

Using Simulcast and Scalable Video Coding to Efficiently Control Channel Switching Delay in Mobile TV Broadcast Networks

CHENG-HSIN HSU and MOHAMED HEFEEDA, Simon Fraser University

Many mobile TV standards dictate using energy saving schemes to increase the viewing time on mobile devices, since mobile receivers are battery powered. The most common scheme for saving energy is to make the base station broadcast the video data of a TV channel in bursts with a bit-rate much higher than the encoding rate of the video stream, which enables mobile devices to turn off their radio frequency circuits when not receiving bursts. Broadcasting TV channels in bursts, however, increases channel switching delay. The switching delay is important, because long and variable switching delays are annoying to users and may turn them away from the mobile TV service. In this article, we first analyze the burst broadcasting scheme currently used in many deployed mobile TV networks, and we show that it is not efficient in terms of controlling the channel switching delay. We then propose new schemes to guarantee that a given maximum switching delay is not exceeded *and* that the energy consumption of mobile devices is minimized. We prove the correctness of the proposed schemes and analytically analyze the achieved energy saving. We also use scalable video coding to generalize the proposed schemes in order to support mobile devices with heterogeneous resources. We implement the proposed schemes in a mobile TV testbed to show their practicality and to validate our theoretical analysis. The experimental results show that the proposed schemes: (i) significantly increase the energy saving achieved on mobile devices: up to 95% saving is observed, and (ii) support both homogeneous and heterogeneous mobile devices.

Categories and Subject Descriptors: C.2.1 [Computer-Communication Networks]: Network Architecture and Design

General Terms: Design

Additional Key Words and Phrases: Mobile TV, DVB-H, channel switching delay, broadcast networks, video simulcast, energy saving, scalable video coding

ACM Reference Format:

Hsu, C.-H. and Hefeeda, M. 2011. Using simulcast and scalable video coding to efficiently control channel switching delay in mobile TV broadcast networks. *ACM Trans. Multimedia Comput. Commun. Appl.* 7, 2, Article 8 (February 2011), 29 pages. DOI = 10.1145/1925101.1925103 <http://doi.acm.org/10.1145/1925101.1925103>

1. INTRODUCTION

Mobile TV trials or commercial services have been deployed in more than 40 countries [DVB-H 2009]. These services enable subscribers to watch their favorite TV shows and sports events while traveling. The number of mobile TV subscribers is expected to be as high as 500 million by 2011 [EU DVB-H

This work was supported in part by the Natural Sciences and Engineering Research Council (NSERC) of Canada and in part by the British Columbia Innovation Council.

Authors' addresses: C.-H. Hsu and M. Hefeeda (corresponding author), School of Computing Science, Simon Fraser University, 250-13450 102nd Ave., Surrey, B. C., Canada V3T 0A3; email: mhefeeda@cs.sfu.ca.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission and/or a fee. Permissions may be requested from Publications Dept., ACM, Inc., 2 Penn Plaza, Suite 701, New York, NY 10121-0701 USA, fax +1 (212) 869-0481, or permissions@acm.org.

© 2011 ACM 1551-6857/2011/02-ART8 \$10.00

DOI 10.1145/1925101.1925103 <http://doi.acm.org/10.1145/1925101.1925103>

2008]. Mobile devices have stringent battery capacity and heat dissipation requirements. Thus, they cannot accommodate mobile TV chips that consume too much energy. For instance, the DVB-H standard [ETSI 2007] states that, with current battery and semiconductor technologies, TV signal receivers with power consumption higher than 100 mW cannot be integrated with handheld devices. However, even the state-of-the-art prototype mobile TV chips consume about 200 mW [Iizuka et al. 2007], while commercial chips consume more than 400 mW [DiBcom 2007] in continuous mode. Therefore, the energy consumption of mobile devices is critical, and many standards, such as DVB-H (Digital Video Broadcast-Handheld) [May 2004; Kornfeld and May 2007; ETSI 2004] and MediaFLO (Forward Link Only) [FLO Overview 2009], *dictate* using energy saving schemes to increase the viewing time on mobile devices. The typical scheme for saving energy is to make the base station broadcast the video data of a TV channel in *bursts* at a bit-rate much higher than the encoding rate of the video stream. Mobile devices, therefore, can receive a burst of traffic and turn off their Radio Frequency (RF) circuits until the next burst. This is called *time slicing*. Previous works in the literature show that time slicing can result in high energy savings for mobile devices [Yang et al. 2004; ETSI 2007]. For example, the DVB-H standard documents [ETSI 2007] mention that the power consumption of the receiving units of mobile devices could be reduced from about 500 mW to 50 mW because of time slicing.

Although time slicing enables mobile devices to save energy, it increases the channel switching delay, which is the time that a user waits before s/he starts viewing a selected channel when a change of channel is requested by that user. The switching delay is an important performance metric, because many users quickly flip through several TV channels before they decide on watching specific ones. Long and variable switching delays are annoying to users and may turn them away from the mobile TV service. Therefore, operators of mobile TV networks have to maintain low and constant switching delays. Unfortunately, as we will show in Section 3.2, current mobile TV networks may incur a prohibitively long channel switching delay. In this article, we consider the problem of controlling the switching delay in mobile TV networks that employ time slicing to save energy. Our goal is to provide a guarantee on the maximum switching delay from a TV channel to any other channel *without* sacrificing the energy saving for mobile devices.

In particular, the contributions of this article can be summarized as follows.

- We analyze the time slicing scheme currently used in many deployed mobile TV networks, and we show that it is not efficient in terms of minimizing the energy consumption for mobile devices, especially when short channel switching delays are required.
- We propose three new time slicing schemes that ensure that a given maximum switching delay is not exceeded, while at the same time the energy consumption of mobile devices is minimized. The new time slicing schemes employ simulcasting of TV channels with and without scalable video coding techniques.
- We prove the correctness of the proposed schemes and derive closed-form equations for the achieved energy saving. We numerically analyze the performance of the proposed schemes and provide guidelines on choosing the most suitable time slicing scheme for a given mobile TV network.
- We generalize the proposed time slicing schemes, and present two new schemes to support mobile devices with heterogeneous resources. The new time slicing schemes allow mobile devices with limited resources to skip data that cannot be rendered on them, so that more energy can be saved in addition to that saved by time slicing. Supporting heterogeneous mobile devices is critical for a mobile TV service, because some mobile devices may not be capable to render TV channels at full quality, and these mobile devices could be denied from the service. This in turn imposes negative impact on the network operator's revenues.

—We implement the proposed schemes in a mobile TV testbed. Our experimental results validate our theoretical analysis and show that the proposed schemes indeed meet the target channel switching delays and achieve significant energy savings for mobile devices: up to 95% energy saving is observed. Our experiments also show that the proposed time slicing schemes can support both homogeneous and heterogeneous mobile devices.

We note that receivers in mobile TV *broadcast* networks have separate RF circuits and antennas for processing TV signals other than the circuits for receiving and making phone calls. Our work focuses only on optimizing the energy saving for TV signal receivers. In addition, because of the one-way nature of the broadcast networks, feedback channels from numerous receivers to the base station are not practical. Thus, many of the energy saving techniques designed for video streaming to general wireless devices are not readily applicable in mobile TV networks. For example, the throttling technique proposed by Tan et al. [2007], which enables a wireless receiver to indirectly control the sending pattern of an Internet streaming server, requires a feedback channel from receiver to server, which may not be possible in mobile TV networks.

The rest of this article is organized as follows. In Section 2, we provide a brief background on mobile TV networks and summarize the related works. In Section 3, we first describe and formally state the problem of bounding the channel switching delay. Then, we analyze the performance of the solution currently used in many deployed mobile TV networks to solve this problem. We also present an overview of our proposed solutions. In Section 4, we present the details and the formal analysis of our solutions. In Section 5, we present the evaluation of our solutions in a mobile TV testbed, and we conclude the article in Section 6.

2. BACKGROUND AND RELATED WORK

2.1 Brief Background on Mobile TV Networks

There are several standards for mobile TV networks, including T-DMB (Terrestrial-Digital Multimedia Broadcasting) [Cho et al. 2007], ISDB-T (Integrated Services Digital Broadcasting-Terrestrial) [Takada and Saito 2006], DVB-H [Faria et al. 2006; Kornfeld and May 2007], and MediaFLO [FLO Overview 2009]. A brief overview of each follows. T-DMB [Cho et al. 2007] is an extension of the DAB (Digital Audio Broadcast) standard [ETSI 2001] to add video broadcast services to the audio services offered by DAB. The extension includes both source coding, such as MPEG-4/AVC encoding, and channel coding, such as Reed-Solomon coding. The development of T-DMB is supported by the South Korean government. ISDB-T [Takada and Saito 2006] is a digital video broadcast standard defined in Japan for both fixed and mobile TV receivers. ISDB-T divides its bandwidth into 13 segments, where 12 of them are used for fixed HDTV and one is for mobile TV programs.

We present an overview of the DVB-H standard defining some concepts that will be used later in the article. As shown in Figure 1, the DVB-H standard defines protocols below the network layer and uses IP as the interface with higher-layer protocols such as UDP and RTP. Standards such as IP Datacast [May 2004; Kornfeld and May 2007] complement DVB-H by defining a set of higher-level protocols for a complete end-to-end solution. DVB-H encapsulates IP packets using Multi-Protocol Encapsulation (MPE) sections to form MPEG-2 transport streams. Thus, data from a specific TV channel form a sequence of MPEs. MPEs are optionally FEC-protected (Forward Error Correction) before being transmitted over the wireless medium for reliable transmission. FEC is important to mobile TV networks because mobile devices often suffer severe wireless signal degradations, while no feedback channel is available for data retransmission. To save the energy of mobile devices, MPEs belonging to a given TV channel are transmitted in bursts. Thus, mobile devices can receive a burst of traffic and then turn off their RF circuits (or put them in sleep mode) until the next burst. The multiprotocol encapsulation,

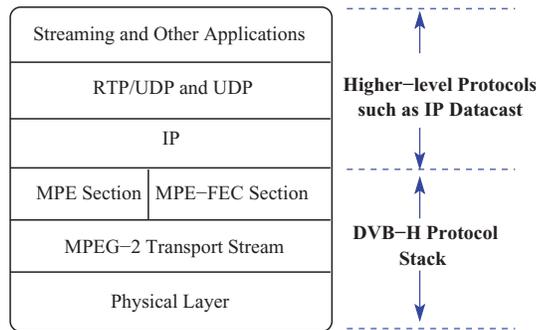


Fig. 1. Network protocol stack defined in DVB-H and IP Datacast.

FEC, and time slicing are done in a network device called IP encapsulator, which is a component of the base station broadcasting TV channels to mobile devices. The broadcast bandwidth of a DVB-H network depends on the employed network parameters such as the modulation scheme and the channel code rate, and can be as high as 31.67 Mbps. In this article, we propose efficient time slicing schemes to control channel switching delays. Hence, our schemes would be implemented in the IP encapsulator of the base station.

MediaFLO [FLO Overview 2009] is a proprietary mobile TV system developed by Qualcomm and the FLO forum [FLO Forum 2008]. MediaFLO base stations transmit signals in a superframe structure, where each superframe consists of four frames and each frame has 250 OFDM (Orthogonal Frequency Division Multiplexing) symbols. A TV channel is assigned several OFDM symbols in every frame, and this assignment defines a period of time that mobile devices must turn on their RF circuits to receive data. Therefore, MediaFLO base stations prepare transmission bursts in the form of assigning OFDM symbols to individual TV channels, and have to prepare transmission bursts in time slicing manner similar to DVB-H base stations [Daoud 2007]. Among the dedicated broadcast networks, only DVB-H and MediaFLO try to minimize the energy consumption of mobile devices by periodically turning their RF circuits off [DVB-H FAQ 2008]. We use the open DVB-H standard in our discussion throughout the article. We abstract away the specific details of the DVB-H standard in our problem formulation and solutions. Therefore, our solutions are also applicable in MediaFLO networks.

2.2 Related Work

Channel switching delay is composed of several parts, in which frame refresh delay and time slicing delay are the two dominating contributors [Rezaei et al. 2007b, 2008]. The frame refresh delay refers to the time period between receiving the first bit of a new video stream and receiving the next random access point, typically an intracoded frame, of that video. The time slicing delay refers to the time period between locking onto a mobile TV signal and reaching the first burst of the selected TV channel. Existing switching delay reduction solutions in the literature can be roughly categorized into three classes: solutions that use an auxiliary cellular network [Ollikainen and Peng 2006], solutions that reduce frame refresh delays [Vadakital et al. 2007; Rezaei et al. 2007a, 2007b, 2008, 2006a, 2006b] and solutions that reduce time slicing delays [ETSI 2007; Vadakital et al. 2007]. We briefly survey each of them in the following.

