelligence. The following analogy is used herewith for applying PSO in scheduling problems. The flying birds are considered as particles which are essentially candidate schedules. Multiple particles i.e. candidate solutions are created at random but subjected to a fitness criteria. The prey is basically the goal of the entire search operation, which is the optimal schedule that is determined based on the best function value. In this paper, the lowest total penalty is considered as the best function value, and this is described in detail under the section titled *Design of the LDEH Scheduler*. When birds or particles fly, the candidate schedules move in the solution space and get updated iteratively by moving in a direction and jump-size decided by the velocity equation. The jump-size indicates the magnitude of shift in every solution that occurs during the iterative position update process. Velocity updates are based on the knowledge of the best-known positions learned from the past self-knowledge and social-knowledge. Learning from each particle's self-knowledge happens when it evaluates and compares its position in the current iteration with the best position observed during its own past flight path and updates the *pbest* i.e. its own best position. Similarly, social knowledge is acquired by comparing all particles' best positions, choosing the best position from all particles' flight-paths and updating it as *gbest* i.e. the global-best position. The particles move towards a promising area and converge upon the optimum resource allocation schedule at the end of all iterations in each frame period. PSO iterations are stopped after a fixed number of iterations in this wireless downlink scheduling problem since it is time-constrained by frame duration. Thus PSO can be considered as a machine learning algorithm since it learns from its own past experiences within each frame duration.

PSO has evolved over many developments and finds other applications such as Cloud/Grid computing, timetable, flow shop and job-shop scheduling. Pandey et al.,[22], optimized scheduling of workflow in the cloud computing environments using PSO.

Multi-objective PSO has proven its efficacy in meeting multiple diverse objectives while searching for an optimal solution in a wide multi-dimensional space. Shubham et al.,[5] have proposed multi-objective PSO for optimization of workflow scheduling using work-span and resource utilization in cloud computing application and achieved 10% increase in capacity utilization. Annauth et al., [17] used multi-objective PSO in OFDM for optimization of sub-carrier versus power combination to improve upon throughput and BER.

Discrete PSO is defined and understood in different ways. One definition of *discrete* is to use integer values instead of continuous variable values for solutions and these problems can be solved by combinatorial methods. Rosendo et al., [28] have proposed the use of discrete-PSO with a set-based combinatorial approach for solving the traveling salesman problem. Discrete PSO algorithms were also proposed in [8,9] for application areas such as job-shop problems.

3.7 Hybrid Algorithms

Recent research studies are oriented towards hybrid schemes that combine heuristic and/or conventional schemes and provide better performance. The hybrid HWEL-MT algorithm was proposed by Asadollahi et al., [2] using Exponential, Logarithm and Maximum-Throughput algorithms. Asadollahi et al., have simulated for traffic up to 70 users (corresponding to an estimated traffic intensity of up to 0.9), and HWEL-MT was able to outperform LOG, E2M and EXP-W algorithms, particularly in real-time classes such as VoIP and video-streaming. But it is noted that the margin of improvement provided by HWEL-MT is low in packet loss ratio and non-real-time delay metrics. Only in half of the parameters studied, particularly for video & BE throughputs and video PLR, HWEL-MT has shown improvement when compared to its next best competitor i.e. E2M.

Mustafa et al.,[4] have proposed hybrid of Fuzzy logic and PSO with an objective of optimizing power allocation and total power usage. Hybrid among conventional algorithms have been proposed by Settembre et al., [25] in which different queuing principles are applied for different traffic classes. For example: fixed bandwidth scheme for UGS class, WRR scheme for RTPS class & NRTPS classes and RR scheme for BE class. Y.Yi et al., [6] used hybrid of GA and PSO algorithms to optimize OFDMA power and frequency in different network scenarios such as unicast and multicast.

Frame Level Scheduler (FLS) was originally proposed by Piro et al.,[7] as a two-level scheduler for LTE downlink. At the top layer, FLS uses closed-loop control designed for a given delay constraint. The upper layer decides the priority parameter that is used for scheduling by the lower layer which is essentially a Proportionately-Fair scheduler. The design of FLS is optimized for bounded packet delay and analysis in [7] is focused on the relationship between delay versus PLR and other performance metrics. FLS is designed to maintain constant delay across various loads, provided the load is medium and within certain bounds.

Afroz et al.,[1] have compared LOG, EXP-W, M-LWDF and FLS algorithms for heterogeneous traffic mix in a fixed mix ratio of 40% : 40% : 20% of Video, VoIP and BE users respectively. Performance metrics are analyzed up to 50 users which provide an estimated traffic intensity of 0.67. Results in [1] have shown that FLS maintains superior throughput for RT traffic in comparison to LOG and EXP-W schedulers, but at the expense of reduced throughput for NRT traffic. FLS results in 1% PLR for VoIP flow even for a traffic intensity of 0.67 (i.e. 50 users). For 50 user traffic, average packet delay for video flow is 55 ms for FLS and $>=$ 58 ms for LOG and other algorithms as stated in [1]. For VoIP traffic, the FLS is able to maintain almost flat average-packet delay of approximately 12 ms up to 50 users, while for all other schedulers, the delays are linearly increasing with the number of users. In fact, FLS is designed with an objective of maintaining this constant delay over given bounds of traffic volume. But it is noted that NRT-Throughput is inferior for FLS when compared to other schedulers for all volumes of traffic. Further [1] analyzed results for a fixed heterogeneous mix ratio.

4 DESIGN OF THE LWDF-DiscMOPSO-EDF HYBRID (LDEH) SCHEDULER

4.1 Description of the Proposed Scheme

In this paper, a hybrid packet scheduling scheme LWDF-DiscMOPSO-EDF HYBRID (LDEH) scheduler is proposed herewith by an innovative and improvised combination of three algorithms, namely M-LWDF, a Discretised version of Multi-Objective PSO (DiscMOPSO) and EDF. A brief introduction about the ingredient algorithms, i.e. discrete-PSO, multi-objective-PSO, M-LWDF and EDF algorithms have been provided in the above literature survey.

The version of discrete-PSO proposed in this paper, processes solution values in integers and allocates full packets without fragmenting them. This results in complete elimination of overhead-addition and the resultant inefficient resource utilization which are basically caused by packet fragmentation. Whereas the conventional PSO algorithm breaks or fragments packets repeatedly over iterations, resulting in excessive overheads which gets aggravated in the OFDM systems due to addition of redundant buffer-slots to form rectangular bursts. In one scenario, the conventional PSO's packet overheads reach 140% of MLWDF. The discrete PSO not only eliminates this excessive overheads, but also reduces it below that of MLWDF.

The version of Multi-Objective PSO (MOPSO) proposed in this paper is able to meet diverse objectives of heterogeneous traffic simultaneously such as maximization of throughput, channel-aware opportunistic scheduling, class-wise QoS deadline compliance, avoidance of starvation for low-priority-packets and achievement of low packet loss even for high-traffic intensity. MOPSO is sensitive to objective function's co-efficient values and threshold settings which are needed to be set optimally. In this paper, a well-designed Hybrid is proposed herewith using Disc-MOPSO and two other conventional high performance algorithms, namely M-LWDF & EDF. The proposed LDEH scheme is channel-aware, load-aware as well as QoS-aware scheme and is dynamic in these three aspects. It is channel-aware since it considers RF-channel quality (MCS index) and opportunistically allocates resources for good channels. It is QoS-aware since it considers QOS requirements such as deadlines before allocating resources. It is dynamic & load-aware since it is adaptive to the queue and the load.

The proposed LDEH scheduler algorithm dynamically classifies packets into Real-Time (RT) and Non-Real Time (NRT). UGS, ERTPS and RPTS classes are classified as RT and NRTPS and BE classes are classified as NRT. NRT packets are further classified into prioritized and non-prioritized packets. Prioritized NRT packets are from connections with very good RF channel quality or whose deadlines are very close. Prioritization process is done as part of the bandwidth-reservation round which is prior to the LWDF round. Two schemes for bandwidth reservation, namely LDEH-1 (or HYB-1) and LDEH-2 (or HYB-2) are proposed herewith. These HYB-1 and HYB-2 schemes vary in few aspects between themselves namely the thresholds and reservation limits as explained in sub-section on the proposed bandwidth reservation policy.

4.2 Overall process for each frame in LDEH Scheduler

Initializing packets as per 802.16m standards:

The packets in input queues and their respective channel conditions are processed initially to subject them to various specifications of 802.16m,[11,16] as follows:

- Size check: i.e. if packet size is $>$ 600 bytes then the packet is fragmented into smaller ones of $<=$ 600 bytes each. Packet sizes in slots are rounded up by buffering to the nearest higher standard burst size .
- Overheads are added to the new fragmented packets created. Each packet gets 8 bytes overhead, i.e 6 bytes header size plus 2 for CRC
- HARQ is done as per specs. It is checked whether the packet's waiting time in queue $>$ deadline for that class. If so, re-insert the packet at the end of the queue. If any such delayed packet has been re-inserted four cycles already, then drop the packet.
- Apply broadband fading by varying the RF channel quality (MCS) index randomly by +/- 2 points max in every 10 ms, i.e. every two frames. Since it is a broadband fading, a particular mobile's frequency hopping does not affect its fading. But each connection has its own sequence of fading that depends on its mobility.

Further a check for $No-Contention$ is done, in which if the total radio resources required to accommodate the entire queue load is less or equal to the current OFDM frame's capacity, then the entire scheduling process is bypassed, all queues are shifted to the pre-pack list and the process proceeds for frame packing.

Flow scheduling-cum-OFDM frame packing:

The overall scheme of the proposed LDEH scheduler is listed in the following steps:

1. Perform bandwidth reservation for LWDF round for RT & all prioritized NRT packets. This is done to ensure real-time and prioritized-non-real-time packets are allotted bandwidth in the LWDF round itself.
2. Flow or packet scheduling is done in three stages and in sequence as follows:
   (a) Perform LWDF (Modified-LWDF) round of packet scheduling
   (b) Perform DiscMOPSO round of packet scheduling.

(c) Perform EDF round of packet scheduling. This completes flow scheduling and a pre-pack list of packets is generated as output of flow scheduler.

3. Perform OFDM frame packing process using modified-OBBP algorithm after taking in the above pre-pack list.

Note: The flow scheduling may terminate anywhere in-between these stages, if either the input queues have been emptied or the pre-pack queue is filled.

