

CADA: channel and delay aware scheduler for real-time applications in WiMAX networks

Melek OKTAY,^{1,2,*} Hacı Ali MANTAR²

¹Department of Computer Engineering, Fatih University, Büyükçekmece, İstanbul, Turkey

²Department of Computer Engineering, Gebze Institute of Technology, Gebze, Kocaeli, Turkey

Received: 04.01.2012 • Accepted: 14.06.2012 • Published Online: 02.10.2013 • Printed: 28.10.2013

Abstract: Scheduling is the core of the worldwide interoperability for microwave access (WiMAX) technology that directly affects the performance of the network. In this study, we focus on scheduling and present a novel algorithm called the channel and delay aware scheduler (CADA) for real-time applications, such as voice over Internet protocol, video-on-demand, and video streaming. CADA has 2 important modules: wireless and network delay monitoring tools. The wireless module, including the compensation and channel state monitoring modules, increases the network throughput and provides fairness among all of the flows in the network. The network delay monitoring tool calculates the estimated network delay of the paths and uses this estimated delay to calculate the packet deadline for meeting the delay requirement of the real-time application flow. The main objectives of this study are: a) fulfilling the delay requirements of the real-time applications using a network monitoring tool, b) providing opportunistic scheduling by taking the channel conditions and network delay into account, and c) surveying the recent trends of the existing scheduling algorithms in WiMAX as guidelines for researchers who are interested in the scheduling algorithms. The simulation results show that CADA improves the performance of the real-time applications by taking the delay metric into account, increasing the network throughput of the WiMAX network, and providing the long-term fairness among all of the flows.

Key words: WiMAX, channel and delay awareness, EDF, scheduling, cross-layer, QoS

1. Introduction

The IEEE 802.16 standard, worldwide interoperability for microwave access (WiMAX), is a broadband wireless access technology that provides a high data rate and ubiquitous access in a large coverage area [1,2]. Cost effectiveness and easy deployment are other important properties of this technology. WiMAX is classified as a metropolitan area network and it offers alternative access in the last mile of wired networks, such as T1/E1 links, digital subscriber lines, and cable modems.

WiMAX supports heterogeneous traffic that needs different quality of service (QoS) requirements. To satisfy the QoS of the heterogeneous traffic, many mechanisms are defined in the WiMAX standard [1–5]. For example, resource allocation, admission control, packet scheduling for uplink and downlink traffic, traffic shaping, and policing are some of them. However, the implementation details are unstandardized and are left open for vendors such as the WiMAX equipment makers and researchers [1,6,7]. The scheduling algorithm is the core of WiMAX and the performance of the scheduling mechanism directly affects the performance of the network and applications [7,8]. Therefore, scheduling is one of the most important components in WiMAX.

*Correspondence: moktay@fatih.edu.tr

In the last decade, many valuable scheduler algorithms have been developed for WiMAX [6,8,9]. The proposed schedulers can be classified into 4 main categories: first generation (legacy) schedulers, hierarchical schedulers, application optimization based schedulers, and opportunistic (cross-layer) schedulers. A comprehensive literature survey about the schedulers in WiMAX is described in the following section.

Although there are many scheduling schemes that are applied to wired networks, these schemes cannot be used directly in wireless networks, since wireless channels have problems like time varying channel behavior, signal attenuation, fading, and multipath propagation. In this study, a novel scheduling architecture, the channel and delay aware scheduler (CADA), which takes the channel conditions and network delay of the subscriber stations (SSs) into consideration for providing the QoS in WiMAX networks, is developed. The WiMAX base station (BS) monitors the channel conditions of the SSs and schedules their packets by taking the time varying channel conditions of the wireless network into account. CADA provides long-term fairness among all of the flows in the WiMAX network using a compensation mechanism. It also provides short-term fairness among good channel state flows. If a flow cannot send its packet due to a bad channel condition, CADA lets another flow send its packet in order to increase the network throughput. Later on, CADA compensates for the original flow to let it send its previous packets. Meanwhile, the scheduler does not let the substituted flow send packets for compensation.

We also present a novel method called the network delay monitoring tool (NDMT) that calculates the estimated network delay of all of the flows in a coarsely grained manner. The BS uses this estimated delay metric to satisfy the end-to-end (E2E) delay requirement of a real-time application. To the best of the authors' knowledge, this is the first study that uses the wireless channel conditions and network delay of the flows to provide a nearly optimal scheduling in the WiMAX networks.

The main contributions of this study are: a) fulfilling the delay requirements of real-time applications using the E2E delay of the network by a scalable network tomography tool that estimates the delay of the paths between the BS and destination addresses, b) providing opportunistic scheduling by taking the channel conditions into account, and c) surveying the recent trends of the existing scheduling algorithms in WiMAX as the means of guidelines for researchers who are interested in the scheduling algorithms.

The organization of the paper is as follows. In Section 2, the scheduling algorithms in WiMAX are described. The CADA architecture is presented in Section 3. The simulation environment and performance evaluation of CADA are described in Section 4. The concluding remarks are provided in Section 5.

2. Background

The selection of the appropriate scheduling algorithm is critical for the performance of WiMAX. When the scheduling algorithms are compared with each other, there are some design challenges. The most important design factors are fairness [10,11], implementation complexity, satisfying QoS parameters, and scalability. In addition to the design challenges and factors listed above, energy consumption, system throughput, delay bound, channel state information, and service degradation are significant design factors that should also be considered by the scheduler.

2.1. WiMAX basics

The WiMAX MAC is a connection-oriented protocol and it creates a connection for each service, even for the connectionless services such as Internet protocols. The BS assigns a unique 16-bit connection identifier (CID) for each connection, and during that communication process this CID is used for primary address purposes.

The WiMAX standard supports a class-based QoS that is mainly inspired from the data over cable service interface specification multiple QoS level standard [12]. The IEEE 802.16-2001 [1] defines the unsolicited grant scheme (UGS), real-time polling service (rtPS), nonreal-time service (nrtPS), and best effort (BE) classes; a fifth service that is called the extended real-time service (ertPS) has been specified in 802.16d-2004 [2] to the WiMAX standard. These 5 scheduling service classes and their definitions are listed as follows:

UGS: This scheduling service is designed for periodic fixed-size data packets. It supports constant bit rate real-time applications such as the voice over Internet protocol (VoIP) without silence suppression.

ertPS: This scheduling service class generates periodic variable-sized packets. It supports a variable bit rate (VBR) real-time applications such as the VoIP with silence suppression.

rtPS: This is designed for real-time applications that generate VBRs, such as audio/video (MPEG) streaming and video on demand (VoD).

nrtPS: This service class is designed for delay-tolerant nonreal-time applications such as the file transfer protocol (FTP) with guaranteed minimum throughput.

BE: This service class is designed for applications that do not need QoS parameters. The hypertext transfer protocol (HTTP) and e-mail are the examples of this service application.

In addition to the scheduling services, mandatory QoS parameters have been defined in the standard. The scheduling services and supported QoS parameters [1,3] are shown in Table 1.

Table 1. Mandatory QoS parameters of the scheduling services.

Services QoS parameters	UGS T1/E1 transport,	ertPS VoIP	rtPS streaming audio/video	nrtPS FTP	BE HTTP, data transfer
Tolerated jitter (TJ)	✓	✓	-	-	-
Maximum latency (ML)	✓	✓	✓	-	-
Maximum sustained traffic rate (MSTR)	✓	✓	✓	✓	✓
Minimum reserved traffic rate (MRTR)	✓	✓	✓	✓	-
Traffic priority (TP)	-	✓	✓	✓	-

2.2. Taxonomy and survey of scheduling algorithms

In the last decade, many valuable scheduler algorithms have been developed for the WiMAX networks. The proposed schedulers can be classified into 4 main categories, as shown in Figure 1.

First generation schedulers (legacy schedulers): This is the first category in the taxonomy. In the beginning, researchers apply existing legacy schedulers to WiMAX that are already used for the same purpose in different areas, such as wired networks and operating systems. The deficit round robin (DRR) and weighted RR (WRR) are actually designed for wired networks, and Cicconetti et al. [6] applied these algorithms to WiMAX. Sayenko et al. [7] preferred to use these RR-based schedulers because the complexity of the RR schedulers is low ($O(1)$) and the BS must determine the scheduling decision within 5 ms according to the WiMAX standard [1]. These schedulers are very simple and easy to implement. However, the RR-based scheduler shows poor performance in satisfying the QoS needs of the flows.