Ollikainen and Peng [2006] propose a vertical handover approach for DVB-H networks, where each mobile device maintains a unicast connection to the base station over a Universal Mobile Telecommunications Systems (UMTS) network. Video streams are not only carried by the DVB-H network but also relayed by the cellular network, which enables a mobile TV device to quickly handoff to the unicast

connection when the DVB-H signal is degraded. This auxiliary network can also be used for reducing channel switching delays. Unfortunately, maintaining a unicast connection for each mobile device for video traffic imposes tremendous load on cellular networks and streaming servers, and therefore is not scalable. In contrast, our solution requires no additional connections over the cellular network and is simpler and more scalable.

To reduce frame refresh delays, Vadakital et al. [2007] propose to periodically add redundant intracoded frames into video streams coded by H.264/AVC. In H.264/AVC, redundant frames are only decoded when primary, normal frames are not decodable. By frequently adding low-quality intracoded redundant frames into a video stream, more random access points are added, which in turn reduces refresh delays. Instead of sending low-quality intracoded frames over dedicated channels, intracoded frames can be dynamically inserted at the beginning of every burst by the IP encapsulator to shorten frame refresh delays [Rezaei et al. 2007b, 2008, 2006a, 2006b]. This is done by sending two coded video streams, primary and auxiliary, from the video streaming server to the IP encapsulator over a local wired network. The IP encapsulator substitutes an intracoded frame (on the auxiliary stream) for an intercoded frame (on the primary stream) that is about to be encapsulated as the first frame in a burst. Since every burst starts with an intracoded frame, a mobile device can decode immediately after receiving the first frame in each burst. The works on minimizing refresh delays are orthogonal to our work, and they can be combined with our work on reducing the time slicing delay.

The closest works to ours are those that try to reduce the time slicing delay [ETSI 2007; Vadakital et al. 2007]. The DVB-H standard suggests to employ parallel elementary streams for channel switching delay reduction [ETSI 2007], in which network operators bundle several TV channels into a channel group. The channel group is then encapsulated into a series of time slicing bursts. Mobile devices which want to receive any of these TV channels will have to process all bursts of the group, despite not all the data is useful to them. In addition, parallel elementary streams cannot reduce channel switching delay if the selected TV channel is not in the same channel group as the current TV channel. Therefore, determining the channel grouping strategy itself is a difficult question. In contrast, our proposed schemes provide guaranteed switching delays between any two arbitrary channels, and do not need any heuristics to group TV channels. Since the channel grouping strategy is not specified in ETSI [2007], we cannot compare our work against the parallel elementary stream method. Even if the grouping strategy is known, it would be an unfair comparison (in favor of our schemes), because our schemes are designed to minimize energy consumption, while the parallel elementary streams method results in suboptimal energy consumption since mobile devices spend more time and energy on receiving irrelevant data. Hence, we compare against the time slicing scheme where each mobile device *only* receives the video data of its TV channel. This is referred to as the current time slicing scheme in this article, because it is currently used in deployed networks.

Vadakital et al. [2007] consider the problem of interleaving C versions of a video stream to minimize the time slicing delay for C classes of mobile devices, where class- c mobile devices can decode any stream version i ($i \leq c$). They show that arranging the bursts of these versions maximally apart from each other minimizes the time slicing delay. Unlike their work, which does not bound the time slicing delay, we propose a systematic way to meet the (controllable) delay requirement. More importantly, their work fixes the inter-burst durations of different stream versions, hence allocates too many bursts for versions at low bit rates, which results in poor energy savings. In contrast, our allocation schemes are provably optimal in terms of energy saving. We do not compare our schemes against their work in this article.

Finally, we note that a preliminary version of this article was presented at the MMCN'09 conference in Hsu and Hefeeda [2009].

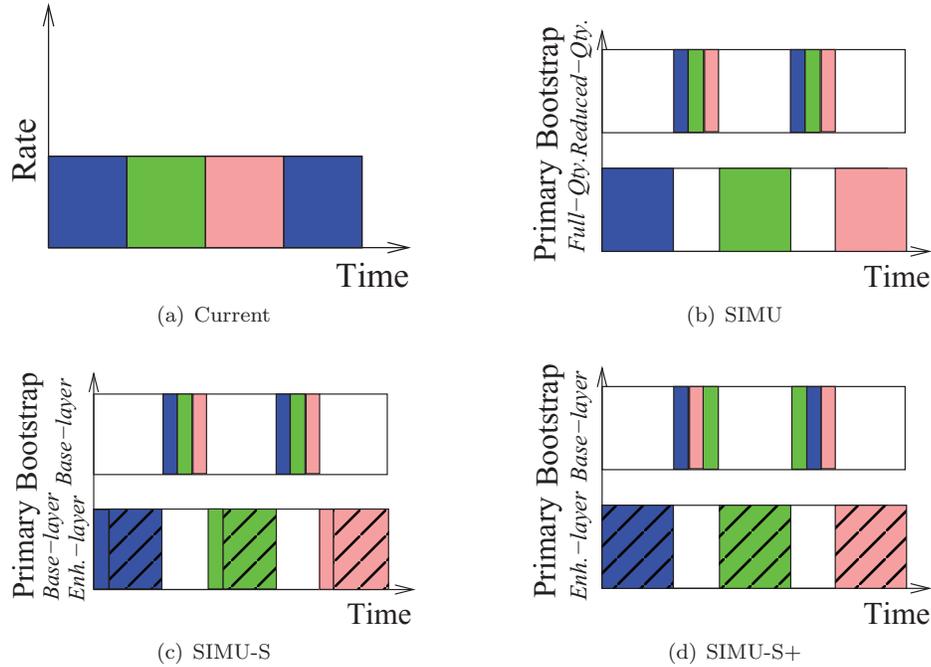


Fig. 2. Time slicing schemes in mobile TV networks. Bursts of three TV channels are shown with different colors. (a) current scheme, and (b)–(d) proposed schemes, which use simulcast with and without scalable video coding. For scalable video streams, hatched areas represent the enhancement layers.

3. PROBLEM STATEMENT AND SOLUTION APPROACH

In this section, we formally state the problem addressed in this article, and we analyze the performance of the solution currently used in many deployed mobile TV networks. We also present an overview of our proposed solutions and discuss the advantages and disadvantages of each.

3.1 Problem Statement

We consider a mobile TV network in which a base station concurrently broadcasts several digital TV channels to a large number of mobile TV devices over a shared air medium with bandwidth R kbps. Each TV channel is allocated a bit-rate r kbps, which is much smaller than R . The base station broadcasts each TV channel in a series of *bursts* at bit-rate R kbps, which is referred to as a *burst train*. The burst size is denoted by b kb. A mobile TV device receives a burst of data and turns off its RF circuit until the next burst of the same burst train. This is called time slicing and as mentioned in Section 1 while it saves the energy of the mobile devices, it may increase the channel switching delay. Figure 2(a) illustrates a simple example in which a base station broadcasts bursts of three TV channels (shown with different colors in the figure). Mobile devices tuned-in for TV channel 1 (dark bursts) turn their RF circuits off during the transmission of the other bursts. If a user watching TV channel 1 decides to switch to channel 3, the user will have to wait until the next burst of channel 3 is broadcast by the base station. This is referred to as the time slicing delay. In addition to the time slicing delay, the user will have to wait for other delays before viewing channel 3, such as the delay until the first intracoded video frame arrives (known as frame refresh delay) and the time to decode and render the stream. These delays combined are referred to as the channel switching delay, which

is denoted by d . Since the time slicing delay is the major component of the channel switching delay, we focus in this article on designing efficient time slicing schemes to control the time slicing delay. We assume that other delays are fixed, and we do not consider them anymore in this work.

We notice that the relative start time of each burst is recorded in the header of its predecessor burst such that the receivers know when they need to wake up to receive data [May 2004; Kornfeld and May 2007; ETSI 2004]. As the start time is sent in relative form, its accuracy is not affected by any constant delays between the base station and its receivers. However, the start time is sensitive to the clock jitter caused by the inaccuracy of the timers of mobile devices. To cope with this, mobile devices have to wake up their RF circuits slightly earlier to absorb the clock inaccuracy, which is referred to as *delay jitter*, and it is in the order of 10 msec [ETSI 2007]. Furthermore, waking up the RF circuits is not instantaneous, because it takes some time to lock to the frequency and synchronize to the symbols before data can be demodulated. This is called *resynchronization time*, and is in the range of 50–250 msec with current DVB-H technology [Kornfeld and May 2007; ETSI 2007; Faria et al. 2006]. For example, the Philips DVB-H chip has a resynchronization time of 150 msec [Philips DVB-H Chip 2006]. We define the *overhead duration* as the amount of time the RF circuits of mobile TV devices must be turned on before the burst time for successful demodulation, which is the sum of the delay jitter and the resynchronization time. We denote the overhead duration by T_o . In Section 3.3, we empirically measure the T_o value of a recent Nokia cellular phone, which is about 100 msec. Therefore, we use $T_o = 100$ msec in our analysis that follows, if not otherwise specified.

The energy saved by a mobile device because of the time slicing scheme is denoted by γ , and it is calculated as the ratio of time the RF circuit is in off mode to the total time [Yang et al. 2004; ETSI 2007]. Since TV channels have the same bit-rate r , their bursts have the same size and are periodically broadcast by the base station. Thus, the energy saving γ achieved by any mobile device is the same regardless of the specific TV channel being received by that device.

With the preceding definitions and notations, we are ready to state the problem of bounding the channel switching delay.

Problem 1 (Bounding Channel Switching Delay). Consider a mobile TV broadcast network with air medium bandwidth R kbps shared among multiple TV channels, and a maximum allowed channel switching delay of d_m sec. Design an optimal time slicing scheme such that the channel switching delay from any channel to any other channel is at most d_m , and the energy consumption of mobile devices is minimized. The time slicing scheme must specify the start time and the size of each burst of data for each TV channel.

The previous problem is important for the success of mobile TV services, and it has direct impacts on the profitability of service providers. This is because high switching delays may drive subscribers away from mobile TV services. In addition, this problem maximizes the energy saving for mobile devices, thus stretches the possible viewing time for mobile users and allows them to consume more TV content. Therefore, optimally solving the problem is beneficial for both operators and users of mobile TV services.

3.2 Limitations of the Current Time Slicing Scheme

Current mobile TV networks implement simple time slicing schemes. For example, the scheme proposed in the DVB-H standard documents [ETSI 2007, pp. 66] provides schedules for one TV channel: it allocates a new burst only after the data of its preceding burst is consumed by the player at the receiver. This means that bursts of all TV channels will be of the same size b , because the TV channels are encoded at the same bit-rate r . While the current time slicing scheme simplifies the design of the base station, it may lead to long switching delays and/or waste the energy of mobile devices. We analyze

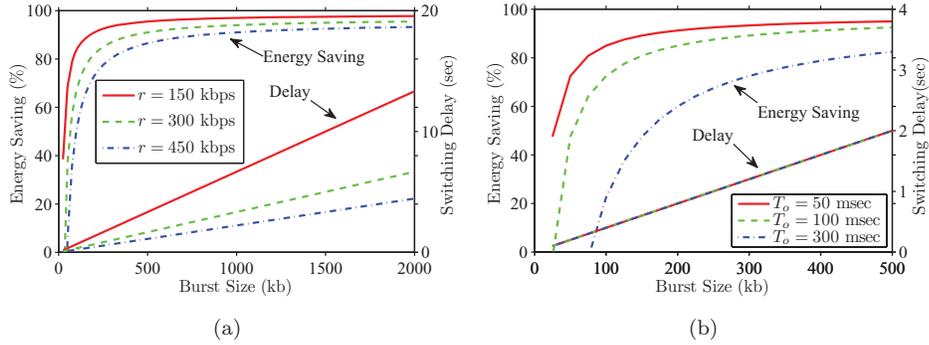


Fig. 3. The trade-off between energy saving and switching delay with different: (a) video bit rates, and (b) overhead values.

the channel switching delay d and energy saving γ of this scheme in the following. Since mobile TV networks allocate periodical bursts to keep receivers in a steady state without buffer over/underflow, the interburst period between two adjacent bursts of the same TV channel should be computed so that the number of received bits is equal to the number of consumed bits during that period. Given that each burst has a size b and the TV channel bit-rate is r , the interburst period should be b/r to avoid buffer over/underflow. Thus, the worst-case channel switching delay is $d = b/r$.