4.3 Bandwidth reservation policy proposed in LDEH HYB-1 and HYB-2 schemes

1. HYB-2 scheme estimates *estimated maxima for NRT* for LWDF round using (5) i.e. max estimated slots to be reserved for NRT in LWDF round

$$Est.Max.NRT = \frac{fslots * \Sigma(Est.Q.NRT)}{2 * \Sigma(Est.Q.All)} \quad (5)$$

Where $Est.Max.NRT$ = Estimated maxima for NRT
$fslots$ = the frame capacity.
$Est.Q.NRT$ = Estimated slots required for all HOL NRT Pkts which is equal to number of NRT connections x 45 slots each.
$Est.Q.All$ = Estimated slots requirement for RT Plus NRT.
45 slots is the average radio resource required for a 600-byte sized packet for an average RF channel quality. Similarly for an average RT-pkt size of 42 bytes, 4 slots are required.

2. HYB-2 scheme estimates using (6) the max estimated reservation for RT in LWDF as:

$$Est.Max.RT = \frac{fslots * \Sigma(Est.Q.RT)}{\Sigma(Est.Q.All)} \quad (6)$$

Where
$Est.Max.RT$ = Estimated Maxima for RT
$Est.Q.RT$ = Estimated slots required for HOL RT packets only. Computed as 4 slots per RT connection.

Note : HYB-1 scheme does not require above steps 1 & 2 since it uses only the actual load data and not the estimates.

3. The following formula (7) is used for both HYB-1 & HYB-2 to compute the actual maxima allocation required for all RT packets in LWDF round:

$$Act.Max.RT = \Sigma(All.RT) \quad (7)$$

Where $Act.Max.RT$ = Actual Maxima for RT
$All.RT$ = Slots required for all RT Pkts in queues

4. The following formula (8) is used for both HYB-1 & HYB-2 to compute the actual maxima required for all prioritized-NRT-pkts:

$$Act.Max.NRT = \Sigma(Priority.HOL.NRT) \quad (8)$$

Where $Act.Max.NRT$ = Actual Maxima for NRT
$Priority.HOL.NRT$ = slot requirements for each of HOL NRT pkts classified as $Prioritized$.
The prioritized NRT packets are chosen using two different sets of criteria for the two Hybrids as proposed below :

For HYB-1 scheme:
Choose HOL-NRT pkts that meet either one of the following
(a)RF-MCS $>$ 13 *OR*
(b)Packet delay $>$ 33% of deadline.
For HYB-2 scheme:
Choose NRT pkts that meet either one of the following
(a)All pkts in NRT queues having RF-MCS $>$ 12 *OR*
(b)The HOL pkt for those NRT connections having Pkt delay $>$ 10% of deadline
Note: The best possible MCS value is 15. MCS threshold values of 12 & 13 as stated above, indicates good channel quality. Thus opportunistic scheduling is used here to prioritize and push through NRT packet if channel quality is good. The above threshold limits ensure good load-adaptive balance between RT and NRT.
Below steps 5, 6 & 7 compute and fix the final resource reservation limits, namely $RT.Max.Limit$, $NRT.Max.Limit$ and $All.Max.Limit$ that will be finally applied in reservation and scheduling processes. These limits are fixed based on the frame capacity, the estimated and actual maximal slot requirements as computed in previous steps. The steps 5 & 6 below form part of the 6 *Frame RT NRT Cycle* that is explained subsequently.

5. For every sixth frame, NRT is given full preference over RT in resource reservation and $NRT.Max.Limit$ is fixed first on priority using the following process.
For HYB-1: $NRT.Max.Limit$ = Minima between ($Act.Max.NRT$ and $Fr.Cap$).
where $Fr.Cap$ = Frame capacity in slots.
For HYB-2: $NRT.Max.Limit$ = Minima between ($Act.Max.NRT$, $Est.Max.NRT$ and $Fr.Cap$).
$RT - maxima$ is fixed secondarily based on balance available slots after fixing NRT limit.

6. For other frames, i.e. every 1^{st} to 5^{th} frame, the $RT.Max.Limit$ is fixed first and then $NRT.Max.Limit$ gets reserved slots if available. Process is as follows:
For HYB-1: $RT.Max.Limit$ = Minima between ($Act.Max.RT$ and $Fr.Cap$)
For HYB-2: $RT.Max.Limit$ = Minima between ($Act.Max.RT$, $Est.Max.RT$ and $Fr.Cap$).

7. Limit the total of RT & NRT = $All.Max.Limit$ by means of equation $All.Max.Limit$ = Minima between ($Fr.Cap$) and (Sum of $RT.Max.Limit$ and $NRT.Max.Limit$)

After the completion of bandwidth reservation round and deriving dynamic reservation limits, namely $RT.Max.Limit$, $NRT.Max.Limit$ and $All.Max.Limit$ as described above, these limit values are passed on to the next round i.e. MLWDF round.

4.4 Process for MLWDF round

1. Enable only those NRT connection queues that are prioritized in BW reservation stage.
2. Enable all non-empty RT queues.
3. Compute M-LWDF function value for each HOL packet from all enabled queues, and choose the packet with the highest value.
4. Based on the frame number & the packet chosen, perform steps in LWDF round to complete the $6\ Frame\ RT\ NRT\ cycle$ as follows:
   (a) If the frame number is not a multiple of 6 and RT packet is chosen, then enlist the RT packet (i.e. fill in pre-pack queue) and subject the filled RT+NRT slots to the limit of $All.Max.Limit$.
   (b) If the frame number is not a multiple of 6 and NRT packet is chosen, then enlist it and subject the filled- NRT slots to the limit of $NRT.Max.Limit$.
   (c) If the frame number is multiple of 6 and RT packet is chosen, then enlist it and subject the filled RT to the limit of $RT.Max.Limit$.
   (d) If the frame number is multiple of 6 and NRT packet is chosen, then enlist it and subject the filled RT+NRT slots to the limit of $All.Max.Limit$.
   (e) Irrespective of any condition above, exit MLWDF round if total enlistment reaches $All.Max.Limit$

SIX-FRAME-RT-NRT-CYCLE

It is observed from an analysis, that the cyclic RT-NRT prioritization improves the results considerably. The RT-NRT prioritization cycle-period is fixed as 6-frames based on the ratio between deadlines of RT and NRT classes. The stringent RT class deadline is 60 ms and the delay threshold considered for NRT prioritization is 330 ms (33% of 1 sec) which forms a ratio of 1:6 approximately. Further this 6-frame cycle period is found to produce optimal results based on an analysis in comparison with various other frame cycles periods.

The six frame cycle that alternatively prioritizes RT & NRT is proposed for both, BW reservation round and LWDF round, thus making it effective. Every sixth frame, BW reservation round computes the limit for dynamic-load-based priority for NRT and MLWDF round that effectively implements this upper limit in the sixth frame, thus completing the cycle. In other five frames, the position of RT and NRT gets swapped, giving RT a dynamic-load-based priority. This six-frame cycle gives freedom and variation in every sixth frame to the LWDF as well as DiscMOPSO rounds to schedule large un-fragmented NRT packets (either prioritized or not), which otherwise would have been starved in heavy RT traffic and due to their large sizes. Further explanations on the limits computed by the BW reservation round is given below:

1. In LWDF round, in every 6^{th} frame where NRT is prioritized by policy, if LWDF chooses a NRT packet, then LWDF stops accepting NRT if filled slots are greater than or equal to $All.Max.Limit$. Thus the chosen NRT packet gets higher maxima limit every 6^{th} frame.
2. Alternatively in the same 6th frame, if RT packet is chosen based on LWDF function, the scheduling process will be subject to $RT.Max.Limit$, which could be lower than $All.Max.Limit$. Thus RT gets lower limit & priority in every 6^{th} frame.
3. The limit check in other frames (1 to 5) is done by swapping the upper-limit of RT with NRT. Thus in 1^{st} *thru* 5^{th} frames RT gets higher limit than NRT.
4. If balance slots available to fill up to $All.Max.Limit$ is less than the chosen packet size, then the packet will be the last one to be enlisted in LWDF round, and it will be fragmented before getting enlisted.

4.5 Process for DiscMOPSO round

DiscMOPSO is a Discrete-Multi-Objective PSO algorithm. The basic PSO algorithm is introduced in the Introduction Section. The discrete form of PSO considered in this paper uses the integer approach, i.e. it processes packets in its full original size without fragmenting it. Multi-Objective PSO is formed by incorporating sub-functions into the objective function, and each sub-function represents one or more objectives namely best-goodput, reduced starvation, deadline compliance and PLR compliance.

Initially before the start of PSO iterations, the particles are generated as a vector list that contains the number of packets against each connection. This forms a candidate solution suggested by each particle. After the particles (solutions) are generated , it is resized for the best fit so that it fills between 98% and 100% of the available resources, i.e. slots in the OFDM frame. If the best fit is not possible, then the particle is resized into one of the nearest possible smaller sizes that fit. Resizing is done by cyclic addition and deletion of random packets in the list, until the particle fits in. Only whole packet counts are used in this resizing exercise to ensure the PSO is *discrete* in nature.

After resizing of particles, each particle is evaluated using an objective function defined by equation (11) that computes the total penalty P. Best particle is selected in each iteration and called as $gbest$. Each particle's best position is found based on its path (values in past iterations) and marked as $pbest$. The resizing operation is done in every iteration after particle update is done.

In the LDEH scheme, DiscMOPSO is used only for scheduling NRT packets and not RT packets, in order to reduce the processing time. Since PSO is a time-consuming process, scheduling RT using PSO multiplies processing time by an order proportionate to the number of RT connections which can go up to 75 per class as per the scenarios considered. The number of NRT connections are limited to 25 and the number of NRT packets to fill a frame is typically a single digit. For 25 NRT connections, there are 10x25=250 possible combinations of discrete packets that load a frame. Thus if the number of particles is set as 45 and the number of iterations is set as 30, the number of possible solutions generated are 30x45=1350. The solution space of 1350 as searched by DiscMOPSO sufficiently covers all possible combinations of loads (i.e. 250) multiple times. Thus the choice on the number of iterations and particles of PSO process is explained. Due to reduced iterations and flows, DiscMOPSO process proposed in this LDEH scheme becomes less complex and faster by an order of 15 as compared to a DiscMOPSO round that handles both RT & NRT Packets. Another reason to restrict DiscMOPSO to NRT is that most of the RT packets normally get scheduled by the LWDF round, due to the RT-prioritization in LWDF that gives 10 to 15 times higher weightage to RT than NRT.

After particles are generated at the beginning of each frame and just before the start of iterations, the process proceeds directly to evaluate the objective function (i.e.penalty) value as described subsequently below. But once the process enters in the PSO iterations, in each iteration and for each particle, position or velocity update is done by means of the equation (9) as follows before evaluating its objective function. This update moves the particles towards $pbest$ and $gbest$.

$$v_i = r1_i + r2_i \tag{9}$$

where

$$r1_i = (pbest_i - p_i)$$

and

$$r2_i = (gbest - p_i)$$

Where $r1_i$ is a random integer limited $<=$ to the difference between the number of packets in the current particle and its $pbest$ value for the i^{th} connection. Similarly, $r2_i$ is limited to the difference between the current particles and the $gbest$ for the given i^{th} connection. This step ensures the particle solutions move towards the best direction, but the jump size is randomized in order to avoid local minima during the optimization process.