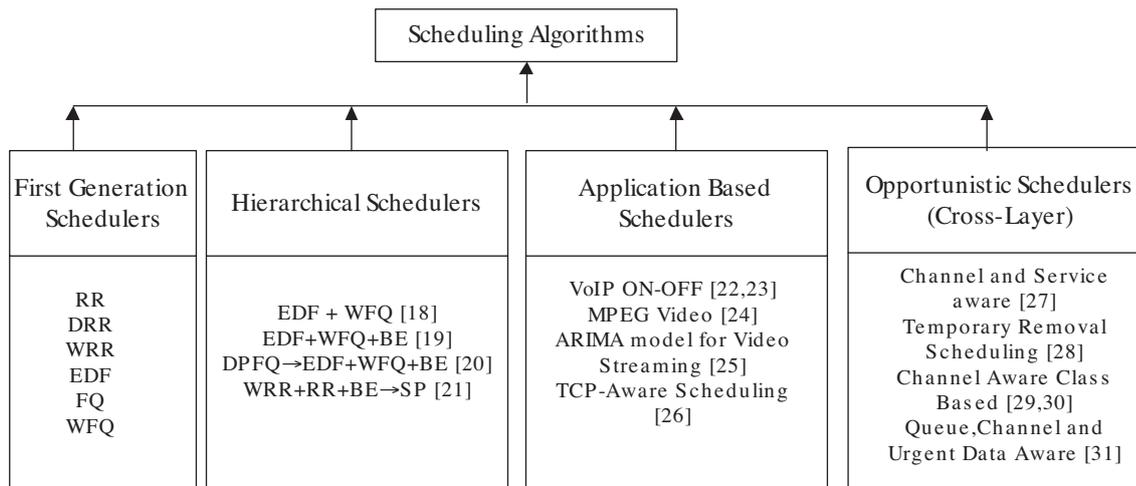


Figure 1. Classification of the WiMAX schedulers.

The weighted fair queuing (WFQ) scheduler [13] is the practical implementation of the fair queuing scheduler and it is known as an approximation of the ideal generalized processor sharing scheduler. The main drawback of the WFQ scheduler is the complexity of $O(N)$, where N is the number of flows in the system. The WFQ provides good fairness among all of the flows in the networks that have various packet sizes.

The earliest deadline first (EDF) scheduler algorithm is another important example of the legacy schedulers that are designed for real-time applications [14]. The EDF scheduler stamps each incoming packet with a deadline, which is equal to the arrival time of the incoming packet plus the latency parameter of the flow. The EDF scheduler was analyzed in [15,16] and it is a delay optimal scheduler in the deterministic environment [17]. Furthermore, it shows good performance, such as low packet loss and delay, in real-time applications in the WiMAX networks [8,18].

Hierarchical schedulers: Since the QoS needs of the scheduling classes have different requirements, researchers use hierarchical schedulers (more than one scheduler) to satisfy these QoS requirements. Wongthavarawat and Ganz [19] used the EDF scheduler for the rtPS, the WFQ scheduler for the nrtPS, and the first-in first-out (FIFO) scheduler for the BE services. Liu et al. [20] used the EDF scheduler for rtPS flows and the WFQ scheduler for nrtPSs.

The multi-stage scheduling that is used in more than one stage is also classified in this category. For instance, Chen et al. [21] proposed a scheduler that uses a deficit fair priority scheduler in the first layer to distribute the resources fairly. In the second layer, hierarchical schedulers such as EDF, WFQ, and RR are used for rtPS, nrtPS, and BE services, respectively. Settembre et al. [22] developed a 2-stage scheduler for WiMAX. In the first stage, different schedulers are used, such as the WRR for rtPS and nrtPS, and the RR for BE flows. In the second stage, strict priority scheduler assigns priority to flows and selects the ones that have the highest priority among flows.

Application based schedulers: Some schedulers are designed for improving the performance of a specific application in WiMAX networks. For instance, VoIP scheduling for WiMAX networks was studied [23,24]. Haghani et al. [25] considered MPEG video scheduling in WiMAX networks. In that study, the authors used MPEG characteristics to differentiate among MPEG frames such as I, P, and B frames. Next, more important frames are assigned a higher priority to protect their frame from dropping, such as the I frame.

Tamimi et al. [9] provided a model that was named the seasonal autoregressive integrated moving average for video streaming over WiMAX networks. This model uses a time series for forecasting future video frames' generation. Kim and Yeom [26] proposed a transmission control protocol aware uplink scheduler for WiMAX. First, the sending rate of each flow was computed and then the resource allocation was done according to the computed rate.

Cross-layer (opportunistic) schedulers: Researchers can exploit the network layers hierarchy in order to adapt the application to a nondeterministic wireless environment. This mechanism is called a cross-layer design and this idea is applied to schedulers in WiMAX. Liu et al. [27] proposed a cross-layer scheduler that assigns a priority for each flow according to its channel and service status. The highest priority flow is then scheduled. Ball et al. [28] proposed a temporary removal scheduling algorithm to WiMAX that takes a carrier to the interference plus noise ratio $C/(I + N)$ into account. This value represents the channel quality of the flow and if channel quality is not good this flow is temporarily removed from the scheduling list. Next, the channel-aware class-based scheduling algorithm was proposed by Pizzi et al. [29,30]. In order to provide long-term fairness, 2 pointers, the lead and lag, are assigned to each flow. Finally, Liang et al. [31] developed a queue and channel aware downlink scheduler (priority-based scheduler) scheme for WiMAX. Urgent and nonurgent data differentiation is also considered in this study. All of the QoS metrics, such as the queue, channel, and urgent and nonurgent data statuses are translated to the priority metric by a black-box formula in the study. The highest priority packet flow is scheduled by the WiMAX BS scheduler.

Critics: In the literature, WiMAX scheduling algorithms have been searched extensively in the last decade. Legacy schedulers are important scheduling techniques that are used for other purposes, such as computer networks and operating systems. RR-based schedulers such as the DRR and WRR have low computational complexity. However, they are not aware of channel conditions such as fading, multiuser diversity, loss rates, and power level. For this reason, they cannot be successfully adapted to a wireless medium environment. Ali et al. [8] showed that there is no single scheduler that satisfies the QoS requirements of the applications. The scheduler algorithm should be selected according to the requirements of an application. For example, if the main objective is satisfying fairness among all of the flows in the system, fair queuing-based schedulers such as the WFQ and worst-case fair weighted fair queuing (WF2Q) are usable. Moreover, if the main objective is satisfying the delay requirements of a real-time application, the EDF scheduler would be a good choice.

Due to the heterogeneous traffic such as the VoIP, the video streaming and HTTP and FTP requirements are different, and so it is a good idea to use hierarchical schedulers for fulfilling these requirements, because every scheduling service class is designed for a specific application requirement. In order to satisfy the various needs of applications, different schedulers can be used for each service class. For example, the EDF scheduler demonstrates good performance in real-time applications and it is suitable for the rtPS and ertPS real-time service classes. The WFQ or DRR schedulers would be selected for the nrtPS service for providing fairness among all of the flows. Since there is no QoS support for the BE service class, simple FIFO or RR-based schedulers are good choices in this service class as well. Hierarchical schedulers outperform the legacy schedulers in the WiMAX; however, the channel conditions are still not taken into account. These kinds of schedulers assume that the channel conditions are perfect, but the assumption of the error-free channel state is not valid for wireless networks or WiMAX.

Application-based schedulers are good for improving the performance of a specific application. The main purpose is to design the WiMAX MAC protocol according to the application characteristics. However, one application-based solution, such as the VoIP scheduler, cannot be directly applied to another application, such

as the MPEG scheduler, due to their different application characteristics. Most of the schedulers in this type do not consider the channel state behavior as well.

One of the most important drawbacks of cross-layer schedulers is their complexity. The complexity of a simple algorithm, the DRR, is incremented from $O(1)$ to $O(N)$ in wireless networks [32] due to the consideration of the channel state behavior. However, the standard defines that the BS must accomplish the scheduling within a 5-ms WiMAX duration [33]. Moreover, when cross-layer schedulers fix a certain problem, they can cause another important problem. For example, in order to increase network throughput, Liu et al. [27] provided more precedence to the flows that have better channel conditions. However, in this case, the flows that have bad channel conditions have a starvation problem.

To the best of our knowledge, all of the schedulers in the literature only take the conditions between the BS and SSs into account, except for our previous studies [18,34]. In our previous studies, the delay and hop-count statuses between the BS and the destination are considered for real-time applications. The delay of the packet that will be faced is computed by the BS and it is used in the deadline calculation of the EDF schedulers. Nevertheless, we do not take the channel conditions into account in those studies. Therefore, in order to make the nearly optimal scheduling decision in the BS, the channel conditions must also be considered. In this study, the channel awareness of the SSs is added to our previous work [18,34] and CADA is developed.

3. CADA uplink scheduler

It is stated that there is no single scheduler that satisfies all of the requirements and design factors of real-time applications. For this reason, the appropriate scheduler should be selected by the scheduler designer according to the application requirements. The EDF scheduler is the most preferable solution for real-time applications in WiMAX networks [8] due to its low delay and packet loss metrics. However, the performance of the classical EDF algorithm is not good because the network delay and channel status of the wireless network are ignored. The performance of the EDF scheduler can be improved using network and channel awareness information.