For the energy saving, consider a burst period b/r between two adjacent bursts. Since we need to transmit b kb over a shared medium with R kbps bandwidth, the burst duration is b/R sec. We also need to take into consideration the overhead duration T_o , during which the RF circuits of the receivers have to be open to search for and lock on the radio signal. Thus, we have

$$\gamma = 1 - \frac{b/R + T_o}{b/r} = 1 - \frac{r}{R} - \frac{T_o r}{b}, \quad (1)$$

where $b/R + T_o$ is the on-time of the RF circuit and b/r is the total time. Eq. (1) reveals an important trade-off between the channel switching delay and the energy saving as both d and γ increase when b increases. To illustrate this trade-off, we vary the burst size b between 10 kb and 2000 kb and compute the energy saving γ and the channel switching delay d . We set $R = 10$ Mbps and $T_o = 100$ msec. Figure 3(a) plots γ and d versus b for three sample TV channel bit rates: 150, 300, 450 kbps. Notice that the figure has two y-axes. This figure shows that the channel switching delay can be as high as 13 seconds for TV channels at low bit rates.

The current time slicing scheme can ensure a given maximum switching delay d_m by scaling down the burst size, that is, by setting $b = rd_m$. To examine the implication of scaling b down, we plot the trade-off again with different overhead T_o values in Figure 3(b), where r is fixed at 250 kbps. This figure shows that there is a dramatic decrease in energy saving when a small switching delay is desired for all practical T_o values. For example, for $T_o = 100$ msec, reducing the channel switching delay from 1.5 to 0.25 sec results in reducing the energy saving from 90% to 55%. This shows that reducing the switching delay by reducing the amount of data transmitted in each burst is not efficient in terms of energy saving, and a better solution for Problem 1 is needed.

3.3 Inferring Overhead Duration of Mobile Devices

We design and conduct an experiment to infer the *actual* overhead duration T_o of a Nokia N96 cellular phone. We chose Nokia N96 because it is the most recent Nokia phone that supports mobile TV at the time of writing. Hence, its T_o value can serve as a *lower bound* of other legacy, older mobile devices. In order to infer the T_o value of Nokia N96, we need to know the power consumption of its DVB-H chip

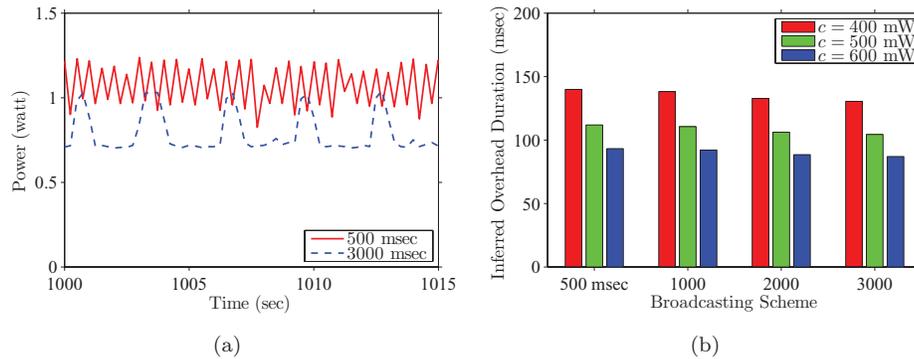


Fig. 4. (a) Sample power consumption results captured on a Nokia N96 cellular phone, and (b) overhead duration inferred from different broadcast schemes and chip specifications.

before applying the time slicing technique, which is denoted by c mW. Unfortunately, the make and model of the DVB-H chip used by Nokia N96 are not public. Therefore, we consider the data sheet released by a popular DVB-H chip manufacturer [DiBcom 2007], which indicates the first generation of DVB-H chips have a power consumption of 700 mW, while the most recent ones have a power consumption of 400 mW. We consider $c = 400, 500,$ and 600 mW in the experiment.

We conduct the experiment using a real mobile TV testbed that implements a DVB-H network. Details of this testbed are given in Section 5.1. We capture a 10-min news clip over a digital cable service, and we encode the video using an H.264/AVC encoder at bit-rate 450 kbps, and the audio using an eAAC+ encoder at 32 kbps. We use our mobile TV base station to broadcast the coded video stream, and we restart the stream upon the end of it is reached. We broadcast the video stream with a fixed inter-burst time period of 250 msec for 3.5 hrs. We cannot conduct longer experiments, because the battery of N96 only lasts for 3.5 hrs in some test scenarios. We use a Nokia N96 to watch this TV channel, and we measure its energy consumption using a built-in battery discharge monitor, called Juice. Juice is a lightweight monitor that runs in the background, measures battery voltage, current, as well as power consumption, and saves the results into a log file. A recent study reports that Juice is fairly accurate [Creus and Kuulusa 2007]. Upon getting the log file for this 3.5-hr experiment, we fully charge the cellular phone battery. We then broadcast the same video with a different inter-burst time period. We consider five different broadcasting schemes with inter-burst periods: 250, 500, 1000, 2000, and 3000 msec, and we collect a log file for each of them.

We plot two sample curves of N96 power consumption in Figure 4(a); curves for other broadcasting schemes are similar and not shown for brevity. This figure illustrates that broadcasting schemes with shorter inter-burst time periods result in higher power consumptions, while schemes with longer inter-burst time periods lead to longer sleep time between bursts. Moreover, these log files allow us to compute the average power consumption of each broadcast scheme over the 3.5-hr broadcast time. Next, we draw an important observation: the power consumption difference between any two of the broadcasting schemes can be completely attributed to the number of overhead durations T_o they impose. This is because we broadcast exactly the same video stream using each of these broadcast schemes, so that the energy consumed by receiving, decoding, rendering, and displaying the video are the same for all of them. That is, if we use the 250 msec broadcast scheme as the baseline, we can derive the T_o value for each broadcast scheme using the energy consumption difference between them. As an illustrative example, consider a 1-sec period of time, the energy consumption difference between the 250 and 500 msec broadcast schemes is due to the *two extra overhead durations* of the 250 msec scheme, compared

to the 500 msec scheme. Therefore, given a DVB-H chip energy consumption c , we can infer the T_o based on these two schemes. Similarly, we can infer the T_o by comparing 200 msec scheme with 1000, 2000, and 3000 msec schemes. We report the inference results in Figure 4(b). This figure reveals that, even with modern cellular phones such as Nokia N96, the overhead duration T_o is about 100 msec, which is nontrivial compared to burst lengths.

3.4 Overview of the Proposed Solutions

To achieve both low switching delays and high energy savings, we propose to *simulcast* each TV channel over two burst trains. Simulcast refers to a strategy that simultaneously broadcasts two versions of the same TV channel, but at different bit rates. We design new time slicing schemes such that one burst train is optimized for energy savings (referred to as the *primary* train), and the other burst train is optimized for switching delays (referred to as the *bootstrap* train). A mobile device that just switches to a new TV channel tunes to the bootstrap train first, which enables the device to start the playout of video very quickly. The device tunes to the primary train upon the next primary burst of its TV channel is transmitted. Switching over to the primary train enables the device to save more energy. Devices that are receiving the bootstrap trains are referred to as in the bootstrap stage, and devices that are receiving the primary trains are considered in the steady stage.

We further propose and analyze two methods for simulcast: (i) simulcast with traditional non-scalable video coding (referred to as SIMU), and (ii) simulcast with scalable video coding (referred to as SIMU-S). In its simplest form, SIMU broadcasts the same video stream twice with different burst sizes over the bootstrap and primary burst trains. This, however, reduces the effective utilization of the air medium as fewer TV channels can be broadcast. To mitigate this problem, a reduced-quality video stream can be transmitted over the bootstrap burst train, while the full-quality stream is transmitted over the primary burst train. Since mobile TV receivers only playout bursts from bootstrap trains for a short time period (order of seconds), showing a lower-quality video in the bootstrap stage is not very noticeable. In Section 5.2, we experimentally show through encoding multiple video sequences using an H.264/SVC coder at reduced bit rates that the quality is reasonable. More importantly, users who quickly flip through TV channels can barely realize that the video quality is reduced, because it takes human eyes some time to detect the visual details in a new scene/channel. Finally, the video quality in the bootstrap stage is a *configurable* parameter, and can be determined by service providers. Figure 2(b) illustrates the SIMU scheme. The SIMU scheme, while effective in delay reduction and energy saving, has two drawbacks. First, for each video, two separate streams need to be compressed and stored, which imposes higher costs on computational power and storage space. Second, switching from the bootstrap stream to the primary stream introduces complexities such as stream synchronization and reference frame management.

To address these drawbacks, SIMU-S employs scalable video coding, where a video stream is encoded into a base layer and one or more enhancement layers. SIMU-S transmits the reduced-quality stream (base layer) over the bootstrap burst train, and the full-quality stream (both base and enhancement layers) over the primary burst train. SIMU-S requires lower computational power and smaller storage space, simplifies video stream switching procedure, and allows gradual quality transition from reduced- to full-quality videos. Figure 2(c) illustrates the SIMU-S scheme. While SIMU-S addresses the storage and multiple compression problems, it still incurs replication over the air medium. As shown in Figure 2(c), the base layers are broadcast over both the primary and bootstrap trains. This replication may not be desirable in broadcast networks with limited bandwidth. To handle this case, we propose to broadcast the base layer only over the bootstrap burst train and the enhancement layer over the primary burst train. We call this scheme SIMU-S+, and it is shown in Figure 2(d).

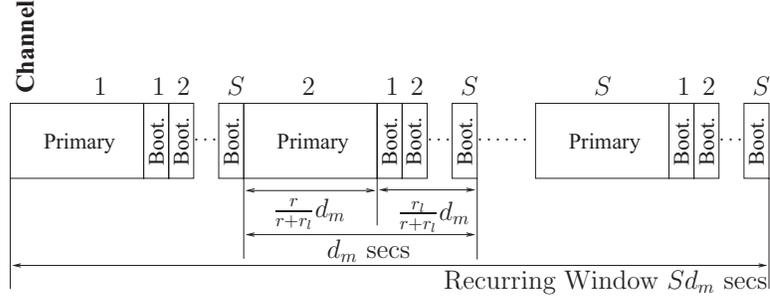


Fig. 5. Burst allocation for the SIMU and SIMU-S schemes.

As we will show in the next section, all three proposed schemes (SIMU, SIMU-S, and SIMU-S+) can provide a guarantee on channel switching delay, that is, they solve Problem 1. However, they have different advantages and are suitable in different environments. SIMU is useful in legacy environments where scalable video decoders are not widely available to mobile devices. SIMU-S is more suitable for mobile TV networks with some idle bandwidth, while SIMU-S+ is for bandwidth saturated mobile TV networks. We note that SIMU and SIMU-S result in much better energy savings than SIMU-S+, but lower bandwidth utilization.

4. DETAILS OF THE PROPOSED SOLUTIONS FOR BOUNDING SWITCHING DELAYS

In this section, we present the detailed design of the proposed time slicing schemes. We also derive closed-form equations for the achieved energy saving.

4.1 Time Slicing with Simulcast (SIMU)

As shown in Figure 2(b), SIMU transmits two versions of each video stream: one over the primary burst train with bit-rate r , and another over the bootstrap burst train with bit-rate r_l , where $r_l \leq r$. We propose an optimal (in terms of energy consumption) time slicing scheme that specifies the transmission time and the size of each burst in the primary and bootstrap trains. This time slicing scheme is illustrated in Figure 5. The basic idea is to divide the time into recurring windows of size Sd_m each, where S is the number of TV channels and d_m is the maximum allowed switching delay. Within this window, only one (large) primary burst of each TV channel is transmitted. This burst has just enough video data to be played back until the next primary burst of the same TV channel. The primary bursts of TV channel s ($s = 1, 2, \dots, S$) are allocated as

$$\langle (s-1)d_m, Sd_m r \rangle, \quad (2)$$

where the first element is the burst start time in seconds and the second element is the burst size in kb. Allocating the primary bursts according to Eq. (2) minimizes the energy consumption in the steady stages (as shown shortly in Theorem 1), but it does not ensure that the switching delay is bounded by d_m . To do so, the inter-burst period between two bursts of the same TV channel in the bootstrap train must be at most d_m . Thus in a window of size Sd_m , we need S bootstrap bursts for each TV channel, which are equally spaced with distance d_m . The bootstrap bursts of TV channel s ($s = 1, 2, \dots, S$) are allocated as

$$\left\langle (k-1)d_m + \frac{r}{r+r_l}d_m + (s-1)\frac{r_l}{r+r_l}\frac{d_m}{S}, d_m r_l \right\rangle, \quad k = 1, 2, \dots, S, \quad (3)$$

where the first element is again the burst start time, and the second is its size. The following theorem shows that the aforesaid allocation for primary and bootstrap bursts is valid and optimal.