Then each particle's current position (i.e. candidate schedule) is updated by adding them to the respective velocity updates by formula (10).

$$p_i = p_i + v_i. \tag{10}$$

Where p_i is the number of packets provisioned by the current particle p for i^{th} connection.

v_i is the jump as described in equation (9).

Note: In discrete-PSO algorithm, variables, $p_i, v_i, r1_i$ and $r2_i$ are all integers and they all indicate that the iterations are based on the number of packets. This ensures that the packets are processed in full, irrespective of the packet sizes. After each update, the particle is subjected to maximum resource availability check and *resized* as done in initialization so that the particle fits in a *discrete* manner.

In each PSO iteration, after completion of position update and resizing operation, the objective function, i.e. the Penalty function value "P" is computed by equation (11) as follows:

$$P = (cP1 * P1) + (cP2 * P2) + (cP3 * P3) \tag{11}$$

Where P1,P2,P3 are sub-penalties representing the multiple-objectives, namely opportunistic scheduling, queue-vs-allotment mismatch and starvation of low-class packets respectively. cP1, cP2 & cP3 are coefficients whose values are set as 1, 2 & 1 respectively based on an analysis to determine the best set of coefficients.

The first penalty P1 is computed by the proposed formula (12)

$$P1 = \frac{(pavgchdur - avgchdur)}{avgchdur}. \tag{12}$$

where

$$pavgchdur = \frac{\Sigma_i(Slots_{ip})/MCS_i}{Tot\ slots_p}$$

$$avgchdur = \frac{\Sigma_i(Q_i)/MCS_i}{(Q\ Tot)}$$

$Slots_{ip}$ is the slots provisioned for i^{th} connection in the current particle.

$Tot\ slots_p$ is the total slots for all connections in the current particle

MCS_i represents the channel quality for i^{th} connection.

Q_i is the queue size for i^{th} connection.

$Q\ Tot$ is the total queue size for all connections.

$avgchdur$ represents the difference between slots demanded by the queue and the respective channel quality, i.e. MCS value.

$pavgchdur$ represents the mismatch between the particle's allocation and the respective channel quality. Effectively, the P1 penalty goes high if resource allocation demanded by the queue and the allocation suggested by the particle are mismatched. P1 is more sensitive when queue and channel qualities match each other and particle is not matched. In case a particle's allocation matches the queue exactly, P1 will be zero. Effectively P1 rewards opportunistic and channel-aware scheduling.

P2 penalty represents the mismatch between the particle and demand as per queue size, normalized over respective CoS's average packet size. All the sizes are expressed in terms of slots. P2 penalty is also sensitive for good channel conditions and it is computed as in the proposed equations (13, 14) for HYB-1 & 2 respectively.

P2 for HYB-1 scheme

$$P2 = \Sigma_i \frac{(qslots_i - alloc\ slots_i)}{(Avg\ pkt\ size\ for\ COS\ of\ i^{th}\ Conn)} \quad (13)$$

P2 for HYB-2 scheme

$$P2 = \Sigma_i \frac{MCS_i * (qslots_i - alloc\ slots_i)}{(Avg\ pkt\ size\ for\ COS\ of\ i^{th}\ Conn)} \quad (14)$$

Where $qslots_i$ = queue size in slots for the i^{th} connection. $alloc\ slots_i$= size of resource allocation done by the particle.

Thus above P2 penalty value is a measure of mismatch between queue requirements and the respective allocation by the particle. The only difference between the above two P2 equations is that, HYB-2 scheme's obj.function is multiplied by the RF channel factor MCS. Thereby it strengthens opportunistic scheduling and improves performance in dynamic channel fading conditions for HYB-2.

P3 penalty is a measure of unfairness and accounts for starved packets. P3 in effect reduces deadline miss and it is computed by the proposed formula (15):

$$P3 = (No.\ of\ Pkts\ starved\ for\ 140\ to\ 280\ ms) \quad (15)$$
$$+\ 3 * (No.\ of\ Pkts\ starved\ for\ >\ 280\ ms)$$

The choices of above starvation delay thresholds are based on the 3GPP specifications [12] which recommends an end-to-end delay budget limit of 300 ms for FTP and buffered streaming non-conversational video (NRT). After deducting 20 ms mobility margin[11], 280 ms is considered as the limit for triggering P3's higher sub-penalty of 3 points. Single-end limit is considered half of 280 = 140 ms for triggering the P3's lower sub-penalty of 1 point.

At the end of each iteration, each particle's best position in its path i.e. *pbest* is updated. Similarly the best position of all particles i.e. *gbest* is also updated. These updates are done if better schedules are found in that iteration with lower penalties. At the end of all iterations of DiscMOPSO in a frame, the *gbest* particle (schedule) is considered as the best schedule output by the DiscMOPSO round. The final step of DiscMOPSO is to enlist those packets from the *gbest* schedule onto the available space in the pre-pack queue for that frame.

The final round of flow scheduling i.e. EDF will be executed if there are vacant slots to be enlisted after the previous two rounds. In this EDF round, the last packet may get fragmented in a manner similar to the LWDF round. In HYB-2, EDF round is also made channel-aware by multiplying MCS value in the EDF function. This modified-EDF effectively prioritizes and schedules packets with higher channel quality.

Thus, at the end of the packet or flow scheduling process that consists of MLWDF, DiscMOPSO and EDF rounds, a pre-pack list is produced which is passed to the frame packing algorithm.

5 FRAME PACKING ALGORITHM

Orientation-Based Burst Packing (OBBP)[10] is a fast and efficient frame-packing algorithm. In the LDEH scheme, a modified-OBBP algorithm is proposed herewith as described below, and this results in improved spectral utilization that goes up to 0.9bps/Hz.

1. Packets in the pre-pack queue are broken into a max of 60 slots or part thereof. Each packet forms an OFDM rectangular burst. This limit of 60 is chosen based on an analysis and comparison with other sizes with high factorization such as 40, 75, 90, 120 and 140. Size of 60 resulted in the best performance for various scenarios. Since the OFDM frame size is 30, which is a factor of 60, it results in 100% occupation of columns height.
2. For frames with high utilization and less number of small bursts, further fragmentation of large bursts (>30 slots) into smaller ones of 5 slots each is done in order to increase frame utilization and spectral efficiency. Only a max of 6 smaller bursts are generated by this fragmentation.
3. The column filling is done by max-sorted bursts (i.e. large burst is filled first) up to a level where 80% of columns are filled. The remaining columns are filled by min-sorted bursts in order to improve RT performances.

6 SIMULATION SPECIFICATIONS

The simulation parameters are summarized in Table (1). The QoS specifications of 802.16m [11] in terms of recommended max packet delay, packet loss ratio, as well as other recommendations such as rectangular OFDM block allocation, HARQ, CRC and dynamic MCS variation have been considered and adopted in this paper for simulation. Most of these are explained in the *Introduction* section and also under the heading *Initializing packets as per* 802.16*m standards* in Sub-Section 4.2 .

Sixteen levels of channel quality (MCS index) is permitted as per radio interface standard of 802.16m [11, 16]. The highest channel quality of MCS index value 1111 (4-bit binary) pertains to 64-QAM modulation and FEC code rate of 237/256. The lowest MCS index of 0000 pertains to QPSK modulation and FEC code rate of

Table 1: SIMULATION SPECIFICATIONS

PARAMETER	VALUE
Number of frames simulated for each run of a given scheduler	1200 frames
Available downlink RF Bandwidth	10 MHz.
Duration per frame.	5 ms.
Duplexing & Direction	FDD & Downlink.
Size of OFDM frame in Freq domain axis	Total of 1024 sub-cxrs for each frame.
Usable size in Freq Domain	30 sub-channels x 28 sub-cxr = 840 usable sub-cxrs per frame.
Size of OFDM frame in Time domain axis :	Total of 24 slot-widths per frame 20 usable slot-widths for a super-frame and 22 usable slot-widths for a normal frame. One slot-width = 2 symbols duration
Granularity of OFDM resource block	Slot Size = 28 sub-cxr height and 2 symbol wide. Total of 56 symbols per slot.
Data and Pilot symbol usage per slot	48 data symbols and 8 pilot symbols per slot.
Total usable slots per super frame & per Normal frame	600 slots per Super frame & 660 slots per normal frame.
Number of traffic scenarios simulated in each scheduler	9 scenarios in Low RT/NRT Ratio and 6 in High RT/NRT Ratio
Traffic intensity	0.8 to 1.55
RT/NRT Traffic Volume ratio	Low RT/NRT Ratio: From 0.1 to 1.0 High RT/NRT Ratio: From 1.1 to 1.9
Total number of TCP connections	55 to 250
Traffic class mix ratio [UGS : ERTPS : RTPS : NRTPS : BE]	Each of 15 Scenarios have different mix ratios between [10 10 10 10 20] to [75 75 75 5 20] .
Downlink Dealines for [UGS, ERTPS, RTPS, NRTPS and BE] pkts	[80, 80, 60, 1000, 1000] respectively in milliseconds
MCS, Coding rate and Modulation Scheme for each mobile.	For each scenario, a fading sequence is used for each connection.The same sequence is used to evaluate all schedulers.

31/256.

The results are compared with other algorithms, namely LOG, EXP-W and M-LWDF. The same set of common simulation environment and parameters including traffic scenarios, fading conditions and frame packing algorithm are used across all scheduling algorithms so that a fair evaluation and comparison is done between the schedulers simulated herewith.

7 RESULTS

The following section presents the simulation results by comparing performance metrics, namely goodput, average packet delay, packet loss ratio, starvation index and spectral efficiency for the proposed LDEH-1 and LDEH-2 Hybrids (HYB-1 and HYB-2) with existing recently referred schedulers, namely M-LWDF, LOG and EXP-W.

The end applications used for simulation of each class are: VoIP without silence suppression for UGS Class, VoIP with silence suppression for ERTPS, MPEG-Video for RTPS, FTP for NRTPS and HTTP for BE classes of service. Highly-overloaded traffic with Traffic intensities ranging from 0.8 to 1.6, with 50 to 250 connections per scenario are fed into these scheduling algorithms. Thus these results test the schedulers under heavily-stressed and demanding load conditions. As stated above in the ”Simulation Specifications” section, the schedulers are compared in an equal platform, i.e. under a set of common simulation specifications and traffic conditions. The results for each RT metric are based on the sum of results of all three RT classes, namely UGS, ERTPS & RTPS. Similarly, NRT results are based on the sum of NRTPS and BE results.