The classical EDF scheduler marks each incoming packet with a deadline and sorts them in ascending order using priority lists. Before transmission, the deadline of the packet is checked. If the deadline is missed, the packet is dropped by the scheduler. The precedence of the packet that is waiting in the EDF queue is increased in the time period. For this reason, EDF is a dynamic scheduler scheme. In addition, the analytical study of the EDF scheduler [15,16] shows that EDF is a delay-optimal scheduler in deterministic environments. However, the channel state of the wireless environment and the network delay are nondeterministic environments. The EDF scheduler should be designed by taking these important issues into account.

Assigning the deadline of the packet is a critical issue for the performance of the EDF scheduler. The deadline, d_i^p , is calculated by summing the arrival time, a_i^p , of the packet that enters the BS and the latency QoS parameter, l_i , of appropriate flow i .

$$d_i^p = a_i^p + l_i \quad (1)$$

This latency parameter is transmitted by the SS to the BS during the admission control step. Eq. (1) states that only the delay between the SS and BS, $D_{SS-to-BS}$, is considered by the EDF scheduler (Figure 2).

In order to satisfy the delay requirement of a real-time application, the E2E ($D_{End-to-BS} + D_{BS-to-SS}$) delay should be considered. The E2E delay from the SS to the destination is:

$${}^i\Gamma_{Dest}^{SS}[p] = {}^i\Gamma_{Dest}^{BS}[p] + {}^i\Gamma_{BS}^{SS}[p]. \quad (2)$$

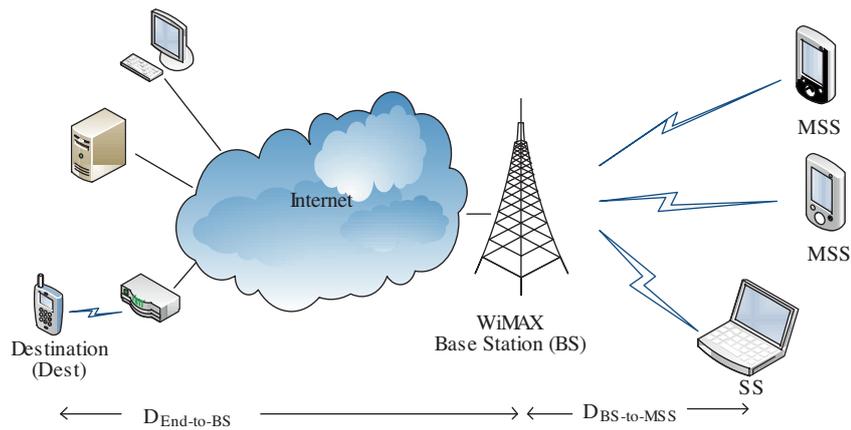


Figure 2. E2E WiMAX network architecture.

The delay of packet p in flow i from the SS to the destination is represented as ${}^i\Gamma_{Dest}^{SS}[p]$, which is equal to the E2E delay. It is calculated by summing: 1) The network delay ${}^i\Gamma_{Dest}^{BS}[p]$, which is the delay of the packet p in the same flow i from the BS to the destination. 2) The local area propagation delay ${}^i\Gamma_{BS}^{SS}[p]$ is the propagation delay of the packet p transmitted in the flow i from the SS to the BS. For a packet to be transmitted, the E2E delay constraint must be less than or equal to the maximum tolerable, ${}^i\Gamma_{MAX}[p]$, delay of the application (Eq. (3)). Otherwise, the packet is dropped.

$${}^i\Gamma_{Dest}^{SS}[p] \leq {}^i\Gamma_{Maximum}[p] \tag{3}$$

In the deadline calculation in Eq. (1), the processing delay and network delay are ignored. However, CADA considers the network delay (coarsely-grained manner) in Eq. (2). Moreover, the important network delay information of the flows is not known by the BS in advance. In this study, we provide a mechanism called the NDMT that measures the network delay without adding high complexity to the BS. The NDMT sends probe packets from the BS to destinations and measures the estimated network delay. The BS monitors the network, measures the delay of flow and the BS, and saves the delay statistics as well. We redesign the EDF scheduler according to the estimated network delay in CADA. The per-flow basis of the CADA scheduler is shown in Figure 3.

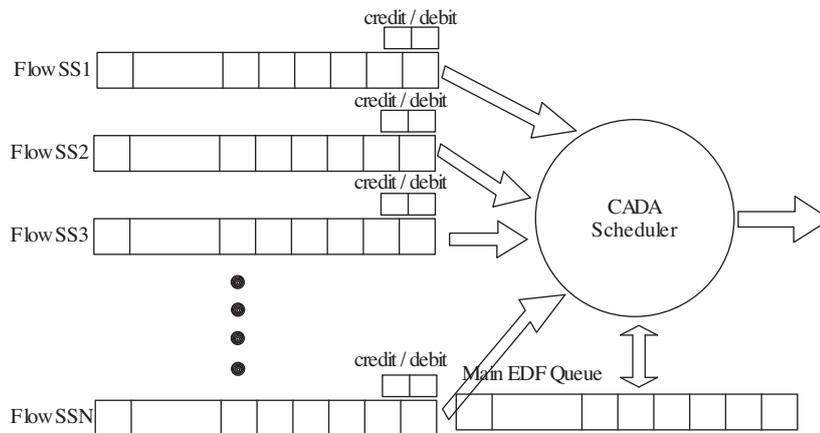


Figure 3. CADA scheduler.

After the calculation of the estimated network delay, we recalculate the deadline of a packet d_i^p in Eq. (1) using the estimated network delay ed_i^p . By subtracting the estimated delay of the packet p in the flow i from d_i^p , we calculate the new deadline, nd_i^p , as follows:

$$nd_i^p = (a_i^p + l_i) - ed_i^p. \quad (4)$$

Until now, we have focused on the delay awareness property of CADA. However, a nonideal channel condition is ignored in Eq. (2). In order to provide efficient and nearly optimal scheduling in wireless networks, the proposed EDF scheduler should consider the channel state. Channel awareness is not a new approach in the literature; however, it has not been applied to the EDF scheduler in WiMAX. In this study, we apply not only the channel awareness property but also the network delay awareness property to EDF in WiMAX networks with CADA. The CADA architecture is illustrated in Figure 4. CADA consists of 2 modules. The first one is the wireless module that includes the compensation module (CM) and the channel state monitoring module (CSMM). The second module is the NDMT and it calculates the estimated delay from the BS to the destination region.

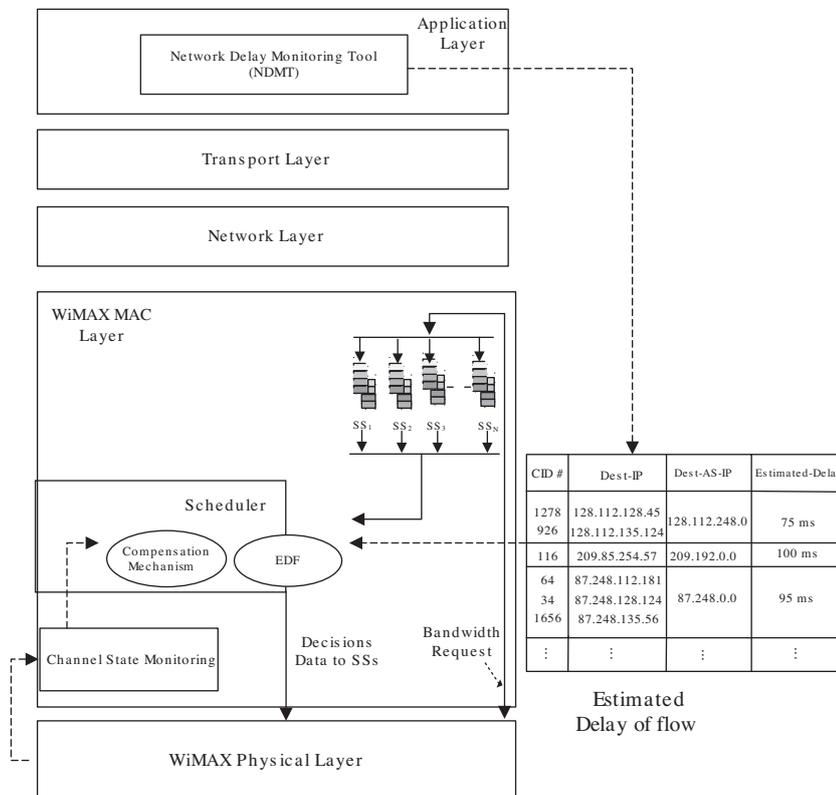


Figure 4. The proposed CADA architecture.

3.1. Wireless module

The BS not only considers packet scheduling but also takes network delay and channel error characteristics into account. If a SS flow suffers from error-prone channels like time-varying channels and location-dependent errors, the BS does not allocate resources to that SS flow in this period in order to prevent wasting the network resources. Moreover, if the flow has a bad channel state, the flow is considered as being in a lagging state (credit), which means that its resources are reassigned to error-free flows. The SS flow that obtains extra resources in

the error-free channel (good channel state) is considered as leading state (debit). A flow that allocates exact resources in its error-free channel is considered as in-sync state. This means that the flow neither gets extra resources nor loses its resources. Later on, when the flow recovers to the clean channel state conditions after the lagging state, the lost resources should be compensated for by the scheduler in a proper way. However, the recovering/compensation process should be done smoothly in order to avoid the starvation of the active flows.