THEOREM 1. *For simulcasting with non-scalable coding in mobile TV networks, allocating the primary and bootstrap bursts according to Eqs. (2) and (3), respectively, yields a valid time slicing scheme (i.e., with no buffer over/underflow for any TV channel), meets the switching delay constraint d_m , and maximizes the energy saving for mobile devices. Moreover, the energy saving is given by*

$$\gamma_b = 1 - \frac{r_l}{R} - \frac{T_o}{d_m}, \quad (4)$$

and

$$\gamma_p = 1 - \frac{r}{R} - \frac{T_o}{Sd_m}, \quad (5)$$

where γ_b and γ_p are energy savings for devices in bootstrap and primary (steady) stages, respectively.

PROOF. Consider a mobile device receiving an arbitrary TV channel out of the S channels. If the mobile device is in the primary stage, it gets $Sd_m r$ kb video data in every recurring time window of length Sd_m sec. This is equivalent to streaming at rate $(Sd_m r)/(Sd_m)$ kbps, which is equal to the full-quality video at bit-rate r . If the mobile device is in the bootstrap stage, it gets $d_m r_l$ kb video data at interval of d_m as there are S bursts assigned to each TV channel in each recurring window of length Sd_m sec. This is equivalent to streaming at rate $d_m r_l/d_m$, which is equal to the reduced-quality video at bit-rate r_l . Since the received bit rates are equal to the consumed bit rates, our allocation leads to no buffer over/underflow and is a valid time slicing scheme.

Next, consider any two adjacent bootstrap bursts (k and $k + 1$) for a TV channel s . According to Eq. (3), the difference in the start time between them is d_m , as $(k - 1)d_m$ is the only term that depends on k . Therefore, for a channel switching event occurring at an arbitrary time, there is a bootstrap burst within at most d_m sec. This shows that our time slicing scheme meets the switching delay constraint.

To show that our burst allocation scheme in SIMU minimizes energy consumption, we first show that any scheme that does not use simulcasting (i.e., broadcasts a *single* version of each video at bit-rate r) will result in higher energy consumption than ours. To guarantee a maximum switching delay of d_m , any such allocation scheme *must* place two successive bursts of the same video no further apart than d_m sec. Therefore, it can achieve energy saving no better than $1 - \frac{r}{R} - \frac{T_o}{d_m}$, which is always worse than the energy saving of our scheme $\gamma_p = 1 - \frac{r}{R} - \frac{T_o}{Sd_m}$ (γ_p is derived later in this proof). In addition, simulcasting schemes with more than two versions of the video will always consume more energy than our scheme (which has only two versions). This is because more versions of each video results in smaller (thus more) bursts and incurs more overhead duration periods T_o .

Now we show that our allocation scheme minimizes energy consumption by contradiction. Since we consider a steady system where the receiving number of bits equals the consumed number of bits, any valid time slicing scheme with the same recurring window size consumes the same amount of energy on receiving video data. Therefore, what differentiates the energy consumption of one time slicing scheme from another is the number of overhead duration T_o periods incurred in receiving the data. This enables us to reduce our problem of minimizing energy consumption to minimizing the number of bursts and hence minimizing the number of overhead periods incurred. Assuming there exists a valid time slicing scheme with recurring window $w_1 > Sd_m$ and the same number of primary bursts as our scheme, that is, the new scheme results in better energy consumption. To meet the delay constraint, the new scheme *must* allocate bootstraps burst for every TV channel at time intervals d_m . Moreover, to keep the number of primary bursts the same as our scheme, the new scheme can only assign one burst

for each TV channel. Since the burst bit rate is R , it takes the new scheme $\frac{rw_1}{R} + \frac{r_l w_1}{R}$ sec to complete one primary burst and S bootstrap bursts. Since $S = \frac{R}{r+r_l}$, manipulating the preceding equation leads to

$$\frac{rw_1}{R} + \frac{r_l w_1}{R} = (r+r_l)\frac{w_1}{R} > \frac{(r+r_l)Sd_m}{R} > d_m.$$

This contradicts the assumption that the new scheme is a valid allocation. Next, assume there is a valid time slicing scheme with recurring window $w_2 < Sd_m$ that produces higher energy savings than our scheme. Both this new scheme and our scheme consist of S primary bursts, while the new scheme has to allocate w_2/d_m bootstrap bursts for each TV channel to meet the switching delay constraint. Consider each burst imposes a fixed overhead duration, the new scheme always results in worse energy savings than our scheme in the steady stage because the overhead ST_o is averaged over a shorter time period $w_2 < Sd_m$. Meanwhile, the energy saving in bootstrap state is the same in both schemes. This completes our proof of optimality (in terms of energy saving).

Last, we derive the energy saving in both steady and bootstrap stages. Consider a recurring window of Sd_m . In the bootstrap stage, S bursts is assigned to each TV channel where the aggregate burst length is $d_m r_l S/R$. Following the definition of energy saving, we have $\gamma_b = 1 - \frac{d_m r_l S/R + ST_o}{Sd_m}$. In the steady stage, a burst with length $Sd_m r/R$ is assigned to each TV channel. Thus, we have $\gamma_s = 1 - \frac{Sd_m r/R + T_o}{Sd_m}$. \square

The previous theorem gives the energy savings for mobile devices in bootstrap and primary stages, and reveals that mobile devices in primary stage achieve higher energy savings than those in bootstrap stage. Therefore, the energy saving achieved by a user is determined by the average time period that user would continuously watch a TV channel, which is referred to as average *watch time* w in this article. The w value captures the channel switching behavior of a user and allows us to derive the average energy saving γ for that user. For a user with $w \geq Sd_m$, the average energy saving can be computed by taking the weighted sum of energy savings in bootstrap and primary stages, where the bootstrap stage lasts for $Sd_m/2$ sec on average. Hence, we write the energy saving for a user with watch time w as

$$\begin{aligned} \gamma &= \frac{Sd_m/2}{w} \left(1 - \frac{r_l}{R} - \frac{T_o}{d_m} \right) + \frac{w - Sd_m/2}{w} \left(1 - \frac{r}{R} - \frac{T_o}{Sd_m} \right) \\ &= \frac{Sd_m}{2w} \left[\frac{r - r_l}{R} - \frac{(S-1)T_o}{Sd_m} \right] + \left(1 - \frac{r}{R} - \frac{T_o}{Sd_m} \right). \end{aligned} \quad (6)$$

This equation shows that the energy saving γ approaches γ_p when the watch time w is large. This is because mobile devices receive primary bursts most of the time. Moreover, we observe that $\gamma = \gamma_b$ when $w < Sd_m$, as mobile devices do not get the chance to switch to primary burst trains at all.

4.2 Time Slicing with Simulcast and Scalable Coding (SIMU-S and SUMU-S+)

We next present our SIMU-S scheme, which adopts scalable video coding to save computational complexity and storage space and to reduce stream switching complexity. As shown in Figure 2(c), SIMU-S encodes each TV channel in two layers: base and enhancement layers. We provide a simple mapping between SIMU-S and SIMU which will enable us to use the time slicing scheme presented in the previous subsection for SIMU-S. This mapping can easily be done by encoding each TV channel at base-layer bit-rate $r_b = r_l$ kbps and enhancement-layer bit-rate $r_e = r - r_b$ kbps. SIMU-S then streams the base layer over bootstrap bursts and the complete stream (both base and enhancement layers) over primary bursts using this mapping. Theorem 1 is also applicable to SIMU-S.

Next, we develop our SIMU-S+ scheme. Since bandwidth saturated networks cannot afford the replication of video data over the shared air medium, we cannot transmit the base layer twice as we did in the SIMU-S scheme. Therefore, as shown in Figure 2(d), we transmit the base layer only over bootstrap bursts and the enhancement layer over primary bursts. Mobile devices that just switch to a new

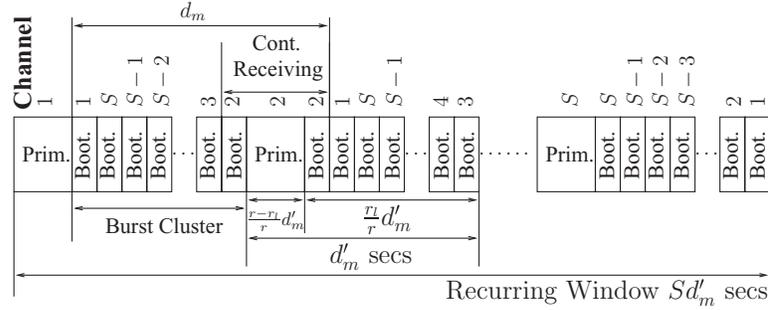


Fig. 6. Burst allocation for the SIMU-S+ scheme.

TV channel can start playing the base layer for reduced-quality videos and add enhancement-layer streams for full-quality videos whenever bursts on primary trains are available. We now design a time slicing scheme for SIMU-S+. The main difference between SIMU-S+ and SIMU-S is that mobile devices have to receive bootstrap bursts even in the steady stage. That is, more antenna on/off operations are required in each recurring window, which leads to higher energy consumption as each antenna on/off operation imposes an overhead duration T_o . To minimize this negative impact, we propose to shift every subsequent bootstrap burst by one as indicated in Figure 6. The shifting operation enables us to concatenate three bursts that belong to the same TV channel, say channel 2 as illustrated in Figure 6, together so that mobile devices can receive three bursts without turning off their antennas.

However, due to the shifting operations, the last burst in each bootstrap burst cluster should be smaller than other bootstrap bursts. This is because the last burst has a shorter inter-burst time as it will become the first burst in the following bootstrap burst cluster, while all other bursts must be shifted to the right for one time slot. Let x be this bootstrap time slot width. Let d'_m be the time length of a primary burst and a cluster of bootstrap bursts. To guarantee that the channel switching delay does not exceed d_m sec, we set $d'_m + x = d_m$, which keeps any two adjacent bootstrap bursts of the same TV channel apart no longer than d_m sec. Since the bit rates of the primary and the bootstrap burst trains are given and the air medium bandwidth is a constant, we allocate $\frac{r-r_l}{r}d'_m$ time slots to each primary burst for any channel s , and $\frac{r_l}{r}d'_m$ for every bootstrap burst cluster. The latter one is then split into $\frac{r_l}{rS}[d'_m - (S-1)x]$ for the last bootstrap burst and $\frac{r_l}{rS}(d'_m + x)$ for others. Following the definition of x we have

$$\frac{r_l}{rS}(d'_m + x) = x \implies x = \frac{r_l}{rS}d'_m \quad \text{and} \quad d_m = \frac{rS}{rS - r_l}d'_m.$$

The previous equation gives us the bootstrap slot size x . We then allocate the primary bursts of TV channel s ($s = 1, 2, \dots, S$) as

$$\langle (s-1)d'_m, (r-r_l)Sd'_m \rangle, \quad (7)$$

where the first element is the burst start time in seconds, the second element is the burst size in kb, and $d'_m = \frac{rS-r_l}{rS}d_m$. The bootstrap bursts of TV channel s ($s = 1, 2, \dots, S$) are allocated as

$$\left\langle kd'_m - \frac{r_l d'_m}{r} + \frac{(i-1)r_l d'_m}{rS}, \frac{r-r_l}{r}d'_m r_l \right\rangle, \quad (1 \leq k \leq S \text{ and } i = S),$$

$$\left\langle kd'_m - \frac{r_l d'_m}{r} + \frac{(i-1)r_l d'_m}{rS}, d'_m r_l \right\rangle, \quad (1 \leq k \leq S \text{ and } i \neq S), \quad (8)$$

where the first element is the burst start time, the second is its size, and $i = [(S + k - s) \bmod S] + 1$. Note that i represents the position of each bootstrap burst within a burst cluster and accommodates the shifting operations. The following theorem shows that the preceding allocation for primary and bootstrap bursts is valid and optimal.