From the analysis of results, it is found that for overloaded heterogeneous traffic, RT to NRT traffic volume ratio factor highly influences the results of various performance metrics. Hence results are herewith segregated for two ranges, namely low-RT/NRT ratio ($<$

1) and high-RT/NRT ratio (>1 and <2.0) and they are presented herewith separately. The ability of the schedulers to be dynamically adaptive in the performance without much degradation in either RT or NRT is a critical aspect that has come out in this analysis.

FLS scheduler or HWEL-MT scheduler are not considered for comparison in this paper, since as analyzed by the respective reference papers [1], [7] and [2], FLS & HWEL-MT have been proved in limited traffic intensity up to TI = 0.9.

Goodput is defined as the data rate of scheduled payload data for the respective i^{th} class during the observation period, and expressed in bps as computed by formula (16).

$$Goodput(i)\ in\ bps = \frac{Total\ payload\ Volume(i)\ in\ bits}{Observation\ period\ in\ secs} \tag{16}$$

Where *Total payload Volume* in above is computed by deducting all overheads, namely packet headers and CRC bits from the gross throughput volume for the i^{th} class and it is expressed in bits per second. Overheads added during packet fragmentation are also deducted.

A method of quick comparison of the competing schedulers is proposed herewith by means of using *cumulative deviation* parameter. For example, cumulative deviation for goodput for any given k^{th} scheduler is computed by formula (17):

$$CumDev(k) = \Sigma(i)\frac{(Best\ Goodput(i) - Goodput(i,k))}{Best\ Goodput(i)} \tag{17}$$

The above Sigma is done over all the traffic scenarios for a given range of RT/NRT ratio.
Where “i” represents traffic scenario or traffic,
$Goodput(i,k)$ is goodput of k^{th} scheduler for i^{th} traffic scenario, and
$Best\ Goodput(i)$ is the best goodput among all schedulers for i^{th} traffic scenario.

Thus the cumulative deviation represents the normalized sum of deviation from the best performing scheduler across all scenarios in a given RT/NRT range. Large cumulative deviation indicates poor relative performance across given set of scenarios. These are illustrated separately for low and high RT/NRT ranges.

The RT-goodput results shown in Figures (2, 3 and 4) prove that HYB-1 scheduler provides the best RT-goodput in all scenarios including low and high RT/NRT conditions. HYB-1 is followed by M-LWDF and then by HYB-2 in results with a variation among these top three schedulers within 1% from the best as averaged per scenario. It also shows that LOG results in the worst RT-goodput among them in low RT/NRT scenarios in which it is lower by an average of 2% per scenario and by a peak drop of 6% in a scenario. In high RT/NRT ratio, LOG result is relatively better at a deviation from the best by 0.4% per scenario.

Note: The *cumulative deviations* illustrated in bar chart figures (4, 7, 10, 13 and 25) summarize the relative performances of all schedulers in the respective performance metrics. These graphs show the normalized variation from the respective best values by using formulae similar to equation (17).

The NRT-goodput results are shown in Figures (5, 6 and 7). For low RT/NRT ratio, HYB-1 and HYB-2 produce the best NRT-goodput with both sharing the top rank. These are followed by LOG and M-LWDF in second rank. For high RT/NRT ratio, HYB-2, LOG and M-LWDF schedulers are equally ranked at the top for all scenarios. Then it is followed by HYB-1. At high RT/NRT, HYB-1 provides slightly lower NRT-goodput due to the very high RT traffic volume and HYB-1’s RT prioritization policy

The real-time-packet-delay results shown in Fig (8, 9 and 10) prove that HYB-1 scheduler provides the best and the lowest RT-packet delay in all scenarios. HYB-1 is followed by HYB-2 and LOG. HYB-2’s average RT-packet delay comes next to HYB-1 in 6 out of 9 low RT/NRT and in all high RT/NRT scenarios with an average deviation of 6 ms from HYB-1 in all conditions. In case of LOG, RT-delays are higher by 6 ms and 1 ms respectively. For M-LWDF, the RT-delays are higher by an average of 11 ms for low- as well as high- RT/NRT conditions. M-LWDF results in the highest RT delays in 9 out of 15 scenarios and second highest RT-delay in remaining 6 scenarios. Note: Packet delays of only scheduled packets are accounted in RT-delay results. Thus the packet delay metric does not account for the starved or dropped packets.

The non-real-time packet delay results shown in Figures (11, 12 and 13) shows that LOG scheduler provides the least and the best NRT packet delay in high- and low-ratio traffic conditions. HYB-2 is the next best performing scheduler for NRT-packet delay with an increase of 9% at low RT/NRT ratio and 13.8 % for high RT/NRT ratio compared to the respective best. When compared to NRT deadlines as per end-to-end performance specifications, the deviation is a low percentage. The corresponding increase of NRT-delays for M-LWDF are 12% and 15% respectively. HYB-1 NRT-delays are higher by 17% and 38% respectively due to its RT prioritization. It should be noted again that these delays are based on scheduled packets and not on starved packets. Further interesting observations are made based on NRT-starvation results in paragraph titled *Summary of Results* in which the delay and starvation metrics are compared.

$Starvation\ Index(SI)$ is a measure of volume of unscheduled gross data and defined by the formula (18).

$$SI(i) = \frac{Vol\ of\ unscheduled\ data\ for\ i^{th}\ class}{Volume\ of\ generated\ data\ for\ i^{th}\ class} \tag{18}$$

Where $SI(i)$ is the Starvation Index for i^{th} class. *Cumulative deviations* of starvation indices in Fig (16 & 19) and RT-PLR in Fig (22) are already in normalized

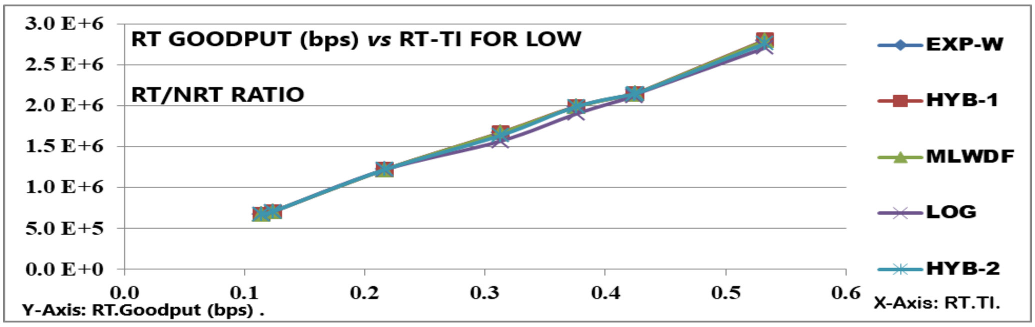


Fig. 2: RT-Goodput (bps) versus RT-Traffic Intensity for Low RT/NRT Trf Ratio

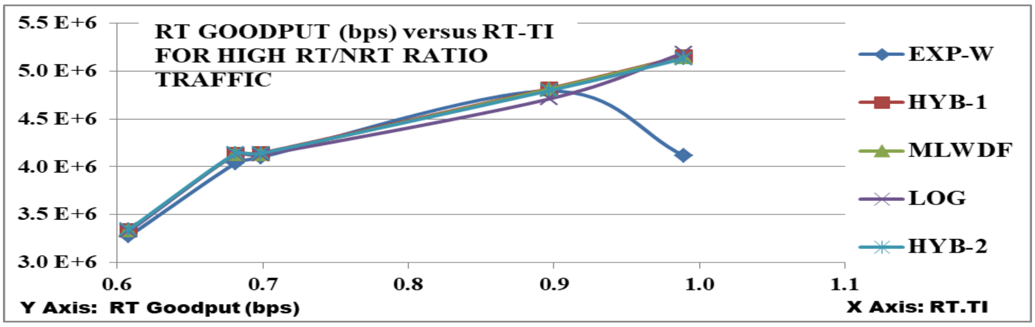


Fig. 3: RT-Goodput (bps) versus RT-Trf Intensity for High RT/NRT Trf Ratio

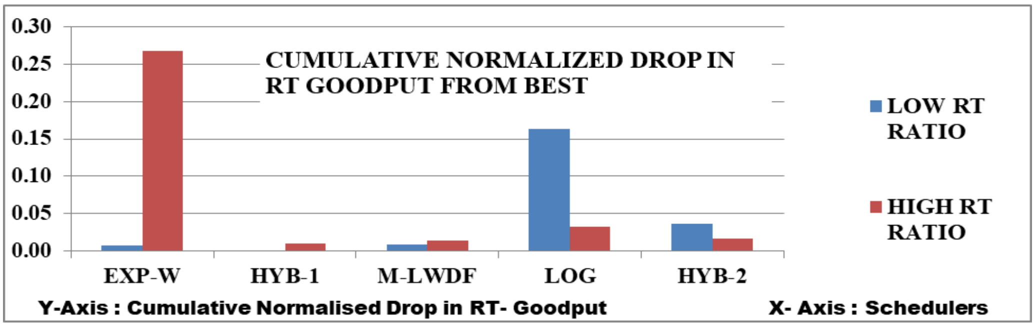


Fig. 4: Cumulative Normalized Drop in RT-Goodput w.r.t Best among various Schedulers