This module was inspired from the channel aware system proposed by Pizzi et al. [29,30] that uses the DRR and WF2Q+ schedulers. These schedulers monitor the channel conditions. In our study, we select EDF as a packet scheduler for providing the latency requirement of the real-time application. EDF selects the head of line (HOL) packet that belongs to the flow i in the EDF queue. Next, the scheduler (in the CM) checks the channel status of flow i (the CID of packet is used) in the CSMM in order to understand whether the channel is 'good' or 'bad'. The CM is responsible for providing fairness and increasing throughput in the WiMAX network (Figure 4). For each flow, 2 dedicated pointers, which are called debit/credit (or lead/lag) pointers, are assigned in the CM. If the channel is good (error-free service), the packet is transmitted. However, if the channel is bad, the EDF scheduler selects the next packet in the queue that belongs to another unmarked flow for transmitting the packet instead of the HOL packet. If one flow substitutes another flow for poor channel conditions, the substituted flow credit counter is incremented by one. Meanwhile, the replaced flow (original flow) debit counter is incremented by one. Later on, if the original flow channel state becomes good, the scheduler lets the flow send its previous packets in the number of debit counter. Moreover, a substituted flow cannot send its packet as credit counter times in this time period. When packet cannot be transmitted due to the bad channel condition, CADA selects flow that has a maximum debit counter among existing flows. Due to the poor channel condition, a flow can become unfair for a short-term period, but with the compensation mechanism, we provide long-term fairness for each flow.

The CSMM is responsible for monitoring and reporting the SSs' channel state conditions to the scheduler. The BS regularly polls the SS to gather the bandwidth request messages and these messages include the carrier to the interference and noise ratio (CINR), which reflects the channel quality of the SS. According to the channel quality and the bit error rate (BER) level of the SS, the appropriate modulation and coding scheme (MCS) is selected by the BS for communication. If the received signal quality of the SS becomes lower than a certain threshold, the BS changes the MCS and uses the more robust MCS for the SS. The selection process of the most robust MCS can go until the BPSK 1/2, which is the most robust MCS in the standard. Later on, when channel quality of the SS becomes lower than the BPSK 1/2 BER level, the SS is banned and the channel status is considered as a 'bad' state. If there is a robust MCS scheme for the connection, the channel condition is considered as a 'good' state. According to the SS signal quality, the MCS of the connection can change with one of the MCS in the list {QPSK 1/2, QPSK 3/4, 16QAM 1/2, 16QAM 3/4, 64QAM 1/2, and 64QAM 3/4} and within this period, the channel state of the SS is considered as 'good'.

3.1.1. Simulating the channel state

The wireless environment usually consists of a large number of obstacles that result in multipath propagations and signal power attenuations. The error characteristic of the channel model is known as a Rayleigh fading channel and it can be represented by a 2-state Markov chain model, which is called a Gilbert–Elliot model [35–37]. This model is used for the error process approximation of wireless radio channels and the state changes between good (error-free) and bad (error-prone). This 2-state Markov chain model is shown in Figure 5.

Every state corresponds to a certain BER range. If the receiver sensitivity drops under the most robust

modulation threshold level, the channel is considered as bad in the Gilbert–Elliot model. The transition probability matrix (M) of the Markov process is given by:

$$M = \begin{pmatrix} 1-p & p \\ q & 1-q \end{pmatrix} \quad (5)$$

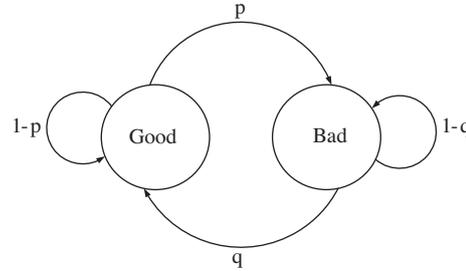


Figure 5. The 2-state Markov chain to model the change of the channel conditions of the SSs.

The transition probability of changing from a good channel state to a bad channel state is represented by $P(g|b) = p$. $P(g|g) = 1 - p$ is the transition probability of changing from a good channel state to a good channel state. The $P(b|g) = q$ is the transition probability of changing from a bad channel state to a good channel. $P(b|b) = 1 - q$ is the probability of transition from a bad channel state to a bad channel state. Next, the steady state probability of being good $\pi(g)$ and bad $\pi(b)$ states is given by:

$$\pi(g) = \frac{q}{p+q}, \quad \pi(b) = \frac{p}{p+q}. \quad (6)$$

Two assumptions are made in order to simplify the simulation of the channel model.

- The BS knows all of the SS's channel state information perfectly, i.e. the CINR status is successfully received from all of the SSs. If the SS channel quality cannot be measured, it is considered as a 'bad' state.
- The channel conditions vary on a frame-by-frame basis. In other words, the quality of the channel remains constant in the transmission of the frame.

3.2. NDMT module

The NDMT is designed for monitoring the network delay of the destination regions. The BS knows the destination IP address after the connection setup with the SS. After the BS learns the destination address, it checks the NDMT table to find the associated network delay with that IP address. The NDMT delay table can be updated using network diagnosis tools such as ping and pathchar [38]. These tools periodically send probe packets to the networks to measure the network metrics, such as delay and jitter. Sending probe packets to every possible fine-grained destination causes a scalability problem in the network, because many probe packets are sent to the network for measurement purposes. We take this issue into account in the design of the NDMT. The main idea in this module is if there are multiple flows that are approximately going to the same destination address (same autonomous system (AS)), one probe packet is enough for these links. For example, if there are multiple connections between the BS and the California State cities, we do not need to calculate the delay of each connection. Instead, we can calculate the delay between the BS and the California State AS

and use this delay for each connection. Hence, we measure the delay metric with a coarse-grained manner and we reduce the scalability problem.

However, how can we know if 2 different IP addresses are included in the same AS? The IP helps us to solve this problem. Actually, IP addresses are hierarchically distributed to ASs. ASs are also hierarchically connected to the each other and this is called AS topologies or Internet service provider (ISP) topologies. However, these real ISP topologies are not publicly available to the research community; ISPs keep their topologies confidential for commercial and security reasons. Luckily, these important ISP topologies can be deducted from periodically published information such as the border gateway protocol (BGP) tables of the ISPs and public trace route servers. Skitter [39] and Rocketfuel [40] are examples of the network topology finder tools that have been developed by researchers. They are heavyweight tools for this research, but the approximation techniques that are used by these tools are the same as in our research.

The details of the NDMT are depicted and described in Figure 6, step-by-step. First of all, the BGP table information/tuples are stored in a database. In the first step, upon receiving a connection request, the BS sends the destination IP address of the request and the CID to this module. In the second step, the longest prefix matching of the destination IP address is calculated from the database and its best-AS-IP address is returned. In the third step, the resulting IP address is checked from the delay table. If it exists in the table, the estimated delay of this AS value is returned. If it does not exist, the estimated delay of this AS is calculated by sending Internet control message protocol probe packets (like ping) to the destination and the result is saved in the table (fourth step). In the final step, the network delay that is saved as the estimated delay is returned as a result. In addition, the estimated-delay in the delay table entries are recalculated and updated periodically in order to keep the network delay up to date.

The NDMT component is installed on the BS. This component actively measures the delay of the appropriate links. Therefore, there is no need for the modification of the WiMAX protocol for this module, which is installed on the SSs.

It is important to note that the network delay in the NDMT table is not computed online. The NDMT module finds the network delays to possible destinations in advance. Once a connection request comes, it just checks delay table to find the associated delay with that IP. In other words, the delays are not computed during the packet forwarding or upon a connection request.

Even if it is rare, the NDMT may not compute the network delay for some destination regions. In such cases, the NDMT computes the network delay upon receiving a connection request. Even this does not result in a scalability problem because the computation is done per flow rather than per packet. Thus, the NDMT table updates do not affect the scheduling process.

Another important scheduling issue is that, as mentioned above, the network delay states are in a coarse-grained fashion in the NDMT. This means that the states are per region. A region can be an AS, a network, or any IP prefix (IP/X, $X < 32$). For example, similar to routing table entries, there can be a delay value for a 128.123.0.0/16 prefix. Thus, for 2^{16} different IP addresses, there is only 1 entry in the NDMT table. This is also important for the network delay computation scalability. For 2^{16} addresses, we have only 1 delay computation.

3.3. CADA report mechanism

In order to schedule the SSs' packets by taking the channel and delay awareness into account, the BS must know the SSs' exact packet deadlines. However, the packets' deadlines are not known in advance in the BS until the packet is transmitted from the SS to the BS. Therefore, a report mechanism is developed in CADA.