THEOREM 2. *For simulcasting with scalable coding in mobile TV networks, allocating the primary and bootstrap bursts according to Eqs. (7) and (8), respectively, yields a valid time slicing scheme (i.e., with no buffer over/underflow for any TV channel), meets the switching delay constraint, and minimizes the energy consumption. Moreover, the energy saving is given by $\gamma = 1 - \frac{r}{R} - \frac{(S-1)T_o}{Sd'_m}$, where $d'_m = \frac{rS-r_l}{rS}d_m$.*

PROOF. We first show the allocation is valid. For any mobile device within a recurring window Sd_m , it receives one primary burst with size $(r - r_l)Sd'_m$ and S bootstrap bursts with aggregate size $\frac{r-r_l}{r}d_mr_l + d_mr_l(S - 1)$. Rearranging the aggregate size of bootstrap bursts leads to

$$\begin{aligned} \frac{r - r_l}{r}d_mr_l + d_mr_l(S - 1) &= d_mr_l - \frac{r_l}{r}d_mr_l + Sd_mr_l - d_mr_l \\ &= \frac{rS - r_l}{rS}d_mSr_l = d'_mSr_l. \end{aligned} \quad (9)$$

Hence, the number of received bits between adjacent bootstrap bursts is equal to the number of consumed bits before the first primary burst arrives (at reduced quality with bit-rate r_l), and the same balance holds when receiving the primary bursts (at full quality with bit-rate r). Therefore, our allocation is valid.

Second, according to Eq. (8), the distance between two consecutive bootstrap bursts is $d'_m + \frac{rd_m}{rS} = d'_m + x = d_m$. Hence, our scheme meets the switching delay constraint.

Third, the optimality of the SIMU-S+ scheme can be shown by contradiction using the same idea of the proof in Theorem 1. The details are not given for brevity.

Last, we derive the energy saving achieved by this scheme, in which mobile devices receive both primary and bootstrap bursts all the time. Notice that as we concatenate three bursts together as mentioned earlier, the total number of bursts in steady stage is $S - 1$. In addition, the aggregate receiving time is Srd'_m/R . The γ is then given as $\gamma = 1 - \frac{Srd'_m/R + (S-1)T_o}{Sd_m}$. \square

4.3 Supporting Heterogeneous Mobile Devices

So far, we consider each TV channel is encoded in two layers for all mobile devices. These mobile devices first receive base layers for reduced quality and switch to enhancement layers for full quality. For mobile devices with heterogeneous resources, some of them may not be capable to render TV channels at full quality, and have to resort to reduced quality provided by base layers. For example, TV channels coded at full quality with a high resolution may not be renderable on some mobile devices that have limited processing power, small display sizes, and/or low battery levels. These mobile devices are effectively denied from the mobile TV service, which in turn degrades revenues of network operators. To address this issue, we propose to generalize the SIMU-S and SIMU-S+ schemes in order to support heterogeneous mobile devices. Details on the generalized schemes are given in the following.

To support heterogeneous mobile devices in SIMU scheme, we encode each TV channel into a base layer with bit-rate r_l kbps, and N equal size enhancement layers, where each enhancement layer has a bit-rate $(r - r_l)/N$ kbps. We then broadcast the base layer over bootstrap bursts and primary bursts. In primary bursts, we place the base layer at the start of each burst, and we append enhancement layers after the base layer in ascending order. Mobile devices that switch to new TV channels receive the bootstrap bursts and decode base layers, which is the same as in the SIMU scheme. Upon reaching primary bursts, each mobile device determines the number of enhancement layers it can render based

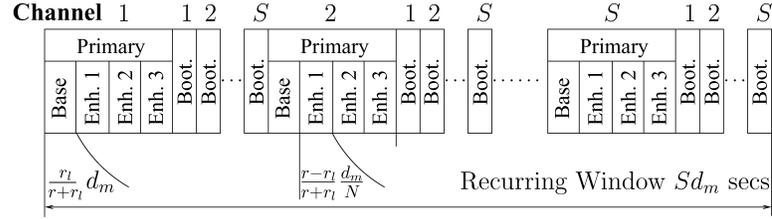


Fig. 7. HSIMU-S: generalized SIMU-S scheme for heterogeneous mobile devices.

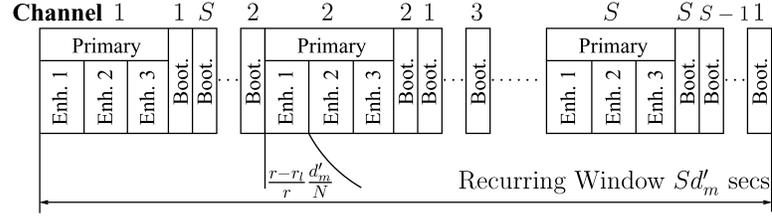


Fig. 8. HSIMU-S+: generalized SIMU-S+ scheme for heterogeneous mobile devices.

on available resources. We refer to a mobile device that renders n ($0 \leq n \leq N$) enhancement layers as a class n mobile device. Class n mobile devices receive a prefix of each primary burst, which includes base layer and enhancement layers from 1 to n . That is, mobile devices turn off their RF circuits earlier than the end of each burst to save energy as data in other enhancement layers are of no use to them. We call this new scheme as HSIMU-S. Figure 7 illustrates that we proportionally divide primary bursts into a base layer and N equal size enhancement layers. Therefore, the first $\frac{r_l}{r+r_l}d_m$ sec of each primary burst is reserved for the base layer, which is followed by N enhancement layers, where $\frac{r-r_l}{r+r_l} \frac{d_m}{N}$ sec is reserved for each enhancement layer.

We also generalize SIMU-S+ scheme to support heterogeneous mobile devices. Figure 8 illustrates the new scheme, which we refer to as HSIMU-S+. This figure shows that each primary burst is divided into N equal size enhancement layers, and $\frac{r-r_l}{r} \frac{d'_m}{N}$ sec is reserved for each enhancement layer, where $d'_m = \frac{rS-r_l}{rS}d_m$. Similar to HSIMU-S, mobile devices receive a prefix of each primary burst and turn off their RF circuits to skip irrelevant enhancement layers in order to save energy.

We analyze the energy savings achieved by these two new schemes in the next theorem.

THEOREM 3. *Encoding each TV channel into a base layer at r_l kbps and N enhancement layers each at $(r-r_l)/N$ kbps, and broadcasting them following the HSIMU-S scheme allows class n ($0 \leq n \leq N$) mobile devices to achieve energy saving γ_p^n in the steady state, which can be written as*

$$\gamma_p^n = 1 - \frac{r_l}{R} - \frac{n(r-r_l)}{NR} - \frac{T_o}{Sd_m}. \quad (10)$$

The average energy saving for a user with watch time w is given as

$$\gamma^n = \frac{Sd_m}{w} \left(\frac{n(r-r_l)}{NR} - \frac{(S-1)T_o}{Sd_m} \right) + \left(1 - \frac{r_l}{R} - \frac{T_o}{Sd_m} - \frac{n(r-r_l)}{NR} \right). \quad (11)$$

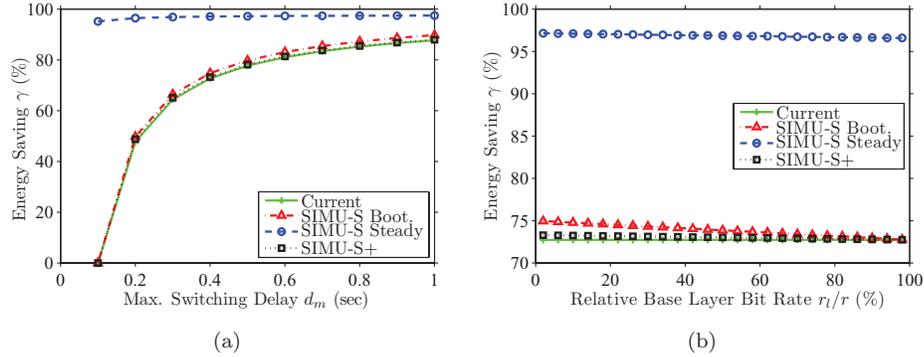


Fig. 9. Energy saving comparison among considered time slicing schemes: (a) with various maximum channel switching delay and (b) with various reduced bit rates.

Furthermore, broadcasting the same coded stream with HSIMU-S+ enables class n mobile devices to achieve average energy saving γ^n as

$$\gamma^n = 1 - \frac{r_l}{R} - \frac{n(r - r_l)}{NR} - \frac{(S - 1)T_o}{Sd'_m}, \quad (12)$$

where $d'_m = \frac{rS - r_l}{rS} d_m$.

PROOF. In HSIMU-S scheme, we observe that class n mobile devices receive a base layer at rate r_l kbps and an enhancement layer at rate $(r - r_l)n/N$ kbps. Following the definition of energy saving, we have Eq. (10). Then, we compute the average energy saving γ^n for a user with watch time w by taking the weighted average of the energy savings in bootstrap and primary stage. That is, we write

$$\gamma^n = \frac{Sd_m/2}{w} \left(1 - \frac{r_l}{R} - \frac{T_o}{d_m} \right) + \left(1 - \frac{Sd_m/2}{w} \right) \left(1 - \frac{r_l}{R} - \frac{T_o}{Sd_m} - \frac{n(r - r_l)}{NR} \right).$$

Manipulating this equation yields Eq. (11).

In the HSIMU-S+ scheme, class n mobile devices receive at the same rate as in HSIMU-S scheme. However, more bursts are received in the HSIMU-S+ scheme, and each burst incurs an overhead duration. Since there are $S - 1$ bursts in each recurring window, following the definition of the energy saving, we have Eq. (12). \square

This theorem shows that while a mobile device can achieve higher perceived video quality by placing itself in a higher class, doing so results in lower energy saving as it cannot turn off their RF circuits earlier to save energy.

4.4 Numerical Analysis and Comparison

We compare the proposed time slicing schemes against each other and against the current scheme by numerically analyzing the equations derived in the previous subsections. We start from homogeneous mobile devices, where we use SIMU-S to refer to both SIMU and SIMU-S because they use the same time slicing scheme, with the only difference in how the video streams are encoded and stored. We then consider mobile devices with heterogeneous resources.

We consider typical system parameters of mobile TV networks and compute energy savings in various settings. We first fix all parameters except the maximum channel switching delay d_m . We set $T_o = 100$ msec, $R = 11$ Mbps, $r_l = 25$ kbps, and $r = 250$ kbps. We vary d_m from 0.1 to 1 sec. Figure 9(a)

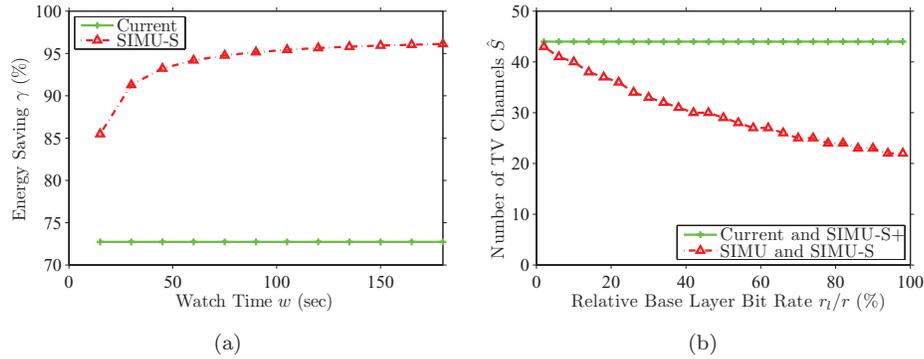


Fig. 10. (a) Energy saving achieved by SIMU-S under different watch time and (b) maximal number of channels of considered time slicing schemes with various base-layer bit rates.

shows the energy saving achieved by all considered solutions. In the figure, SIMU-S Boot and SIMU-S Steady denote the energy saving achieved by the SIMU-S time slicing scheme when mobile devices receive bootstrap and primary bursts, respectively. This figure indicates that SIMU-S scheme always achieves more than 95% energy saving for mobile devices in the steady stage. In contrast, the current solution results in much lower energy saving especially when d_m is small. Actually, when d_m approaches T_o , the current solution degrades to *continuous* transmission that leads to zero energy saving. Another observation we can draw from Figure 9(a) is that, for mobile devices in the bootstrap stage, the SIMU-S scheme yields a marginal energy saving gain over the current scheme. This, however, is not a big issue because most devices spend a small fraction of time in the bootstrap stage. We also note that our SIMU-S+ scheme does not result in additional energy savings compared to the current scheme. This is because mobile devices in SIMU-S+ networks always receive all bursts, which leads to high energy overhead due to antenna on/off operations.