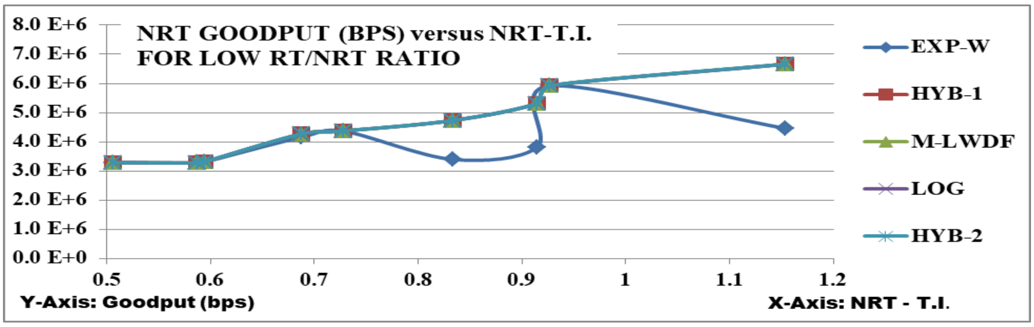


Fig. 5: NRT-Goodput (bps) versus NRT-Traffic Intensity for Low RT/NRT Trf Ratio

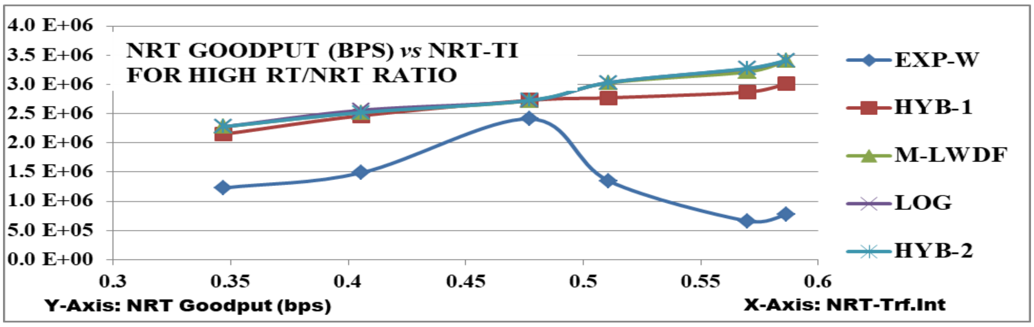


Fig. 6: NRT-Goodput (bps) versus NRT-Traffic Intensity for High RT/NRT Trf Ratio

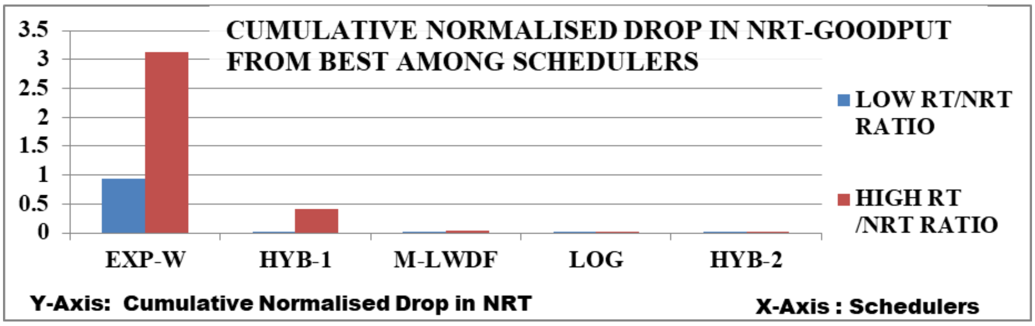


Fig. 7: Cumulative Normalized Drop in NRT-Goodput w.r.t Best among various Schedulers

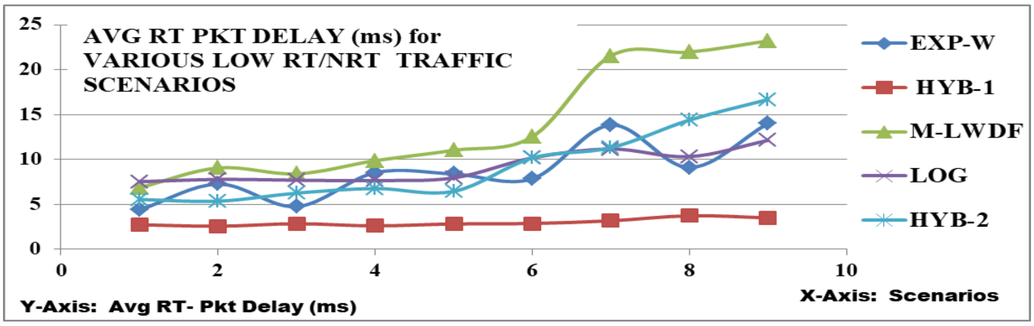


Fig. 8: Avg RT-Packet-Delay (ms) for various Traffic Scenarios with Low RT/NRT Trf Ratio

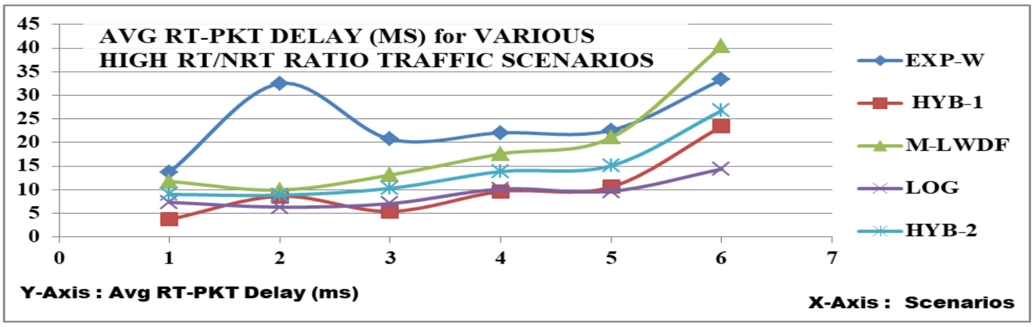


Fig. 9: Avg RT-Pkt-Delay (ms) for various Traffic Scenarios with High RT/NRT Trf Ratio

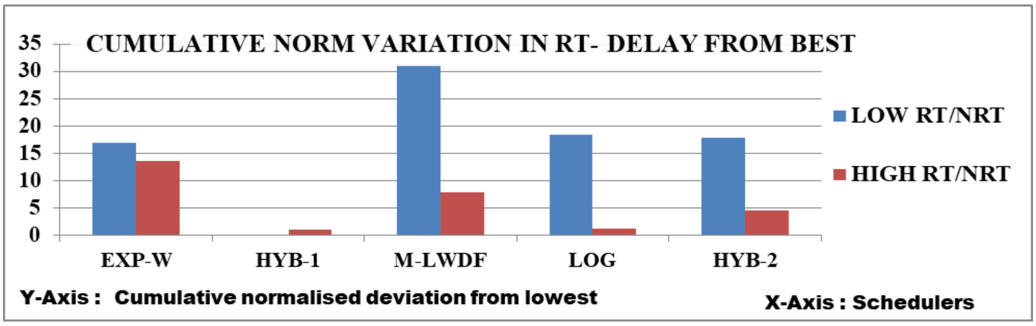


Fig. 10: Cumulative Normalized Increase in RT-Pkt-Delay w.r.t Best among various Schedulers

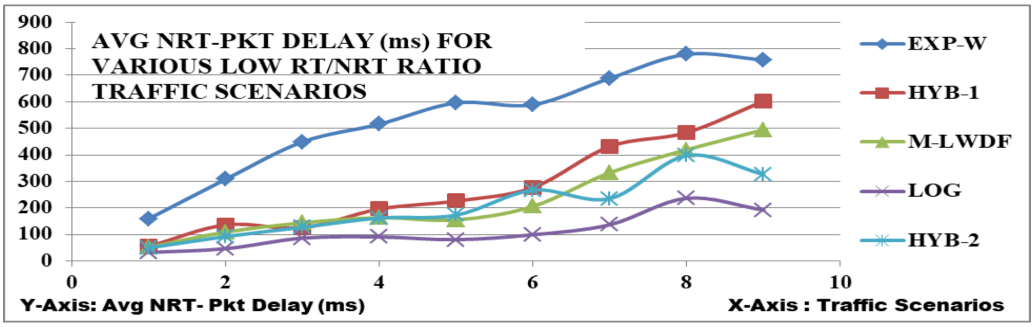


Fig. 11: Avg NRT-Pkt-Delay (ms) for various Traffic Scenarios with Low RT/NRT Trf Ratio

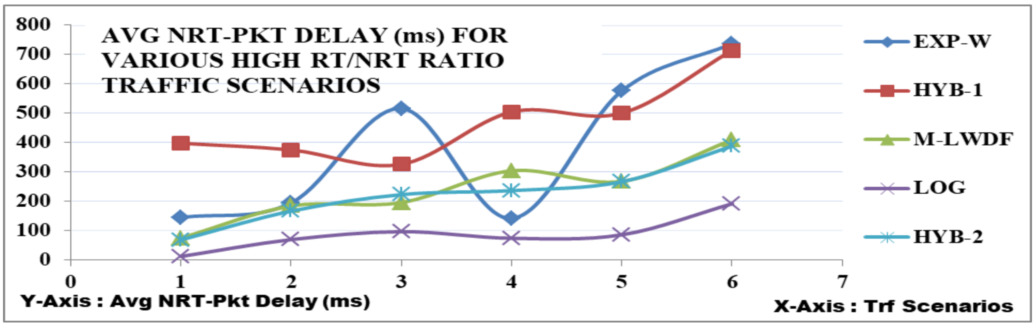


Fig. 12: Avg NRT-Pkt-Delay (ms) for various Traffic Scenarios with High RT/NRT Trf Ratio

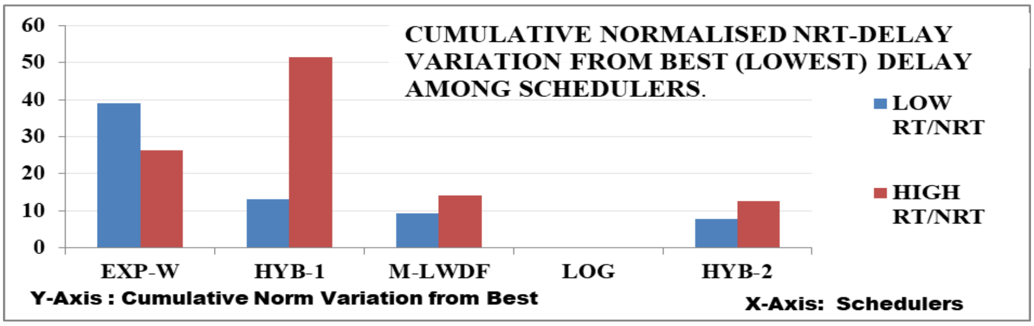


Fig. 13: Cumulative Normalized Increase in NRT-Pkt-Delay w.r.t Best among various Schedulers

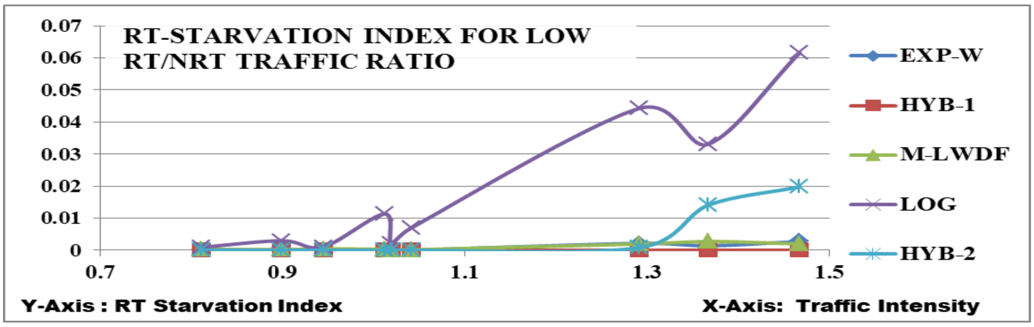


Fig. 14: Avg RT-Starvation Index versus Traffic Intensity for Low RT/NRT Trf Ratio