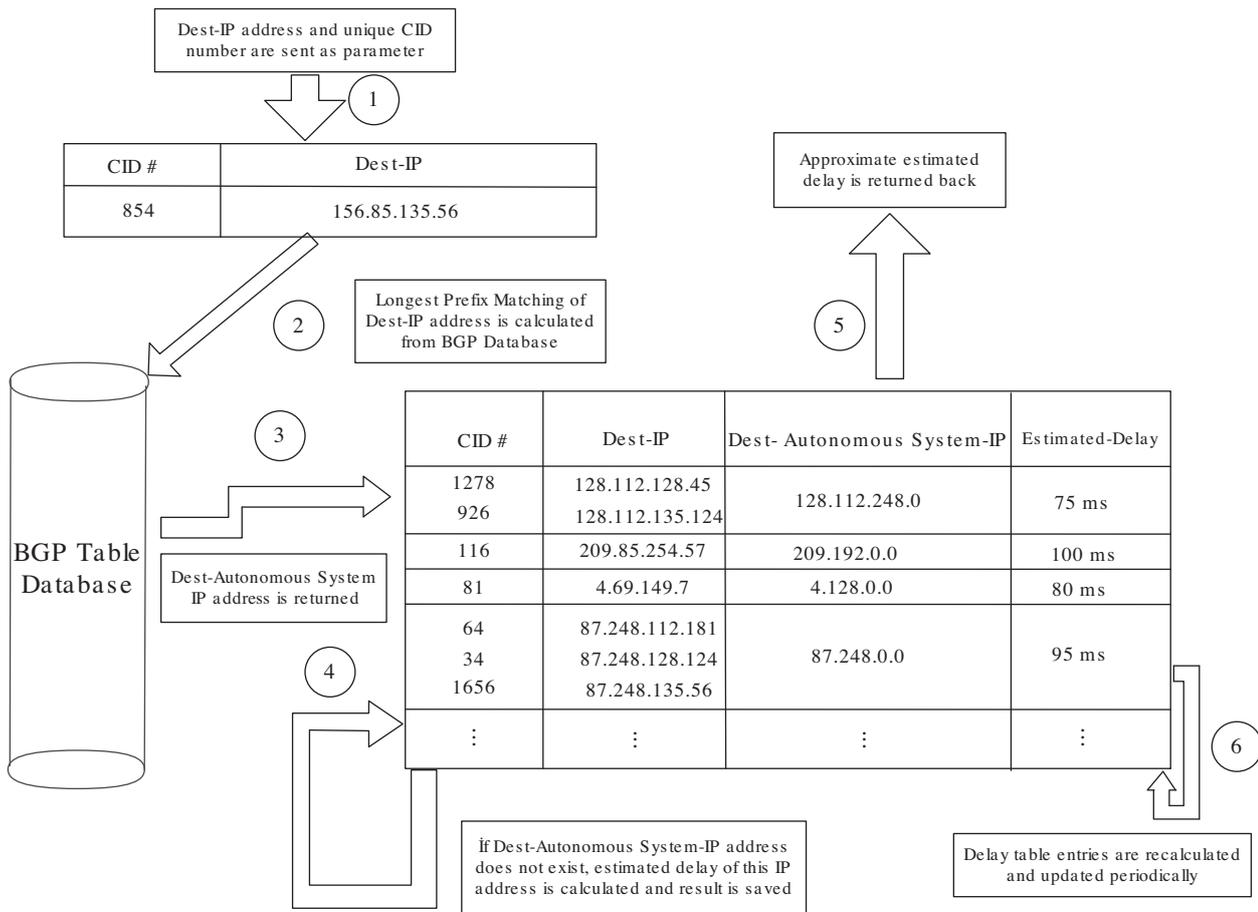


Figure 6. Network delay monitoring tool (NDMT) module.

Periodically, the SS sends its first 10 packets' deadlines to the BS using a report mechanism. The extended subheader field (ESF) is defined in the MAC header of the IEEE 802.16 (WiMAX) standard. The ESF is used for creating a number of additional subheaders within the protocol data unit, and it is applicable in both the uplink and downlink directions.

The ESF field is set to 1 and extended subheader group (ESG) will immediately follow the WiMAX MAC header. The ESG is composed of an 8-bit ESG length and 7-bit ESF header, and an extended subheader body (this field is variable; the user decides the length of this field). The CADA extended subheader is created using the ESF as shown in Table 2, to inform the BS about the packets' deadline.

Table 2. CADA extended subheader.

Name	Length	Description
Deadline of 1st packet	16 bits	Unsigned integer
Deadline of 2nd packet	16 bits	Unsigned integer
...

The packet deadline data type is *unsigned int* and it is represented by 2 bytes (16 bits) and the SS sends its first 10 packets' deadlines to the BS. If there are less than 10 packets in the queue of the SS, it fulfills the deadlines of the nonexisting packets with 0.

3.4. Complexity analysis

We compared the CADA algorithm with the EDF [14] and the EDF-based scheduling system [17,19]. In EDF, a packet deadline is computed as: $\text{deadline} = \text{arrival time} + \text{latency}$. The packets are then inserted into a sorted priority queue. This requires $O(\log(N))$ time complexity, where N is the number of flows. Processing a packet from a priority queue has a $O(1)$ time complexity.

The major concern of EDF is the complexity of inserting a packet into a queue. When the queue length is large, the cost of the packet insertion becomes a severe problem.

In CADA, the deadline computation is the same; $\text{deadline} = (\text{arrival time} + \text{latency}) - \text{estimated network delay}$. As shown, taking the network delay into account does not introduce any complexity in addition to EDF. Similar to the latency of a packet (in EDF and CADA), the network delay is known in advance. During the packet processing (forwarding), the latency and network delay are not computed. These values are taken from the NDMT table using an indexing mechanism. Thus, the complexity of CADA and EDF is the same $O(1)$ in the case of the deadline computation. The detailed explanation of the NDMT scalability is given in Section 3.2.

The next and the most important complexity issue is the packet scheduling mechanism (inserting packets into a queue and sending packets from queues). Unlike EDF [14] and EDF-based model [17,19], CADA has per-flow queuing. In per-flow queuing, the insertion complexity is $O(1)$. However, per-flow scheduling has a $O(N)$ complexity, where N is the number of flows.

The $O(N)$ complexity is an important issue, especially in the forwarding plane, where the packets need to be processed in line speed. In CADA, we use a 2-level queuing model in order to reduce the complexity. As depicted in Figure 3, CADA scheduling model has a second level queue (main queue in Figure 3). In this queue there is only one packet from each flow (each queue in the first level queue). The packets in the main queue are sorted based on their deadline. Once a packet is sent, a new packet from the appropriate queue is inserted into the second (main queue) queue. The insertion process has a $O(\log(N))$ time complexity, where N is the number of packets, and in turn, the number of flows (because there is only one packet from each queue). Thus, the CADA complexity is $O(1)$ to $O(\log(N))$. This shows that the CADA complexity is the same as that of EDF and EDF-based models.

When the number of packets in the priority queue increases, the EDF performance decreases due to the large inserting cost. Because EDF has a $O(\log(Q))$ complexity, where Q is the queue length, Q can be considered as $Q = M \times N$, where N is the number of flows and M is the average number of packets of each flow in the queue. On the other hand, CADA complexity is independent of the number of packets from each queue and it has $O(\log(N))$. When the flows perceive a bad channel because of channel errors, CADA has a $O(N)$ complexity, as it has to search for a good channel flow among the N flows. This is the worst-case scenario.

The classical EDF is not per-flow-based and the channel awareness cannot be used for the compensation mechanism. We change the EDF scheduler to per-flow-based in CADA for resilient channel errors and reduce the complexity.

4. The simulation framework and results

In order to evaluate the performance of CADA, a discrete-event simulator is developed. The proposed CADA architecture, EDF scheduler, and priority-based scheduler [31] algorithms are implemented in this simulator. The priority-based scheduler assigns a priority for each SSs according to their channel, queue, and urgent data conditions. The system parameters that are used by our simulator are given in Table 3, which represents the IEEE 802.16 standard.

Table 3. System parameters.

Parameter	Value
FFT	1024
Subchannel model	PUSC
DL/UL ratio	2/1
Frame duration	5 ms
Types of traffic	rtPS

Only the rtPS scheduling service is considered in the simulation in order to focus on real-time applications. The proposed method can be applied to other scheduling services such as ertPS and UGS as well. High-quality movie traces [41,42], encoded in different formats such as H.261, H.263, and MPEG-4, generate variable bit rate (VBR) traffic in the simulator. These video streaming parameters are listed in Table 4.

Table 4. Video streaming parameters.

Parameter	Value per SS
MRTR	128 Kbps
MSTR	521 Kbps
Mean rate	300 Kbps
Maximum latency	100–160 ms
Packet size	1300–8000 bytes

The characteristic of the wireless channel model is the Rayleigh fading channel and it is represented by a 2-state Markov chain. The channel status and quality of the SS is changed according to its remoteness to the BS. For example, if the SS is near the BS and there is no obstacle between them, the channel state probably becomes good. We used a good state transition matrix for these types of connections. There are 3 possible state transition matrices for the good, normal, and bad channel states, which are listed in Table 5. These values are taken from [31,43]. We are not focusing on how p and q can be selected and how they are changed. This is an important issue, but it is outside the scope of this paper. For this case we rely on previous studies [31,43]. However, our analysis shows that the performance of CADA is not dependent on the values of these matrices. According to the remoteness and environmental effects between the BS and SS connection, one of the transition matrices is set to the SS.