We next study the effect of changing the base layer bit-rate r_l on energy saving. We fix $d_m = 400$ msec, $T_o = 100$ msec, $R = 11$ Mbps, and $r = 250$ kbps. We vary r_l from 15 to 200 kbps. Figure 9(b) illustrates the energy saving under different relative base-layer rates resulted by all considered schemes. This figure shows that our SIMU-S scheme achieves much higher energy savings than the current scheme for all possible base-layer bit rates. However, our SIMU-S+ scheme results in marginal improvement, about 1%, compared to the current scheme. Based on this observation, we conclude that both current and SIMU-S+ schemes scarify energy saving for low switching delays while our SIMU-S scheme achieves both high energy savings and low switching delays.

We then consider the average energy saving achieved by the SIMU-S scheme under different user watch time w . We set the maximum switching delay $d_m = 0.4$ sec for this experiment, and we vary the w from 15 sec to 3 min. We plot energy savings achieved by current and SIMU-S schemes in Figure 10(a). This figure shows that the SIMU-S scheme achieves as high as 95% average energy saving when a user watches a TV channel for more than 90 sec, while the current scheme always leads to 73%. Moreover, even when users frequently switch TV channels, the SIMU-S scheme still achieves 85% energy saving, which is 12% more than the current scheme. Finally, we note that SIMU-S trades broadcast bandwidth for good energy savings. Figure 10(b) shows that as the relative base-layer rate increases, the number of TV channels that can be broadcast decreases when the SIMU-S scheme is used.

To summarize, there are three main aspects in mobile TV networks: energy saving, channel switching delay, and bandwidth utilization. The first two are important for extending the viewing time and improving the quality of experience for users, while the third aspect is important for network operators

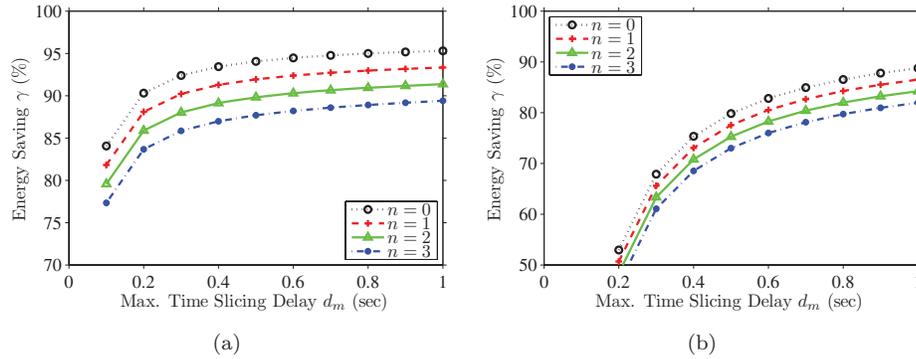


Fig. 11. The proposed schemes: (a) HSIMU-S and (b) HSIMU-S+ allow mobile devices to receive fewer enhancement layers, thus save energy in addition to that achieved by time slicing.

in order to offer more TV channels. Our analysis in the previous subsections shows that these three issues are inherently conflicting, meaning that all of them cannot be optimized at the same time. For example, the current time slicing scheme can achieve full bandwidth utilization, but it trades energy savings for shorter switching delays. Our proposed SIMU and SIMU-S schemes achieve optimal energy saving and small switching delays, but at the expense of reduced bandwidth utilization. Our SIMU-S+ scheme achieves a slightly better energy saving than the current scheme (but less than SIMU and SIMU-S) while fully utilizing the bandwidth and meeting a required maximum channel switching delay.

We next report the performance of two time slicing schemes, HSIMU-S and HSIMU-S+, designed for heterogeneous mobile devices. These two schemes enable mobile devices to receive fewer enhancement layers and turn off their RF circuits earlier to skip irrelevant layers in order to save energy. We let $T_o = 100$ msec, $R = 11$ Mbps, $r = 1$ Mbps, and $r_l = 250$ kbps. We vary the maximum channel switching delay from 0.1 to 1 sec. We assume all TV channels are encoded into three enhancement layers. Each mobile device then places itself into one of the four classes: 0, 1, 2, and 3. For both considered algorithms, we compute the energy saving achieved by each class of mobile devices under various channel switching delay d_m from 0.1 to 1 sec. We plot the results in Figure 11. This figure clearly shows that both time slicing schemes enable energy saving differentiation among mobile devices in various classes. For example, in Figure 11(a), with $d_m = 0.6$ sec, class 0 mobile devices achieve 94% energy saving and class 3 mobile devices achieve 87%. This shows that mobile devices that cannot render at full quality can skip irrelevant enhancement layers and achieve higher energy savings in addition to that achieved by time slicing.

5. IMPLEMENTATION AND EVALUATION IN A DVB-H MOBILE TV TESTBED

In this section, we evaluate our proposed time slicing schemes in a mobile TV testbed. For homogeneous mobile devices, we focus on the SIMU-S scheme because it combines the advantages of the SIMU scheme with the low storage and simple stream management of scalable coding. We also have shown in Section 4 that both SIMU and SIMU-S achieve the same energy saving, which is higher than that of SIMU-S+. For heterogeneous mobile devices, we implement both HSIMU-S and HSIMU-S+ schemes.

5.1 Testbed Setup and Metrics

To evaluate our time slicing schemes in real systems, we have implemented a DVB-H testbed network in our lab. The testbed provides a realistic platform for analyzing various aspects of these networks,

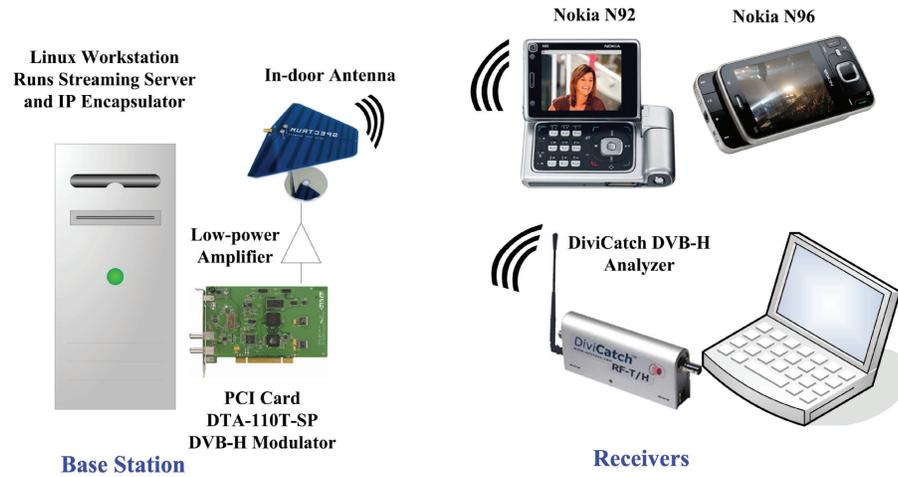


Fig. 12. Setup of the mobile TV (DVB-H) testbed.

including the energy saving achieved by the time slicing mechanism, average channel switching delay, network capacity in terms of number of TV channels that can be broadcast, visual quality of TV channels transmitting different types of video streams, information exchange and interactivity between base station and receivers, among many others. As shown in Figure 12, our mobile TV testbed has two main parts: base station and receivers, which are described next.

The base station is a commodity Linux box which runs a video streaming server and a broadcast base station. The video streaming server supports scalable streams and sends RTP packets to the broadcast base station over the loopback interface. The broadcast base station receives these RTP packets, encapsulates them into MPE and MPE-FEC sections, and fits these sections into time slicing bursts. The encapsulated MPE and MPE-FEC sections form MPEG-2 traffic streams, which are sent to the DVB-H modulator and then transmitted over the air to mobile devices. We installed a Dektec DTA-110T-SP PCI modulator [Dektec Modulator 2008], which implements the physical layer of the protocol stack and transmits DVB-H standard compliant signals via an indoor antenna. The RF output level of the modulator, however, is quite low (~ -29 dBm). We use a low-power amplifier to boost the RF signal to about 0 dBm, which gives us more than 20-meter range in our lab environment with a regular cellular phone antenna.

We use Nokia N96 and N92 devices as receivers. They are equipped with the receiver-side of the DVB-H protocol and video player. While N96 and N92 phones help in assessing the visual quality of videos, they do not provide detailed logging functions of the low-level signals, which are needed to evaluate the performance of different protocols. To address this shortcoming, we added a DVB-H analyzer available from DiviCatch Analyzer [2008] to the testbed. This analyzer is attached to a PC via a USB port and comes with visualization software for analysis. The analyzer records traffic streams as well as provides detailed information on the RF signals, the MPEs, jitter, time slicing, and so on.

We have implemented the SIMU-S, HSIMU-S, and HSIMU-S+ time slicing schemes in C++ in the broadcast base station. We have also implemented the current time slicing scheme for comparison. We have redesigned the open-source IP encapsulator [FATCAPS Project 2008] to support different burst allocation algorithms with a well-defined interface. The original IP encapsulator could only support static burst allocation with uniform inter-burst distance for all TV channels.

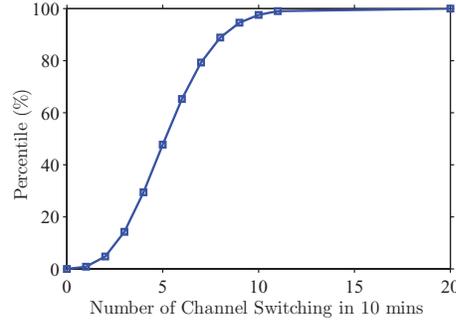


Fig. 13. Distribution of the emulated channel switching events for 1 million devices.

To conduct our experiments, we configure the testbed as follows. We use a 5 MHz DVB-H channel with Quadrature Phase-Shift Keying (QPSK) modulation, which leads to 5.445 Mbps air medium bandwidth according to the DVB-H standard [ETSI 2007]. To evaluate the SIMU-S scheme, we configure the streaming sever to send packets of size 1.5KB at bit-rate $r = 300$ kbps, and we set the reduced-quality bit-rate $r_l = 100$ kbps. Therefore, our broadcast system can concurrently broadcast up to 13 TV channels. We set the target maximum channel switching delay to be 500 msec. For each considered time slicing scheme, we broadcast 8 TV channels for 10 minutes. Our Linux server running the base station code could not handle (encapsulate, FEC-encode, etc.) more than 8 TV channels in real time. This is why we broadcast only 8 channels. To conduct statistically meaningful performance analysis, we collect detailed event logs from the base station. The logs contain the start time (in msec) of broadcasting every burst of data and its size. Using these logs, we develop a software utility to emulate the behavior of a large number (1 million) of mobile devices. We generate random channel switching events using Bernoulli trials. For every mobile receiver, we toss a biased coin every second and issue a channel switching command if the trial is success. The new selected channel is randomly chosen from all broadcast channels other than the currently watched one. We set the probability of success in a way that we have on average $w = 100$ sec watch time for each channel. We chose a small w for conservative evaluation, because our proposed schemes consume more energy in the bootstrap stage. Figure 13 depicts the distribution of the channel switching events with $w = 100$ sec, which shows mobile devices on average change their channels 6 times within the 10-min broadcast.

We then consider the proposed SIMU-S scheme under extreme conditions by varying the watch time w from 10 to 160 sec. We repeat the preceding experiment for each w value. To evaluate the HSIMU-S and HSIMU-S+ schemes, we configure the streaming sever to send each TV channel as a base layer at $r_l = 35$ kbps and two equal size enhancement layers so that the total bit rate is $r = 250$ kbps. We let the watch time $w = 100$ sec. For each time slicing scheme, we broadcast 8 TV channels for 30 min and we store the details on individual bursts in log files.

We run the emulator against each log file produced by the actual DVB-H broadcast base station and we measure channel switching delays and energy savings. We measure the channel switching delay by searching for the next burst of the new selected TV channel and computing the time difference. We measure the energy saving by computing the fraction of time that the antenna is on. We set the overhead duration $T_o = 100$ msec. When using the proposed schemes, we divide each watching period into a bootstrap stage and a steady stage. We measure the energy savings in both stages, and report their weighted average as mobile devices first receive bootstrap bursts and switch to primary bursts whenever the latter ones become available.

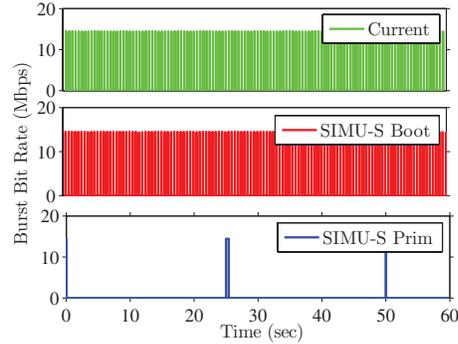


Fig. 14. Bursts allocated by SIMU-S and the current allocation schemes for a TV channel.