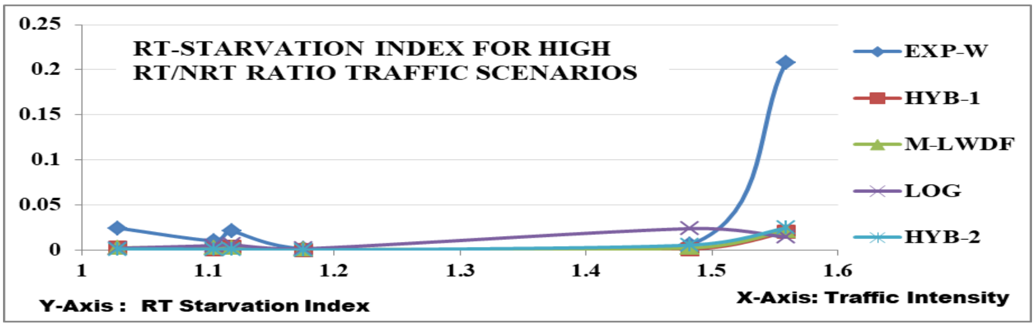


Fig. 15: Avg RT-Starvation Index versus Traffic Intensity for High RT/NRT Trf Ratio

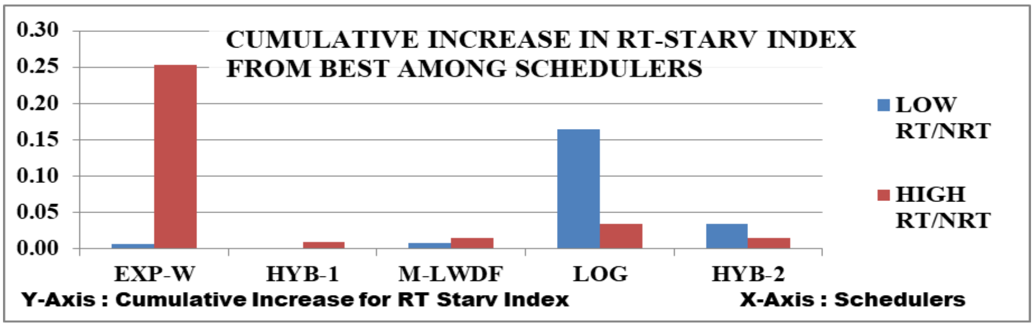


Fig. 16: Cumulative Normalized Increase in RT-Starvation Index w.r.t Best among various Schedulers

form, i.e. normalized with respect to generated traffic volume in the denominator.

The Real-Time Starvation Index (RT-SI) results are shown in Figures (14, 15 and 16). RT-starvation determines and affects the end-to-end QoS performance of streaming data significantly, since it is measure of undelivered data that is not re-transmitted for streaming RT services. The results prove that HYB-1 scheduler provides zero RT-starvation in all scenarios and thus is the best for this metric. HYB-1 is followed by M-LWDF and HYB-2 with an average variation $<$ 0.33% and 0.5% for high- and low-RT/NRT ratios respectively. Non-delivery of information is within 1% limit for the Hybrids and MLWDF and it complies with ITU specifications [14]. It also shows that LOG's RT-starvation performance is the worst for low RT/NRT ratio as it starves an average of 2% RT data per scenario with a peak of 6% in one scenario. For high RT/NRT ratio conditions, LOG results in an average starvation of 1% per scenario with a peak of 2%.

The Non-Real-Time Starvation Index (NRT-SI) results are shown in Figures (17, 18 and 19). For low RT/NRT, results prove that HYB-1, HYB-2, LOG and M-LWDF schedulers equally provide almost zero NRT-starvation and they are the best in NRT-SI for all conditions. EXP-W results in the worst performance for this metric with an average of 10% starvation per

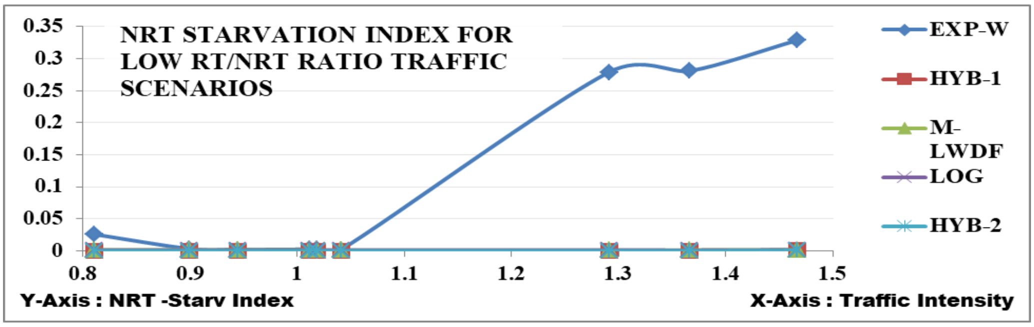


Fig. 17: NRT-Starvation Index versus Traffic Intensity for Low RT/NRT Trf Ratio

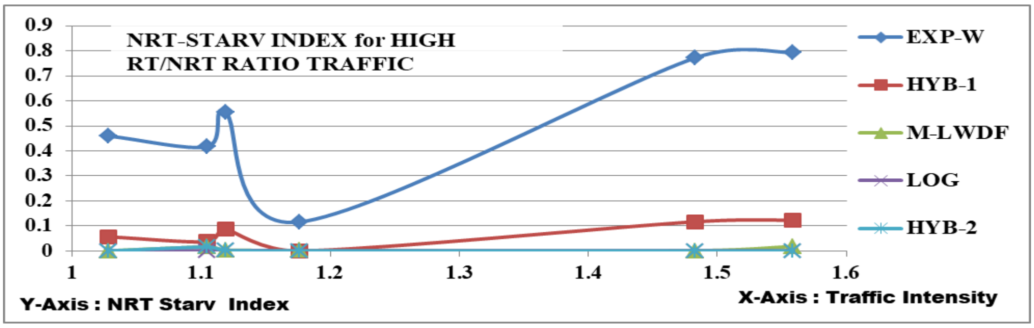


Fig. 18: NRT-Starvation Index versus Traffic Intensity for High RT/NRT Trf Ratio

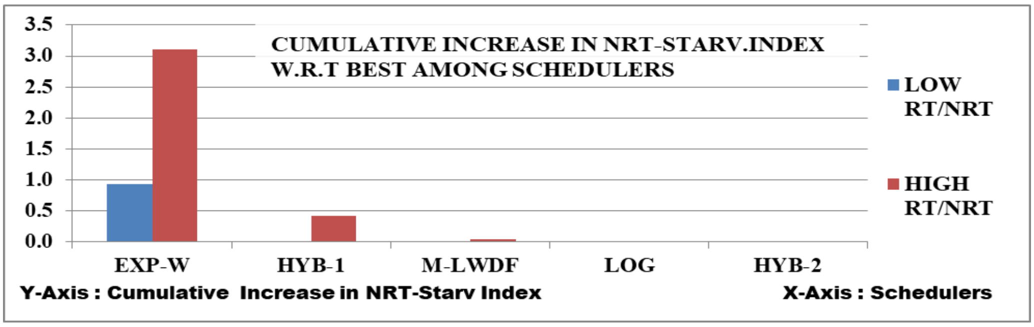


Fig. 19: Cumulative Normalized Increase in NRT-Starvation Index w.r.t Best among various Schedulers

scenario. For high RT/NRT conditions, LOG and HYB-2 are the best with results of almost zero NRT-starvation. M-LWDF results in about 0.67% starvation per scenario, followed by HYB-1 at 6.8% and EXP-W at 52%.

Packet Loss Ratio(PLR) is the number of packets lost (due to non-scheduling over the max HARQ period) and is computed by formula (19).

$$PLR(i) \ in \ \% = \frac{No.\ of\ Pkts\ dropped\ for\ i^{th}\ class * 100}{Total\ No.\ of\ Pkts\ generated\ for\ i^{th}\ class} \tag{19}$$

RT-PLR is expressed as a percentage of total number of packets generated for that class.

The real-time packet loss ratio results shown in Figures (20, 21 and 22) prove that HYB-1 scheduler provides the lowest and best RT-PLR in all scenarios. HYB-1 is followed by M-LWDF and then by HYB-2 as shown by the cumulative deviation graph. The variation among the top three, namely HYB-1, M-LWDF and HYB-2 is within 0.23% from the best for high & low RT/NRT ratios. It also shows that LOG's performance is the worst in RT-PLR metric for both high- and low-RT/NRT ratios in which LOG drops an average of 1.5% of RT packets per scenario with a peak of 5% in one scenario for low RT/NRT conditions. LOG drops an average of 0.7% of RT packets per scenario with a peak of

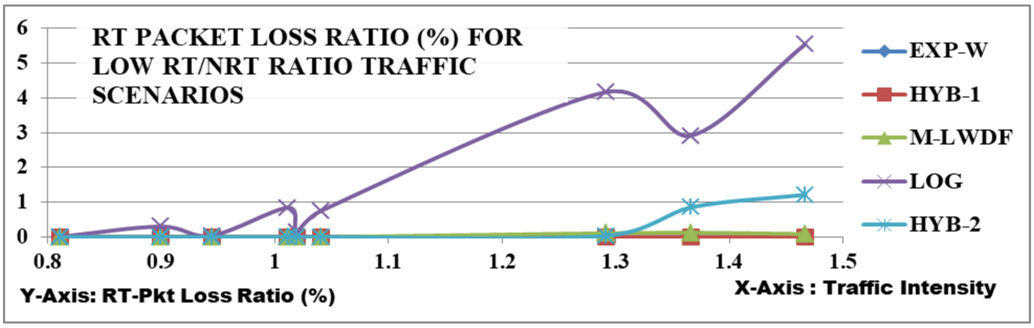


Fig. 20: RT-Packet Loss Ratio (%) versus Traffic Intensity for Low RT/NRT Trf Ratio

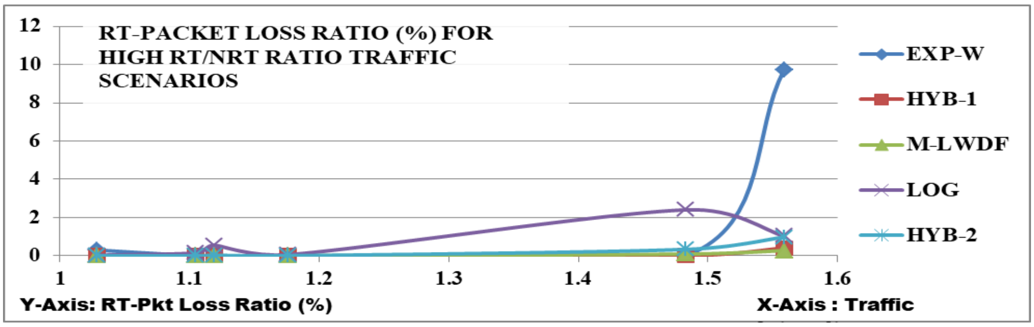


Fig. 21: RT-Packet Loss Ratio (%) versus Traffic Intensity for High RT/NRT Trf Ratio