Table 5. State transition matrices.

Good channel transition matrix	Normal channel transition matrix	Bad channel transition matrix
$M = \begin{pmatrix} 0.9 & 0.1 \\ 0.8 & 0.2 \end{pmatrix}$	$M = \begin{pmatrix} 0.8 & 0.2 \\ 0.7 & 0.3 \end{pmatrix}$	$M = \begin{pmatrix} 0.3 & 0.7 \\ 0.3 & 0.7 \end{pmatrix}$

In addition to the channel state awareness, CADA supports the delay awareness for providing a better E2E delay for the SSs. The NDMT module measures the network delay (estimated delay) for possible paths and this delay is used in the deadline calculation. Hence, CADA adopts itself to the possible congestion in the network and it lets the long delay (urgent) connections send their packets as soon as possible. In the simulation, the delay of the connections varies between 100 and 160 ms. CADA considers both the channel state condition and the delay awareness of the network condition. However, EDF does not consider the channel condition and delay awareness of the network. Therefore, if the channel condition is bad, EDF cannot transmit the packet like CADA, but there is compensation in CADA.

4.1. Simulation results

Figure 7 shows the simulation topology. The BS centrally controls the communication between the SSs and their destinations for both the uplink and downlink directions. A standard point-to-multipoint configuration is used.

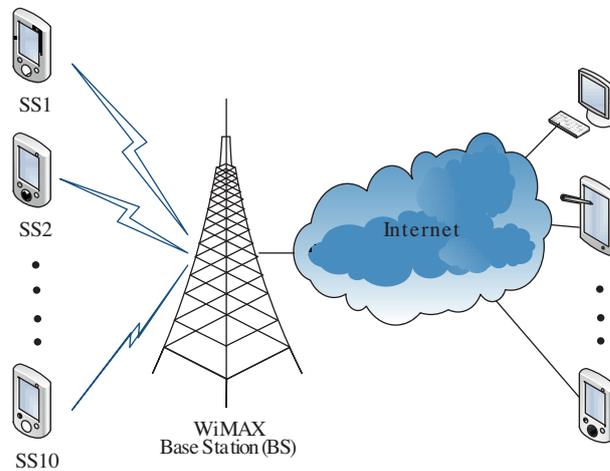


Figure 7. Simulation topology.

In the simulation, there are 10 SS nodes and they send high-quality video traces to the uplink directions. Every SS has a different channel quality according to the remoteness of the BS and it uses one of the transition matrices that are listed in Table 3. SS6, SS7, and SS8 have bad channel states in the beginning. SS4, SS5, and SS9 have good channel states. The remaining SSs have normal channel states in the simulation. In addition to the channel state information, the packets of SS1, SS2, SS7, and SS9 will face a long delay on the Internet. The packets of SS4, SS5, SS6, and SS10 are sent in short delay links on the Internet because their destination path statuses are normal and there is no congestion. The packets of SS3 and SS8 are sent in the normal delay links.

It is assumed that the BS has a perfect knowledge of the channel state information of all of the SSs. The Rayleigh channel model is simulated for each SS. It is assumed that the channel state status is fixed during the frame period. The sample simulation runs 10 times and the averages of the resulting metrics are given below:

4.1.1. Fairness index

When a resource allocation decision is made by the scheduler, it must be fair among all of the flows. Each flow in the same class should take approximately the same service. To quantify this parameter, the Jain's fairness index (JFI) [10], Eq. (7), is used, where n represents the total number of flows in the system and x_i represents total number of received throughputs of flow i .

$$JFI = \frac{(\sum_{i=1}^n x_i)^2}{n \cdot \sum_{i=1}^n x_i^2} \quad (7)$$

If the JFI index converges to 1 (the best case), all of the flows share the same amount of radio resource in the WiMAX network and the resources are shared fairly. Otherwise, if the JFI index converges $1/n$ (the worst case), where n is the number of flows, the radio resources are shared unfairly between the flows.

In the simulation, the fairness indices of these 3 scheduling schemes, CADA, EDF and priority, are compared. Figure 8 illustrates the average of 10 different simulation results of the JFI. The JFI is used for the calculation of the fairness. CADA performs better than other 2 schedulers' schemes in terms of fairness. It is observed that the JFI of CADA is close to 1, which means that CADA distributes the network throughput among the SSs more fairly than EDF and priority schedulers. Due to the channel conditions of the SSs, the short-term fairness may not be satisfied, but the long-term fairness is achieved by the compensation mechanism of CADA.

4.1.2. Average throughput

The throughput is defined as an amount of successful data transferred over the communication channel at a specified amount of time. The second comparison metric is the average throughput of the SSs. After 10 simulation runs, we calculate the average throughput of each SS and the results are shown in Figure 9. It can be easily observed that SSs with a bad channel condition (SS6, SS7, and SS8) suffer from network throughput in EDF and priority schedulers. However, these SSs get better throughput in CADA because of the compensation mechanism. In CADA, it can be easily observed from Figure 8 that the network resources are distributed among the SSs fairly. The average throughputs of the SSs are nearly the same as each other in CADA. However, EDF and priority schedulers suffer from channel conditions and SSs that have bad channel conditions, such as SS6, SS7, and SS8, do not get enough resources. The priority-based scheme performs better than EDF in terms of throughput. However, starvation, where the SS does not get resources for a long period, is a major problem for the priority-based scheduler because a SS that has a good channel condition gets a higher priority in the scheduler. The priority-based scheduler takes the queue status into account when calculating the priority for dealing with the lost resources. However, it is not as efficient as the compensation mechanism of CADA, as shown by the performance of average throughput.

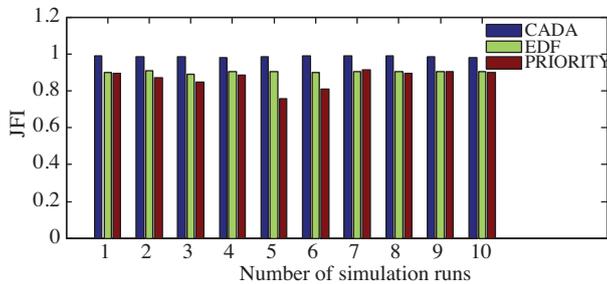


Figure 8. JFI of CADA, EDF, and priority-based schemes.

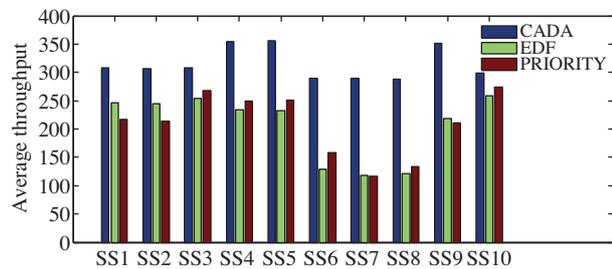


Figure 9. Average throughputs of the SSs for EDF, CADA, and priority-based schemes.

4.1.3. Average delay

The average E2E delays of the SSs for a time period are shown in Figures 10, 11, and 12. It can easily be inferred that CADA performs better than EDF and priority schedulers because of the NDMT module. Aside from better average throughputs and fairness among the SSs, CADA also provides better delay requirements (low delay), which are very significant for real-time applications using the NDMT module. The NDMT module calculates the network delay of the possible paths with a coarsely grained manner and uses this delay in the calculation of the packet deadline. Therefore, the packets that will face a long delay on the Internet will be more urgent in CADA and will be sent as soon as possible. Hence, CADA adopts itself to Internet congestions and it will be a nearly delay-optimal scheduler. The performances of EDF and priority schedulers are approximately

the same in terms of average E2E delay. The priority-based scheduler assigns a higher priority to urgent data but the deadline of the packet is not considered. In addition, network delays are not taken into account in EDF and priority schedulers.

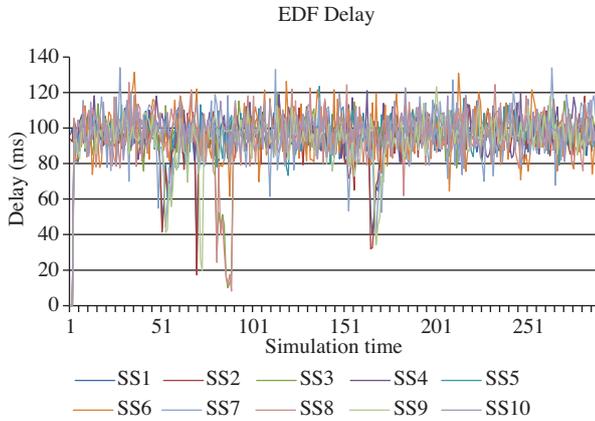


Figure 10. Average delays per SS in EDF.