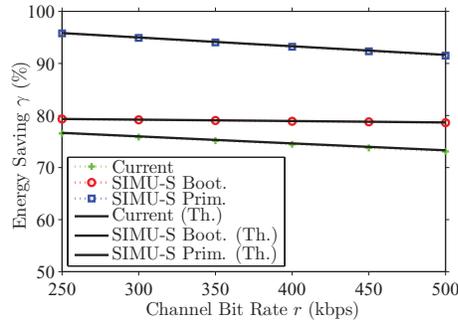


Fig. 15. Comparison between analytical and empirical energy saving curves.

5.2 Results

Correctness. We first validate the correctness of our testbed implementation. Figure 14 plots a sample result of allocated bursts for a TV channel during a minute of broadcast. Burst allocations for longer time periods and for other TV channels are similar. This figure shows that to meet a channel switching delay of 500 msec, the current scheme allocates many short bursts, thus results in low energy savings. While the SIMU-S scheme also employs short bootstrap bursts, it allocates primary bursts with a longer inter-burst distance. By switching from bootstrap to primary bursts, the SIMU-S scheme achieves both short switching delays and high energy savings.

Validation of our analytical analysis. We validate the analytical energy saving formulas using the collected empirical data as follows. We consider several TV channel bit rates: from 250 to 500 kbps. For each TV channel bit-rate r , we select a reduced-quality bit-rate $r_l = 0.2r$ kbps. We then broadcast for 15 minutes at various r using both current and SIMU-S schemes. We measure the energy saving from the collected logs for the SIMU-S and the current time slicing schemes. We also compute the theoretical energy saving from the corresponding equations in Section 3 and Section 4. We plot the theoretical (denoted by Th.) and the empirical results in Figure 15. This figure clearly shows that our analytical formulas closely follow the empirical data collected from the real DVB-H testbed.

Switching delay guarantee. In Section 3, we prove that our time slicing schemes provide a guarantee on channel switching delay. Here, we show that our testbed implementation does achieve the

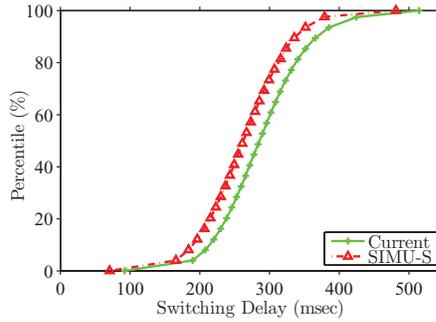


Fig. 16. Channel switching delay distribution of considered schemes.

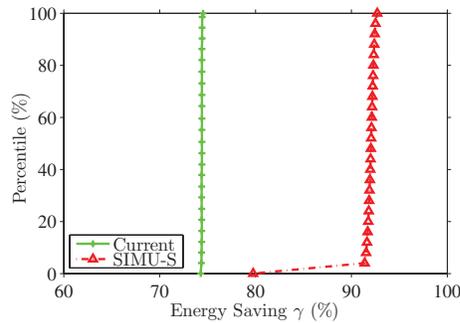


Fig. 17. Energy saving comparison between SIMU-S and the current schemes.

guaranteed delays. Figure 16 shows the distribution of the channel switching delay observed by the mobile devices switching among randomly chosen channels. This figure shows that the switching delays for all mobile devices are below the target delay of 500 msec.

Energy saving. Figure 17 reports the average energy saving achieved by the current and SIMU-S schemes. This figure shows that while the current scheme results in 74% energy saving, our proposed SIMU-S scheme yields up to 93% energy saving while providing a guarantee on switching delay. Note that, as the current scheme allocates bursts at uniform distance, the average energy saving does not depend on the time of channel switching events. Hence, most mobile devices achieve about 74% energy saving. In contrast, in the SIMU-S scheme, mobile devices achieve slightly diverse energy saving. This is because of the existence of the bootstrap and primary burst trains. Recall that mobile devices first tune for the bootstrap bursts, then switch to the primary bursts once they become available. Figure 18 plots the average period of time a mobile device waits in the bootstrap stage until it receives the first primary burst. We observe some diversity in this waiting period, which explains the minor differences in the achieved energy saving for SIMU-S in Figure 17. This figure illustrates that it takes mobile devices on average only 3.8 sec to switch to high-quality videos in this test scenario. The short period in the bootstrap stage is desirable, because it means that users will obtain full-quality video sooner, and they may not observe the reduced quality during the short transition period.

Implication of watch time. We report the energy savings achieved by the SIMU-S scheme under different watch time w . We repeat the experiment several times with various w values: 10, 20, 40, 80, and 160 sec, and we measure the average energy saving achieved by mobile devices. We plot the results in Figure 19. This figure shows that the proposed scheme achieves on average 87% even under

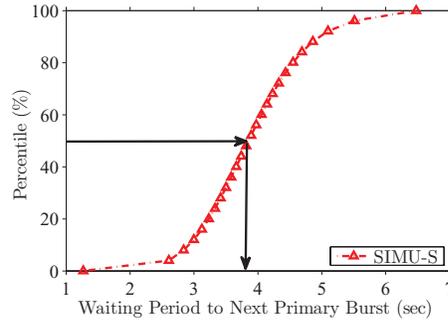


Fig. 18. Waiting period of time before switching to primary trains.

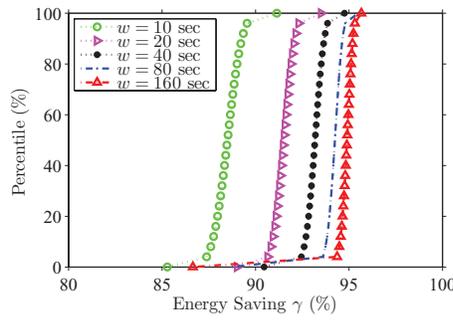


Fig. 19. Energy saving achieved by SIMU-S scheme under different watch time w .

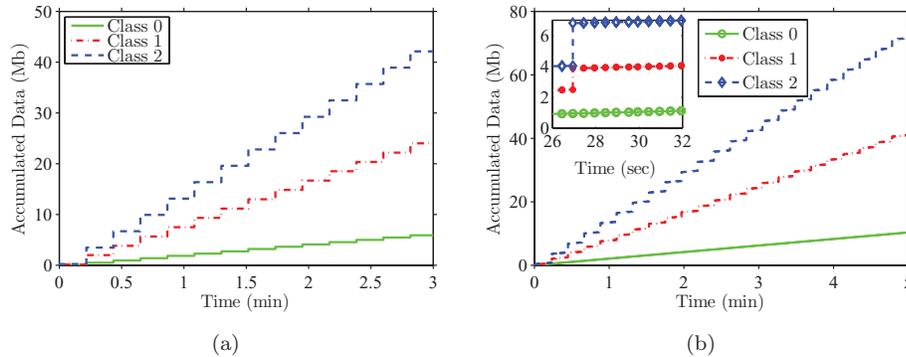


Fig. 20. Accumulated received data amount of: (a) TV channel 1 in HSI-MU-S scheme and (b) TV channel 2 in HSI-MU-S+ scheme.

an extreme condition where users watch each TV channel for only 10 sec. This is in contrast to 74% energy saving achieved by the current scheme, as illustrated in Figure 17. As high as 95% energy saving is possible when users continuously watch a TV channel for more than a couple of minutes, which is the common scenario as TV commercials are apart for at least 10 minutes.

Supporting heterogeneous mobile devices. We show the practicality of HSI-MU-S and HSI-MU-S+ schemes, in which each TV channel is encoded into a base layer for basic quality and several enhancement layers for quality improvements. Each mobile device can determine how many enhancement layers it can render based on its capability. Figure 20 plots the received data amount of a sample TV

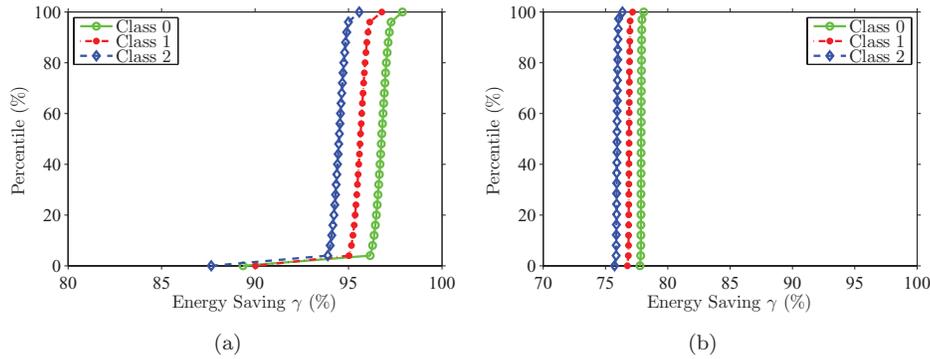


Fig. 21. Energy saving comparison among mobile device classes in: (a) HSIMU-S and (b) HSIMU-S+ schemes.

channel in the HSIMU-S and HSIMU-S+ schemes; other TV channels lead to similar results. This figure shows that both time slicing schemes enable mobile devices to skip enhancement layers that cannot be rendered. We note that the main difference between these two schemes is that mobile devices in HSIMU-S receive longer, primary bursts, while mobile devices in HSIMU-S+ scheme have to receive much shorter, bootstrap bursts. This can be observed in the zoom-in subfigure of Figure 21(b), where each marker represents a bootstrap burst. Therefore, HSIMU-S achieves higher energy savings compared to HSIMU-S+. Figure 21 shows that mobile devices that receive fewer enhancement layers achieve higher energy savings, as they can turn off the RF circuits earlier. It confirms that mobile devices in the HSIMU-S scheme achieve higher energy savings than those in HSIMU-S+.

Video quality. Finally, we study whether the reduced bit rates for bootstrap bursts can provide reasonable video quality. We encode two video sequences, soccer and harbour, using the H.264/SVC reference coder [Joint Video Team 2008]. Both video sequences are in CIF format at 15 frames per sec (fps), which is very close to the popular QVGA resolution adopted by many mobile TV devices in the market. We compress these two sequences at various bit rates. To visually inspect the resulting frame quality, we present sample reconstructed pictures of each sequence at two bit rates in Figures 22 and 23. Notice that several impairments can be observed in the reduced-quality pictures, for example, there are many blocking artifacts around the rightmost runner’s leg in frame 96 of the soccer sequence when it is encoded at 100 kbps. Nevertheless, the reduced-quality pictures provide users rough representations, and enable a much higher energy saving as we have shown before.

To better quantify the video quality, we compute the average Peak Signal-to-Noise Ratio (PSNR) values of individual coded streams. We plot the Rate-Distortion (R-D) curves in Figure 24. This figure shows that H.264/AVC achieves reasonable video quality even at a low bit rate. For example, 30 dB video quality can be achieved at 300 kbps for the harbour sequence, and at 100 kbps for the soccer sequence. We note that PSNR values above 30 dB are considered fairly good quality [Wang et al. 2001, pp. 29]. In summary, this experiment shows that video streams coded at a fairly low bit rate could result in acceptable video quality.

6. CONCLUSIONS AND FUTURE WORK

6.1 Conclusions

In this article, we considered the problem of controlling the channel switching delay in mobile TV networks that use time slicing to save energy. Our objective is to provide a guarantee on the maximum switching delay from a TV channel to any other channel, without sacrificing energy savings for mobile devices. We analyzed the time slicing scheme used in the current mobile TV networks and showed that



Fig. 22. Sample reconstructed frames from soccer sequence coded at two bit rates.

it does not minimize the energy consumption for mobile devices. We proposed new time slicing schemes: SIMU, SIMU-S, and SIMU-S+, which are provably optimal in terms of energy saving. Our analysis showed that there are three aspects in mobile TV networks: energy saving, channel switching delay, and bandwidth utilization. These three aspects cannot be optimized at the same time. For example, the current time slicing scheme can achieve full bandwidth utilization, but it trades energy saving for shorter switching delays. Our proposed SIMU and SIMU-S schemes achieve optimal energy savings and small switching delays, but at the expense of reduced bandwidth utilization. Our SIMU-S+ scheme achieves a slightly better energy saving than the current scheme while fully utilizing the bandwidth and meeting a required maximum channel switching delay.

Using our analysis, the appropriate time slicing scheme can be chosen for a given environment: most broadcast networks should adopt the SIMU or SIMU-S scheme to achieve optimal energy saving on mobile devices, and bandwidth saturated broadcast networks should employ the SIMU-S+ scheme. We generalized these schemes to support heterogeneous mobile devices, and presented two new time slicing schemes: HSIMU-S and HSIMU-S+. These two schemes utilize scalable video coding and transmit each TV channel in a base layer and multiple enhancement layers, and allow mobile devices to skip irrelevant data that cannot be rendered on them. Network operators who have a stringent bandwidth budget should use HSIMU-S+, while others should use HSIMU-S for higher energy savings.