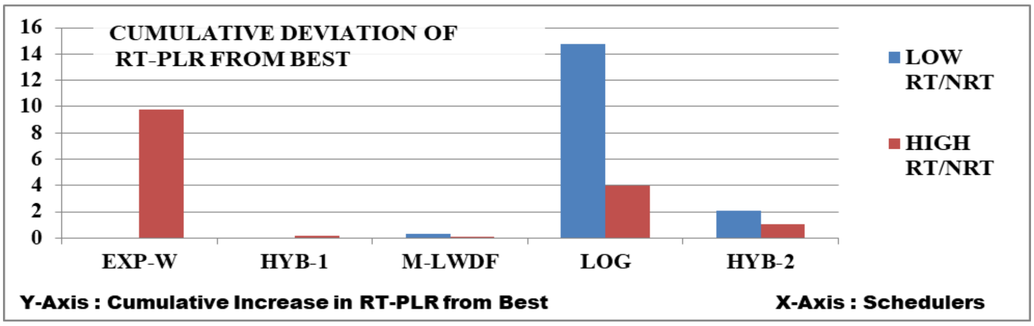


Fig. 22: Cumulative Normalized Increase in RT-Packet Loss Ratio w.r.t Best among various Schedulers

2.4% in one scenario for high RT/NRT conditions. Thus LOG exceeds the ITU specification [14] of 1% upper limit for packet drop, in 5 out of 15 scenarios.

The spectral efficiency results are shown in Figures (23, 24 and 25). Spectral efficiency is defined as the total data rate that the system can transmit in one Hertz of spectrum. Spectral efficiency is expressed in bits per second per Hertz and calculated by the formula (20).

$$Spectral\ Eff = \frac{Total\ payload\ data\ volume\ scheduled}{(Obs\ period * Spectral\ Bandwidth)} \quad (20)$$

Where *Total payload data volume scheduled* is the net volume of data in bits computed without adding the OFDM frame redundancies or overheads etc.
Obs period is the total simulation period which is 6 seconds here.
Spectral Bandwidth is the RF channel bandwidth which is 10 MHz.

The spectral efficiency is computed based on net throughput without considering buffer-slots, FEC bits or other OFDM frame overheads, since the net throughput shows the effective payload efficiency. Inefficient schedulers allot more buffer slots, carry less payload and show higher gross throughput. Thus analysis of net-throughput-based spectral efficiency identifies such inefficient schedulers. Spectral efficiency is also traffic

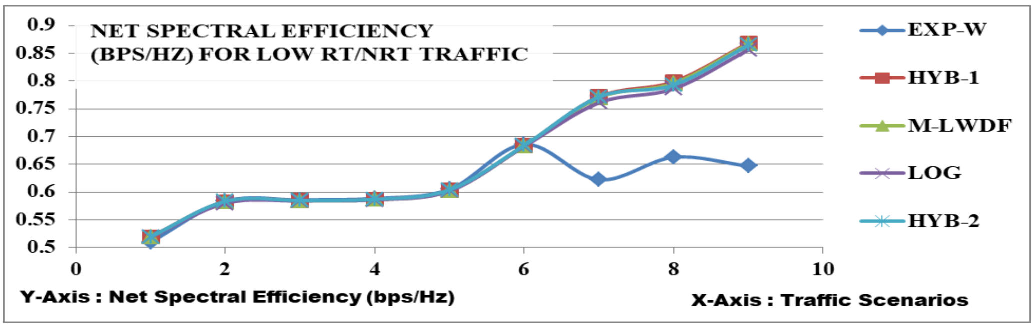


Fig. 23: Net Spectral Efficiency (bps/hz) for Scenarios with Low RT/NRT Traffic Ratio

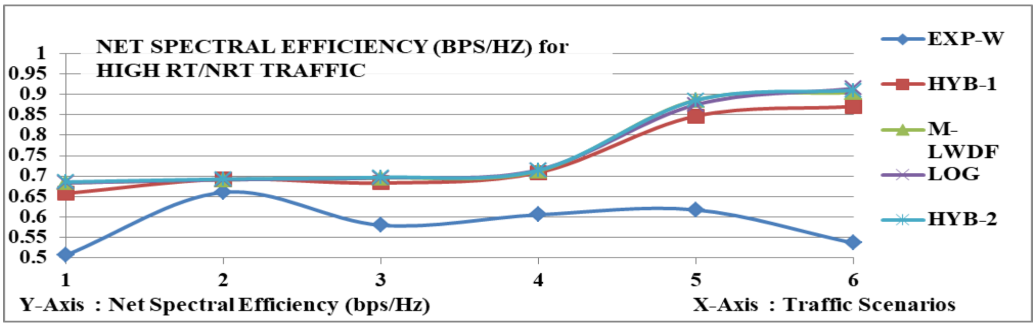


Fig. 24: Net Spectral Efficiency (bps/hz) for Scenarios with High RT/NRT Traffic Ratio

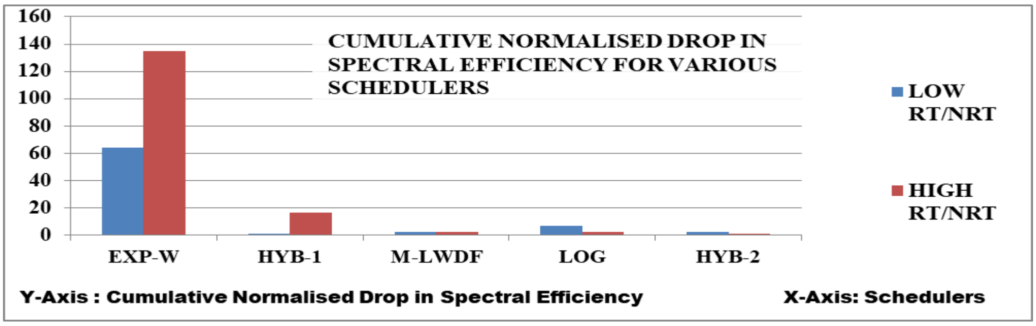


Fig. 25: Cumulative Normalized Drop in Spectral Efficiency for various Schedulers.

dependent since it is based on the volume of generated traffic.

The spectral efficiency results are shown in figures (23, 24 and 25). For low RT/NRT, HYB-1 and HYB-2 are the best ranked. They are followed very closely by M-LWDF. For high RT/NRT, HYB-2 is again the best followed by M-LWDF and LOG. HYB-1 has 2.7% lower average spectral efficiency in such high RT/NRT due to its RT prioritization. RT adds to spectral inefficiency due to smaller packet size and relatively higher overheads. When spectral efficiency results of low and high RT/NRT are put together, it is proved that HYB-2 delivers the best spectral efficiency for all traffic conditions simulated, followed by M-LWDF and then by LOG schedulers. HYB-1 shares the best Sp.Efficiency rank only for low RT/NRT scenarios.

The Exp-W scheduler's performance is poorer, volatile & inconsistent than others in all metrics due to its exponential objective function which makes it highly sensitive to traffic conditions.

Summary of Results:

1. In RT-goodput metric, HYB-1 as well as HYB-2 provide the best or close to the best results.
2. In NRT-goodput metric, HYB-2 provides the best results in all scenarios of low and high RT/NRT traffic. HYB-1's NRT–goodput gets affected slightly

for overloaded RT traffic conditions, i.e. high RT/NRT ratio.

3. In RT packet delay metric, HYB-1 produces the best results in all traffic conditions.
4. NRT-delays of HYB-1 are relatively high in high-traffic. In spite of this, HYB-1's average NRT-delay are well within the deadline limit considered, particularly in the overloaded traffic.
5. The RT-starvation index results prove that HYB-1 scheduler is the best for RT services as it causes zero starvation in all low RT/NRT scenarios and close to zero in high RT/NRT scenarios. RT-starvation affects the end-to-end RT performance badly. In the top ranks, HYB-1 is followed by M-LWDF and then by HYB-2 with negligible difference between them in all scenarios. LOG's RT-starvation performance is the very bad in this.
6. For NRT-SI metric and under low RT/NRT traffic, most of schedulers except EXP perform well. Under high RT/NRT, HYB-2 ranks the best in NRT-SI, co-existing with LOG. HYB-1's NRT-SI for high RT/NRT is at third rank due to its RT prioritization.
7. RT-PLR results are similar to RT-SI.
8. Net spectral efficiency is a good measure to prove that HYB-2 is the most consistent in all low and high ratio traffic conditions. HYB-1 ranks among the best in low RT/NRT and next to HYB-2 in high RT/NRT.
9. EXP-W performs poorly in many metrics with high levels of volatile or sensitive performance that gets affected by loading conditions.

Comprehensive performance ranking of the schedulers is shown in rankings Tables (2, 3 and 4) across all scenarios & performance metrics. A scheduler with Rank=1 has the best result i.e. lowest cumulative deviation for that parameter totaled across all traffic scenarios in given RT/NRT range. Schedulers with higher ranks have poorer results. More than one scheduler can get the same ranking for a parameter when their cumulative deviation differs by less than 1 % between each other. Table (4) provides the overall average ranks for low and high RT/NRT put together, and these prove that LDEH-1 has the best (i.e. lowest) average rank followed by LDEH-2.

The standard deviation of ranks indicates the consistency of the scheduler's performance across metrics and its ability to maintain fairness and avoid compromise of one metric or class over the other. LDEH-2 has the lowest standard deviation of ranks and thus is the most consistent and fair one.

8 PERSPECTIVE

The LDEH schedulers proposed has been tested to produce the best performance in near-real-world overloaded traffic conditions and meet the QoS requirements of IEEE802.16m standard. The testing methods had simulated high-traffic intensities up to 1.6, hundreds of connections and various traffic mix ratios. Accordingly, LDEH schedulers with two variants have been developed, simulated and tested under such load conditions. The results are compared with contemporary and recently referred high performance channel-aware, load-aware & QoS-aware schedulers such as LOG, M-LWDF and EXP-W. The results prove that the proposed LDEH-1 (HYB-1) Scheduler provides the best overall performance among the compared schedulers, across various performance metrics and scenarios. LDEH-1's top performance is followed closely by the proposed LDEH-2 scheme which gets the second overall rank. LDEH-2 is also the most consistent and fair scheduler as it performs very well and remains unbiased across all metrics even under overloaded traffic conditions.

The proposed hybrid schemes can be deployed in any OFDM scheduler handling multi-media heterogeneous traffic such as 5G network or other new generation networks, with due adaptation of new standards.