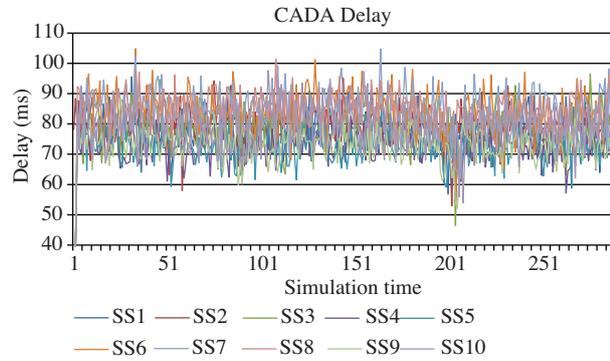


Figure 11. Average delays per SS in CADA.

4.1.4. Packet loss

When a packet travels from source to destination, it can be lost and fail to reach its destination. Moreover, if the real-time packet deadline cannot be satisfied because of a long network delay, the packet will be missed (it is discarded) and it is dropped. As a result, the packet loss ratio of the flow increases, and the performance of the real-time application is negatively affected. Figure 13 demonstrates the mean packet loss (%) rates of EDF, CADA, and priority schemes. The CADA packet loss rate is less than the others because CADA considers the estimated network delay of the SSs in the network by the NDMT module. EDF and priority schemes do not consider the network delay and as a result CADA provides a better packet loss rate (low packet loss). The priority-based scheduler outperforms EDF in terms of loss rate because the channel conditions and buffer status metrics are used in the calculation of packet priority.

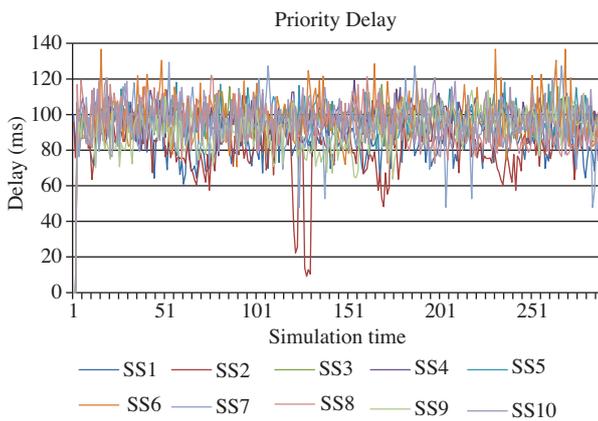


Figure 12. Average delays per SS in the priority-based scheduler.

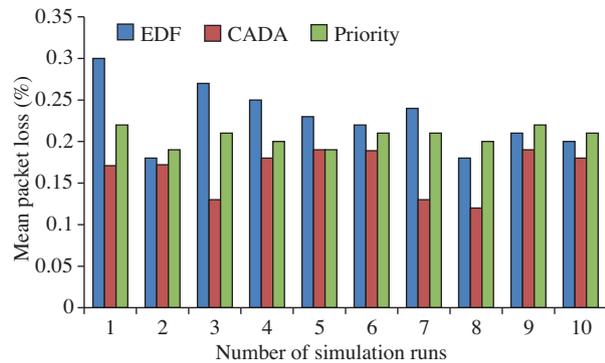


Figure 13. Mean packet loss of EDF and CADA.

4.1.5. Jitter

Jitter is defined as the statistical variance of the packets' interarrival time. The jitter J is calculated as defined in the RFC 1889 [44] formula given in Eq. (8). $D(i - 1, i)$ is the difference of the relative transit times for the 2 packets to packets $i - 1$ and i . If the delay variances of the packets fluctuate, the jitter values increase. Therefore, the performance of the real-time application is negatively affected. Figures 14, 15, and 16 illustrate that the fluctuation is more in EDF and priority schedulers than in CADA. CADA has better jitter values (low jitter) than the others because its jitter values are smoother. EDF and priority schedulers' jitter values fluctuate more than CADA's jitter values. The EDF scheduler has lower jitter values compared with the priority scheduler. The priority-based scheduler takes the buffer, channel, and urgent data into consideration and the delay variation between the packets is greater than that of EDF. As a result, the priority-based scheduler shows the worst performance in terms of jitter value.

$$J = J + (|D(i - 1, i)| - J)/16 \tag{8}$$

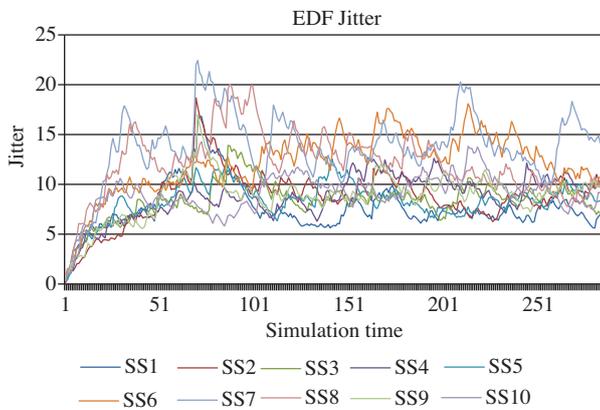


Figure 14. Jitter values of the EDF scheduler scheme.

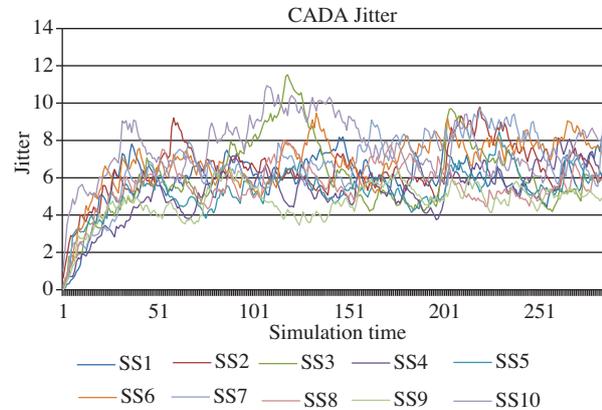


Figure 15. Jitter values of the CADA scheduler scheme.

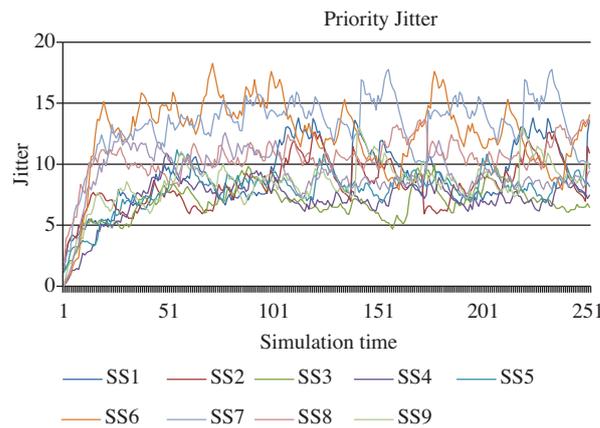


Figure 16. Jitter values of the priority-based scheduler scheme.

5. Conclusions

In this paper, a novel scheduler scheme, called CADA, is proposed to fulfill the channel and network delay awareness of the EDF scheduler algorithm. The main contribution of CADA is that it can meet the delay

requirements of real-time applications by making use of network monitoring, and it provides opportunistic scheduling that takes the channel conditions and network delay into account. These contributions are provided by 2 modules, which are wireless and the NDMT. The wireless module includes 2 important mechanisms: the CSMM and CM. The CSMM individually monitors the channel status of all of the SSs that are included in the WiMAX network and reports them to the scheduler component. According to the SS channel quality, the CSMM considers its channel status as a ‘good’ or ‘bad’ state. Next, CM considers the throughput loss of the SS due to the channel condition and improves the fairness between all of the SSs that are included in the same service class. The CM assigns 2 dedicated pointers for each flow, the lead and lag pointers (credit and debit), and according to the channel status and transmitting options these pointer values change. When the state of the channel condition becomes good, the previous lost service is compensated for by this module.

The NDMT module is another important module that is developed in CADA. The NDMT calculates the network delay (estimated delay) of the possible paths in a coarsely grained manner. The proposed scheme uses this network delay in the calculation of the real deadline of the packets.

CADA, EDF, and priority-based schedulers are evaluated with an extensive simulation. The numerical simulation results (fairness, throughput, delay, and packet loss) validate that CADA outperforms the other 2 schedulers. According to the delay performance metric, CADA provides a low delay to the SSs. In addition, CADA fairly distributes the network resources among the SSs and increases the network throughput by taking the nonideal condition of the wireless medium into account. In addition, the jitter performance metric shows that CADA does not affect the jitter of the real-time application. Last but not least, CADA provides a lower packet loss rate. The simulation results show that the channel and delay awareness approach is effective and CADA adopts itself to a nondeterministic network delay and stochastic channel condition.