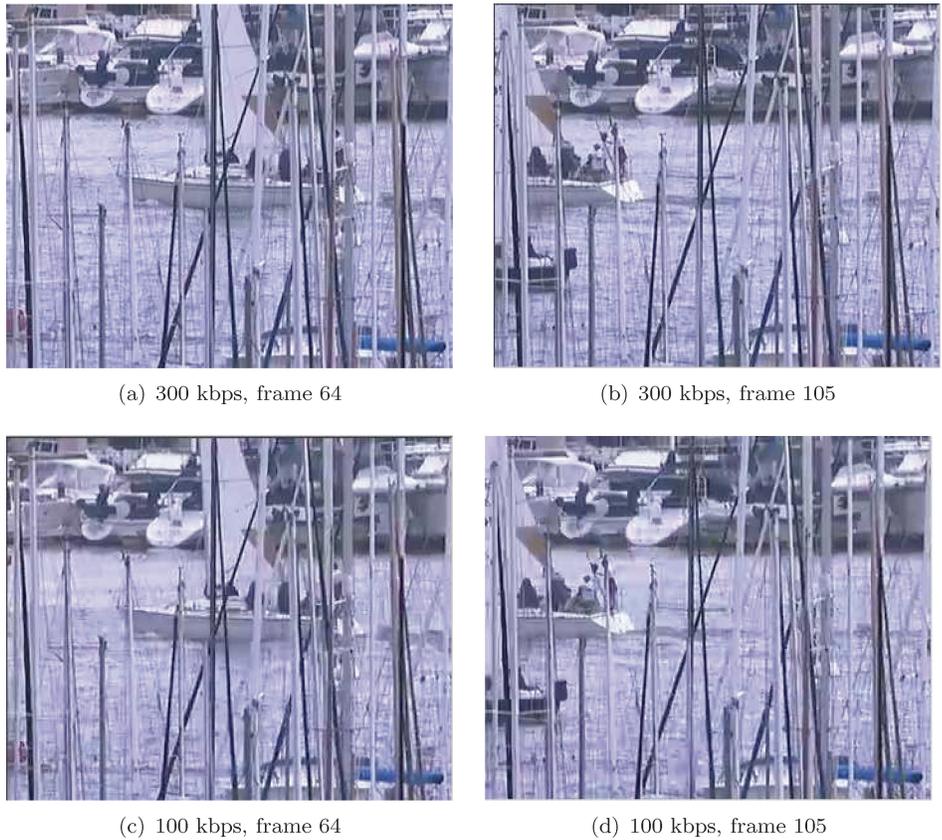


Fig. 23. Sample reconstructed frames from harbour sequence coded at two bit rates.

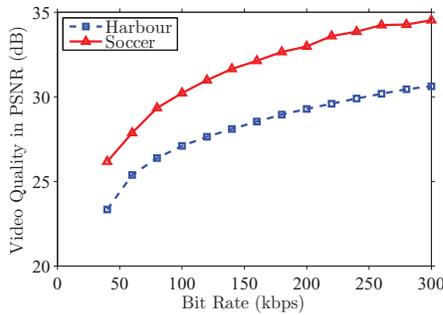


Fig. 24. Modern video coding standards enable us to transmit reasonable-quality videos at bootstrap time.

We implemented the proposed time slicing schemes in a mobile TV testbed. We conducted real-time broadcast experiments and emulated a large number of mobile devices that are randomly switching among many TV channels. Our experiments confirmed the correctness of our analysis and demonstrated that using our time slicing schemes, an energy saving as high as 95% can be achieved while guaranteeing a channel switching delay of at most 500 msec. In contrast, the current time slicing

scheme can achieve 500 msec switching delay, but at a much lower energy saving of 73%. Our experiments also revealed that the proposed schemes can support both homogeneous and heterogeneous mobile devices.

6.2 Future Work

We are extending the presented work in several directions. We describe some of them in the following.

- We stated that a lower-quality video in the bootstrap stage is not noticeable, and we showed that using modern video coders allows us to deliver essential visual information at a fairly low bit rate in Section 5.2. However, we did not provide guidelines for network operators to choose the minimum encoding rate r_l to provide satisfactory video quality in the bootstrap stage. We are designing a rigorous user study to develop these guidelines, which are useful not only for mobile TV networks but also for multimedia applications on mobile devices in general.
- We are studying the problem of prefetching adjacent TV channels for shorter channel switching delays. This problem is challenging in mobile TV networks, because mobile devices have stringent energy budgets and prefetching TV channels may result in higher energy consumption. Therefore, prefetching must be carefully done only when the prefetched channels are very likely to be viewed, otherwise the energy is wasted. A promising direction is to leverage on the correlation between channel switching events and commercial ads, because several studies, such as Siddarth and Chattopadhyay [1998], indicate that they are highly correlated.
- We are modifying the proposed HSI-MU-S scheme for higher energy savings on mobile devices in all mobile device classes. The proposed scheme is not optimal in terms of energy saving for mobile devices that render a smaller number of the enhancement layers. This is because these devices effectively receive shorter primary bursts, and each burst incurs a fixed overhead duration. Therefore, they have relatively longer overhead durations on average and achieve suboptimal energy savings. We are designing a time slicing scheme that organizes bursts in a way that mobile devices, despite their classes, always receive long primary bursts and achieve optimal energy saving.

REFERENCES

- CHO, S., LEE, G., BAE, B., YANG, K., AHN, C., LEE, S., AND AHN, C. 2007. System and services of Terrestrial Digital Multimedia Broadcasting (T-DMB). *IEEE Trans. Broadcast.* 53, 1, 171–178.
- CREUS, G. AND KUULUSA, M. 2007. *Mobile Phone Programming, Optimizing Mobile Software with Built-in Power Profiling*. Springer, Chapter 25, 449–462.
- DAOUD, K. 2007. Performance comparison of the DVB-H and FLO mobile broadcasting systems. In *Proceedings of the IEEE International Symposium on Consumer Electronics (ISCE'07)*. 1–7.
- DEKTEC MODULATOR. 2008. Dektec DTA-110T PCI modulator. <http://www.dektec.com/Products/DTA-110T/>
- DIBCOM. 2007. Evolution of DVB-T front-end receivers through integration. http://www.dibcom.info/Images/Upload/pdf/whitePaper2_integration_MD_V2.pdf
- DIVICATCH ANALYZER. 2008. Divi Catch RF-T/H transport stream analyzer. <http://www.enensys.com/>
- DVB-H FAQ. 2008. DVB-H global mobile TV : FAQ. <http://dvb-h.org/faq.htm>
- DVB-H HOME PAGE. 2009. Digital Video Broadcasting - Handheld (DVB-H) homepage. <http://www.dvb-h.org/>.
- ETSI. 2001. Radio broadcasting systems: Digital Audio Broadcasting (DAB) to mobile, portable and fixed receivers. European Telecommunications Standards Institute (ETSI) Standard EN 300 401 Ver. 1.3.3.
- ETSI. 2004. Digital Video Broadcasting (DVB); transmission system for handheld terminals (DVB-H). European Telecommunications Standards Institute (ETSI) Standard EN 302 304 Ver. 1.1.1.
- ETSI. 2007. Digital Video Broadcasting (DVB); DVB-H implementation guidelines. European Telecommunications Standards Institute (ETSI) Standard EN 102 377 Ver. 1.3.1.
- EU DVB-H. 2008. Mobile TV across Europe: Commission endorses addition of DVB-H to EU list of official standards. <http://europa.eu/rapid/pressReleasesAction.do?reference=IP/08/451&format=PDF>
- ACM Transactions on Multimedia Computing, Communications and Applications, Vol. 7, No. 2, Article 8, Publication date: February 2011.

- FARIA, G., HENRIKSSON, J., STARE, E., AND TALMOLA, P. 2006. DVB-H: Digital broadcast services to handheld devices. *Proc. IEEE* 94, 1, 194–209.
- FATCAPS PROJECT. 2008. FATCAPS project. <http://amuse.ftw.at/downloads/encapsulator>
- FLO FORUM. 2008. FLO forum homepage. <http://www.floforum.org/>
- FLO OVERVIEW. 2009. FLO technology overview. http://www.mediaflo.com/news/pdf/tech_overview.pdf
- HSU, C. AND HEFEEDA, M. 2009. Bounding switching delay in mobile TV broadcast networks. In *Proceedings of the ACM/SPIE Multimedia Computing and Networking Conference (MMCN'09)*.
- IIZUKA, K., KAWAMURA, H., FUJIWARA, T., KAGOSHIMA, K., KAWAMA, S., KIJIMA, H., KOUTANI, M., TOYOYAMA, S., AND SAKUNO, K. 2007. A 184 mW fully integrated DVB-H tuner with a linearized variable gain LNA and quadrature mixers using cross-coupled transistor. *IEEE J. Solid-State Circ.* 42, 4, 862–871.
- JOINT VIDEO TEAM. 2008. Joint scalable video model reference software. JSVM 14.0.
- KORNFELD, M. AND MAY, G. 2007. DVB-H and IP Datacast—Broadcast to handheld devices. *IEEE Trans. Broadcast.* 53, 1, 161–170.
- MAY, G. 2004. The IP Datacast system—Overview and mobility aspects. In *Proceedings of the IEEE International Symposium on Consumer Electronics (ISCE'04)*. 509–514.
- OLLIKAINEN, V. AND PENG, C. 2006. A handover approach to DVB-H services. In *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME'06)*. 629–632.
- PHILIPS DVB-H CHIP. 2006. Philips SDIO TV1000/TV1100 mobile/portable TV solutions. http://www.nxp.com/acrobat_download/other/products/rf/SDIO_TV_final.pdf
- REZAEI, M., BOUAZIZI, I., AND GABBOUJ, M. 2008. Joint video coding and statistical multiplexing for broadcasting over DVB-H channels. *IEEE Trans. Multimedia* 10, 7, 1455–1464.
- REZAEI, M., BOUAZIZI, I., VADAKITAL, V., AND GABBOUJ, M. 2007a. Optimal channel changing delay for mobile TV over DVB-H. In *Proceedings of the IEEE International Conference on Portable Information Devices (PORTABLE'07)*. 1–5.
- REZAEI, M., HANNUKSELA, M., AND GABBOUJ, M. 2006a. Video encoding and splicing for tune-in time reduction in IP datacasting (IPDC) over DVB-H. In *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME'06)*. 601–604.
- REZAEI, M., HANNUKSELA, M., AND GABBOUJ, M. 2006b. Video splicing and fuzzy rate control in IP multi-protocol encapsulator for tune-in time reduction in IP datacasting (IPDC) over DVB-H. In *Proceedings of the IEEE International Conference on Image Processing (ICIP'06)*. 3041–3044.
- REZAEI, M., HANNUKSELA, M., AND GABBOUJ, M. 2007b. Tune-In time reduction in video streaming over DVB-H. *IEEE Trans. Broadcast.* 53, 1, 320–328.
- SIDDARTH, S. AND CHATTOPADHYAY, A. 1998. To zap or not to zap: A study of the determinants of channel switching during commercials. *Market. Sci.* 17, 2, 124–138.
- TAKADA, M. AND SAITO, M. 2006. Transmission system for ISDB-T. *Proc. IEEE* 94, 1, 251–256.
- TAN, E., GUO, L., CHEN, S., AND ZHANG, X. 2007. PSM-Throttling: Minimizing energy consumption for bulk data communications in WLANs. In *Proceedings of the IEEE International Conference on Network Protocols (ICNP'07)*. 123–132.
- VADAKITAL, V., HANNUKSELA, M., AND GABBOUJ, M. 2007. Time-Interleaved simulcast and redundant intra picture insertion for reducing tune-in delay in DVB-H. In *Proceedings of the IEEE International Packet Video Workshop (PV'07)*. 123–132.
- WANG, Y., OSTERMANN, J., AND ZHANG, Y. 2001. *Video Processing and Communications* 1st Ed. Prentice Hall.
- YANG, X., SONG, Y., OWENS, T., COSMAS, J., AND ITAGAKI, T. 2004. Performance analysis of time slicing in DVB-H. In *Proceedings of the Joint IST Workshop on Mobile Future and Symposium on Trends in Communications (SympoTIC'04)*. 183–186.

Received March 2009; revised August 2009; accepted October 2009