The results analysis and the proposed methods of evaluation based on deviation analysis and rankings provide a comprehensive assessment of each schedulers abilities and brings out the impact of RT/NRT ratio on the performance. The methods of analysis proposed here can be used for future evaluation of schedulers for new generation of high-performance networks. The results also highlight the necessity to test heterogeneous schedulers to various traffic conditions.

Acknowledgment

The authors are grateful to the anonymous referee for carefully checking the details and providing helpful comments that improved this paper. The authors thank the management of Coimbatore Institute of Technology for providing the facilities for doing this research work.

References

[1] F. Afroz, S. Barua, and K. Sandrasegaran, Performance analysis of FLS, EXP-W, LOG and M-LWDF Packet Scheduling Algorithms in Downlink 3GPP LTE System, International Journal of Wireless & Mobile Networks, **6**, 77-91, (2014).

[2] F. Asadollahi and R. D. Heydari, Introduction of a Novel Hybrid Weighted Exponential Logarithm-Maximum Throughput (HWEL-MT) Scheduler for QoS Improvement of LTE/4G Cellular Networks, Wireless Personal Communication, **98**, 91-104, (2018).

[3] Akashdeep , K. S. Kahlon and H. Kumar, Survey of scheduling algorithms in IEEE 802.16 PMP networks, Egyptian Informatics Journal,**15**, 25-36, (2014).

Table 2: COMPARISON OF PERFORMANCE RANKS AMONG SCHEDULERS FOR LOW RT/NRT RATIO i.e. < 1.0

PERF METRIC / PARAMETER	EXP-W	HYB-1	M-LWDF	LOG	HYB-2
RT-Goodput	2	1	2	4	3
NRT-Goodput	3	1	2	2	1
RT-Pkt-Delay	2	1	5	4	3
NRT-Pkt-Delay	5	4	3	1	2
RT-SI	2	1	2	4	3
NRT-SI	3	1	2	2	1
RT PLR	1	1	2	4	3
Spectral Eff.	4	1	2	3	1
Avg Perf Rank	2.75	1.38	2.50	3.00	2.13
Std Devn of Rank	1.28	1.06	1.07	1.2	0.99

Table 3: COMPARISON OF PERFORMANCE RANKS AMONG SCHEDULERS FOR HIGH RT/NRT RATIO i.e. > 1.1

PERF METRIC / PARAMETER	EXP-W	HYB-1	M-LWDF	LOG	HYB-2
RT-Goodput	4	1	2	3	2
NRT-Goodput	4	3	2	1	1
RT-Pkt-Delay	5	1	4	2	3
NRT-Pkt-Delay	4	5	3	1	2
RT-SI	4	1	2	3	2
NRT-SI	4	3	2	1	1
RT PLR	4	1	1	3	2
Spectral Eff.	4	3	2	2	1
Avg Perf Rank	4.13	2.25	2.25	2.00	1.75
Std Devn of Rank	0.35	1.49	0.88	0.93	0.71

Table 4: OVERALL PERFORMANCE RANKS OF SCHEDULERS

	EXP-W	HYB-1	M-LWDF	LOG	HYB-2
OVER ALL AVG RANK	3.44	1.81	2.38	2.50	1.94
OVER ALL STD DEV	1.15	1.33	0.95	1.15	0.85

[4] F. Mustafa and P. Lohiya, Optimum Resource Allocation in Orthogonal Frequency Division Multiplexing Communication System using Fuzzy Rule Base System and Particle Swarm Optimization and Comparison with Conventional other Techniques, I.J. Wireless and Microwave Technologies,**6**, 44-52, (2015).

[5] Shubham, R. Gupta, V. Gajera and P. K. Jana, An Effective Multi-Objective Workflow Scheduling in Cloud Computing: A PSO based Approach, In Proc. Ninth International Conference on Contemporary Computing, IEEE, Noida, India, 1-6, (2016).

[6] Y. Yi, Z. Q. Yu and W. Ye, Modified Particle Swarm optimization and Genetic Algorithm based Resources Allocation algorithm for Multi-user Orthogonal Frequency Division Multiplexing system, Information Technology Journal, **10**, 955-964, (2011).

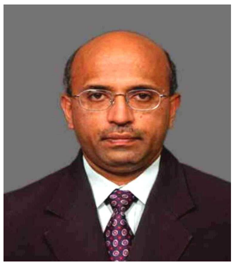
[7] G. Piro, L. A. Grieco, G. Boggia, R. Fortuna and P. Camarda, Two-Level Downlink Scheduling for Real-Time Multimedia Services in LTE Networks, IEEE Transaction on Multimedia,**13**,1052-1065, (2011).

[8] S. Strasser, R. Goodman and J. Sheppard, A new Discrete Particle Swarm Optimization Algorithm, In Proc. Genetic and Evolutionary Computation Conference, ACM Conference, Denver, USA, 53–60 (2016).

[9] K Rameshkumar and C Rajendran, A novel discrete PSO algorithm for solving job shop scheduling problem to minimize makespan, In Proc. IOP Conference Series: Materials Science and Engineering, Bangalore, **310**, p012143, (2018).

[10] O. M. Eshanta, M. Ismail, and K. Jumari, OBBP: An Efficient Burst Packing Algorithm for IEEE 802.16e Systems, International Scholarly Research Network ISRN Communications and Networking, **2011**, 1-9, (2011).

[11] WiMAX and the IEEE 802.16m Air Interface Standard, WiMAX Forum, (2010).
[12] 3GPP Technical Specifications TS 23.203 V14.0.0, 3GPP Forum, (2016).
[13] Cisco : Enterprise QoS Solution Reference Network Design Guide, Cisco, April, (2014).
[14] ITU-T Recommendation G.1010 for End-User Multimedia QOS Categories, ITU, Nov, (2001).
[15] WiMAX VoIP Solutions for 4G Networks, WMF-M14-v01, WiMAX Forum, August, (2010).
[16] Overview of IEEE P802.16m Technology and Candidate RIT for IMT-Advanced, IEEE 802.16 IMT-Advanced Evaluation Group, IEEE WirelessMAN 802.16, (2010).
[17] R. Annauth and H. Rughooputh C.S, OFDM Systems Resource Allocation using Multi-Objective Particle Swarm Optimization, International Journal of Computer Networks & Communications (IJCNC), **4**, 4, 291-306, (2012).
[18] B. Sadiq, S.J.Baek, and G. de Veciana, Delay-Optimal Opportunistic Scheduling And Approximations: The Log rule, IEEE/ACM Transactions on Networking **19**, 2, 405-418, (2010)
[19] S. Shakkottai and A. Stolyar, Scheduling for Multiple Flows Sharing a Time-Varying Channel: The Exponential Rule, Analytic Methods in Applied Probability, vol 207, American Mathematical Society, Rhode Island, USA, 185-201, (2002).
[20] M. Andrews, K. Kumaran, K. Ramanan, A. Stolyar, P. Whiting and R. Vijayakumar, Providing Quality of Service over a Shared Wireless Link. IEEE Communications Magazine, **39**, 2, 150–154, (2001).
[21] A. Sherry, K. Manikandan, K. Sistila and S. Golla, Survey on MAC Scheduling Algorithms for WiMAX(IEEE 802.16) Networks, IJARC, **7**, 5, 100-103, (2016).
[22] S. Pandey, L. Wu, S. M. Guru and R. Buyya, A Particle Swarm Optimization-based Heuristic for Scheduling Workflow Applications in Cloud Computing Environments, In Proc. 24th IEEE International Conference on Advanced Information Networking and Applications, Perth, Australia, 400-407, (2010).
[23] V.Singh and V.Sharma, Efficient and fair scheduling of uplink and downlink in IEEE 802.16 OFDMA networks, In Proc. IEEE Wireless Communications and Networking Conference, Las Vegas, USA, 984-990, April (2006).
[24] J. Lu, M. Ma , Cross-layer QoS support framework and holistic opportunistic scheduling for QoS in single carrier WiMAX system, Journal of Network and Computer Applications, **34**, 765–773, (2011).
[25] M. Settembre, M. Puleri, S. Garritano, P. Testa, R. Albanese, M. Mancini and VL Curto, Performance analysis of an efficient packet-based IEEE 802.16 MAC supporting adaptive modulation and coding, In Proc. International Symposium on computer networks, Istanbul, 11-16, (2006).
[26] D. Niyato and E. Hossain, Queue-aware uplink bandwidth allocation and rate control for polling service in IEEE 802.16 broadband wireless networks. IEEE Transactions on Mobile Computing, **5**, 6, 668–79, (2006).
[27] M. Fathi, B. Rashidi, Samaneh, K. Alireza and T. Hassan, A dynamic joint scheduling and call admission control scheme for 802.16 networks, Telecommunication Systems **52**, 195–202, (2013) .
[28] M. Rosendo and A. Pozo, Applying a Discrete Particle Swarm Optimization Algorithm to Combinatorial Problems, In Proc. Eleventh Brazilian Symposium on Neural Networks, Sao Paolo, IEEE Computer Society, 23-28 Oct. (2010).
[29] J. Kennedy and R. Eberhart, Particle Swarm Optimization, In Proc. IEEE International Conference on Neural Networks, **4**, pp. 1942-1948, Perth, Australia, (1995).



T. Sentthil Vinayakam is currently serving as Associate Professor in Coimbatore Institute of Technology, Coimbatore, India. He has an accomplished experience of 19 years in reputed industries including VSNL aka Tata Communications Ltd, 12 years of academic and 9 years of research experience. He is a resource person in the fields of Communication, Networking and RF. He has delivered over 27 invited lectures in these fields. He has published 21 research papers. He has done Artificial Intelligence based project in ISRO Trivandrum. He has reviewed chapters for two books on “Information Theory and Coding” and “Fibre Optical Communications” of renowned publishers. He is a reviewer for reputed International journals in the domain of Telecommunication, Networking and Security. He obtained his Masters degree in Applied Electronics from Bharathiar University in 1990 and Bachelors degree in ECE in 1985 from Madras University.

A. Rajeswari is currently working as Professor and Head of the Department at Coimbatore Institute of Technology, Coimbatore, India in the department of Electronics and Communication Engineering. She has an experience of 29 years in teaching and 12 years in research. She has produced six doctorates and is presently guiding twelve research scholars in the discipline of Wireless Communication and Biomedical signal processing. She obtained her doctoral degree in Information and Communication from Anna University, Chennai in 2007. She obtained her Masters degree in Communication Systems from Bharathiar University in 2002 and Bachelors degree in ECE in 1990 from Madurai Kamaraj University. Her areas of research include sensing in cognitive radio networks, Equalization techniques for OFDM systems and Biomedical signal processing.