References

- [1] IEEE 802.16-2001, “IEEE standard for local and metropolitan area network – Part 16: Air interface for fixed broadband wireless access systems”, 2002.
- [2] IEEE 802.16d-2004, “IEEE standard for local and metropolitan area networks – Part 16: Air interface for fixed broadband wireless access systems”, October 2004.
- [3] IEEE 802.16e-2005, “IEEE standard for local and metropolitan area networks – Part 16: Air interface for fixed and mobile broadband wireless access systems amendment 2: physical and medium access control layers for combined fixed and mobile operation in licensed bands”, February 2006.
- [4] IEEE 802.16j-2009, “IEEE standard for local and metropolitan area networks – Part 16: Air interface for fixed and mobile broadband wireless access systems – multihop relay specification”, May 2009.
- [5] IEEE 802.16m-2010, “IEEE standard for local and metropolitan area networks – Part 16: Air interface for broadband wireless access systems amendment 3: advanced air interface”, 2011.
- [6] C. Cicconetti, L. Lenzini, E. Mingozzi, “Quality of service support in IEEE802.16 networks”, *IEEE Network*, Vol. 20, pp. 50–55, 2006.
- [7] A. Sayenko, O. Alanen, T. Hämäläinen, “Scheduling solution for the IEEE 802.16 base station”. *Computer Network*, Vol. 52, pp. 96–115, 2008.
- [8] N.A. Ali, P. Dhrona, H. Hassanein, “A performance study of uplink scheduling algorithms in point-to-multipoint WiMAX networks”, *Computer Communicaiton*, Vol. 32, pp. 511–521, 2009.
- [9] A.K. Al Tamimi, S. Chakchai, R. Jain, “Modeling and Resource Allocation for Mobile Video over WiMAX Broadband Wireless”, *IEEE Journal on Selected Areas in Communications*, Vol. 28, pp. 354–365, 2010.

- [10] R. Jain, D. Chiu, W. Hawe, "A quantitative measure of fairness and discrimination for resource allocation in shared computer systems", Digital Equipment Corporation Technical Report, 1984.
- [11] M. Mehrjoo, M. Dianati, X.S. Shen, K. Naik, "Opportunistic fair scheduling for the downlink of IEEE 802.16 wireless metropolitan area networks", Proceedings of the International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks, 2006.
- [12] Data over Cable Service Interface Specification (DOCSIS 3.0), Third generation transmission systems for interactive cable television services – IP cable modems, ITU-T Recommendation J.222.
- [13] A. Demers, S. Keskav, S. Shenker, "Analysis and simulation of a fair queuing algorithm", Proceedings of the Symposium Proceedings on Communications Architectures and Protocols, ACM, Vol. 19, pp. 1–12, 1989.
- [14] C.L. Liu, J.W. Layland, "Scheduling algorithms for multiprogramming in a hard-real-time environment", Journal of the ACM, Vol. 20, pp. 46–61, 1973.
- [15] V. Sivaramana, F. M. Chiussi, "Statistical analysis of delay bound violations at an earliest deadline first (EDF) scheduler", Elsevier Performance Evaluation, Vol. 36-37, pp. 457 - 470, 1999.
- [16] V. Sivaraman, F. M. Chiussi, "End-to-end statistical delay guarantees using earliest deadline first (EDF) packet scheduling", Proceedings of the Global Telecommunications Conference, IEEE, Vol. 2, pp. 1307–1312 1999.
- [17] L. Georgiadis, R. Gurin, A. Parekh, "Optimal multiplexing on a single link: Delay and buffer requirements", IEEE Transactions on Information Theory, Vol. 43 pp. 1518–1535, 1994.
- [18] M. Oktay, H.A. Mantar, "A real-time scheduling architecture for IEEE 802.16 — WiMAX systems", Proceedings of the IEEE Applied Machine Intelligence and Informatics (SAMi), pp. 189–194, 2011.
- [19] K. Wongthavarawat, A. Ganz, "Packet scheduling for QoS support in IEEE 802.16 broadband wireless access systems", International Journal of Communication Systems, Vol. 16, pp. 81–96, 2003.
- [20] N. Liu, X. Li, C. Pei, B. Yang, "Delay character of novel architecture for IEEE 802.16 systems", Proceedings of the Parallel and Distributed Computing, Applications and Technologies, pp. 293–296, 2005.
- [21] J. Chen, W. Jiao, H. Wang, "A service flow management strategy for IEEE 802.16 broadband wireless access systems in TDD mode", Proceedings of the International Conference Communications, IEEE, Vol. 5 pp. 3422–3426, 2005.
- [22] M. Settembre, M. Puleri, S. Garritano, P. Testa, R. Albanese, M. Mancini, V.L. Curto, "Performance analysis of an efficient packet-based IEEE 802.16 MAC supporting adaptive modulation and coding", Proceedings of the IEEE International Symposium on Computer Networks, pp. 11–16, 2006.
- [23] D. Zhao, X. Shen, "Performance of packet voice transmission using IEEE 802.16 protocol", IEEE Wireless Communications, Vol. 14, pp. 44–51, 2007.
- [24] E. Haghani, N. Ansari, "VoIP traffic scheduling in WiMAX networks", Proceedings of the Global Telecommunications Conference, IEEE, pp. 1–5, 2008.
- [25] E. Haghani, S. Parekh, D. Calin, E. Kim, N. Ansari, "A quality-driven cross-layer solution for MPEG video streaming over WiMAX networks", IEEE Transactions on Multimedia, Vol. 11, pp. 1140–1147, 2009.
- [26] S. Kim, I. Yeom, "TCP-aware uplink scheduling for IEEE 802.16", IEEE Communications Letters, Vol. 11, pp. 146–148, 2007.
- [27] Q. Liu, X. Wang, G.B. Giannakis, "A cross-layer scheduling algorithm with QoS support in wireless networks", IEEE Transactions on Vehicular Technology, Vol. 55, pp. 839–847, 2006.
- [28] C.F. Ball, F. Trembl, X. Gaube, A. Klein, "Performance analysis of temporary removal scheduling applied to mobile WiMAX scenarios in tight frequency reuse", Proceedings of the IEEE Personal, Indoor and Mobile Radio Communications, Vol. 2, pp. 888–894, 2005.
- [29] S. Pizzi, A. Molinaro, A. Iera, "Channel-aware class-based scheduling for QoS support in IEEE 802.16/WiMAX networks", Proceedings of IEEE Information Networking , pp. 1–5, 2009.

- [30] A. Lera, A. Molinaro, S. Pizzi, “Channel-aware scheduling for QoS and fairness provisioning in IEEE 802.16/WiMAX broadband wireless access systems”, *Network*, Vol. 21, pp. 34–41, 2007.
- [31] J.M. Liang, J.J. Chen, Y.C. Wang, Y.C. Tseng, B.S. Lin, “Priority-based scheduling algorithm for downlink traffics in IEEE 802.16 networks”, *IEEE Asia-Pacific Wireless Com. Symposium*, 2009.
- [32] H. Fattah, C. Leung, “An efficient scheduling algorithm for packet cellular networks”, *Proceedings of the IEEE Vehicular Technology Conference*, Vol. 4, pp. 2419–2423, 2002.
- [33] IEEE P802.16Rev2/D2, “Draft standard for local and metropolitan area networks – Part: 16 Air interface for broadband wireless access systems”, pp. 2094, 2007.
- [34] M. Oktay, H.A. Mantar, “A Distance-Aware Scheduler for Real-Time Applications in 802.16 WiMAX Networks”, *Proceedings of the IEEE Networked Digital Technologies*, pp. 435–440, 2009.
- [35] E.N. Gilbert, “Capacity of a burst-noise channel”, *Bell System Technical Journal*, Vol. 39, pp. 1253–1265, 1960.
- [36] E.O. Elliot, “Estimate of error rates for codes on burst-noise channels”, *Bell System Technical Journal*, Vol. 42, pp. 1977–1997, 1963.
- [37] H.S. Wang, P.C. Chang, “On verifying the first-order Markovian assumption for a Rayleigh fading channel model”, *Proceedings of the Universal Personal Communications*, pp. 160–164, 1994.
- [38] A.B. Downey, “Using pathchar to estimate Internet link characteristics”, *Proceedings of the Applications, Technologies, Architectures, and Protocols for Computer Communication*, Vol. 29, pp. 241–250, 1999.
- [39] Skitter, <http://www.caida.org/tools/measurement/skitter/>, 2012.
- [40] N. Spring, R. Mahajan, D. Wetherall, T. Anderson, “Measuring ISP topologies with Rocketfuel”, *IEEE/ACM Transactions on Networking*, Vol. 12, pp. 2–16, 2004.
- [41] F.H.P. Fitzek, M. Reisslein, “MPEG-4 and H.263 video traces for network performance evaluation”, *Network*, IEEE, Vol. 15 pp. 40–54, 2001.
- [42] Video Trace Files, <http://trace.eas.asu.edu/TRACE/ltvt.html>, 2012.
- [43] Z. Yumei, S. Yu, “Scheduling algorithm with quality of service support in IEEE 802.16 networks”, *China Unicom, National Engineering Lab*, 2009.
- [44] RFC 1889, <http://www.ietf.org/rfc/rfc1889.txt>, 2012.