Testbed and Experiments for Mobile TV (DVB-H) Networks

Mohamed Hefeeda, Cheng-Hsin Hsu, and Yi Liu
School of Computing Science
Simon Fraser University
Surrey, BC, Canada

ABSTRACT

We present a complete, running, testbed for mobile TV networks that employ the Digital Video Broadcast–Handheld (DVB-H) open standard. DVB-H based networks have been deployed in several countries around the world and currently being pilot-tested in many others. Nevertheless, there exists no open-source testbed in the literature to enable researchers to analyze and optimize the performance of such networks; most testbeds are proprietary. Our testbed implements the complete stack of the DVB-H standard and it streams real videos to actual handheld devices. It integrates several off-the-shelf hardware components and devices with software components. Some of the software components are developed by us and others are leveraged (after bug fixes and modifications) from open-source projects. In addition, we present several experiments to: (i) evaluate and compare multiple energy-saving techniques recently proposed for mobile TV networks, and (ii) demonstrate a new method to reduce the channel switching delay.

Categories and Subject Descriptors: C.4[Computer Systems Organization]: Performance of Systems

General Terms: Experimentation

1. INTRODUCTION

Delivering TV programs to mobile devices such as cell phones and PDAs can be done over cellular networks [1] or over dedicated broadcast networks. We focus on dedicated broadcast networks, which have the potential to serve TV content to a large number of subscribers. We note that dedicated broadcast networks usually employ cellular networks to enable user interaction with some TV programs, but not to transmit video. In dedicated networks, a base station concurrently broadcasts multiple TV channels to mobile devices over a common wireless medium. Example of such systems include Digital Video Broadcast-Handheld (DVB-H) [2,3] and Forward Link Only technology (MediaFLO) [4] networks.

We present a brief overview of DVB-H networks for which we have built an experimental testbed. DVB-H is an extension to the DVB-T (Digital Video Broadcast-Terrestrial) standard tailored for mobile devices. At the time of writing this paper (Summer 2008), trial or full-service DVB-H networks have been deployed in 38 countries [3]. 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. Another standard called IP Datacast [2] complements DVB-H by defining a set of higher-level protocols for a complete end-to-end solution. The complete protocol stack of video broadcasting over DVB-H networks is illustrated in the left part of Fig. 1. DVB-H uses a physical layer compatible to the DVB-T, which employs Orthogonal Frequency Division Multiplexing (OFDM) modulation. 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 before transmitted over the wireless medium. To save energy of mobile devices, MPEs belonging to a given TV channel are transmitted in *bursts* with a bit rate much higher than the video stream itself. Thus, mobile devices can receive a burst of traffic and then turn off their RF circuits till the next burst. This is called *time slicing*.

2. MOBILE TV TESTBED

We have implemented a complete end-to-end testbed for DVB-H networks. The testbed provides a realistic platform for analyzing various aspects of these networks, 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. To the best of our knowledge, there exists no *open-source* testbed for DVB-H. The details of testbeds and pilot networks created by companies are usually not published, and the source code is not available. Thus academic researchers designing algorithms and protocols for mobile TV networks, including ourselves, had to resort to simulation and/or theoretical analysis. To address this problem, we make the details and source code of our testbed available to the research community. As shown in Fig. 1, our mobile TV testbed has two main parts: base station and receivers, which are described below.

Base Station. The base station is a Linux box in which we installed the RF signal modulator available from [5]. This modulator 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 ($\sim$ -29 dBm) and can only reach up to 1-meter broadcast range with a 6 dB receiver antenna. Using a low-power amplifier, the RF signal can be boosted to about 0 dBm, which gives us approximately 20-meter range in our lab environment. IP packets of the video streams are encapsulated in MPEs and FEC-coded using the open-source IP Encapsulator available at [6]; several bug fixes were needed to get it running properly with the RF modulator. The IP Encapsulator also

MM'08, October 26–31, 2008, Vancouver, British Columbia, Canada.
ACM 978-1-60558-303-7/08/10.

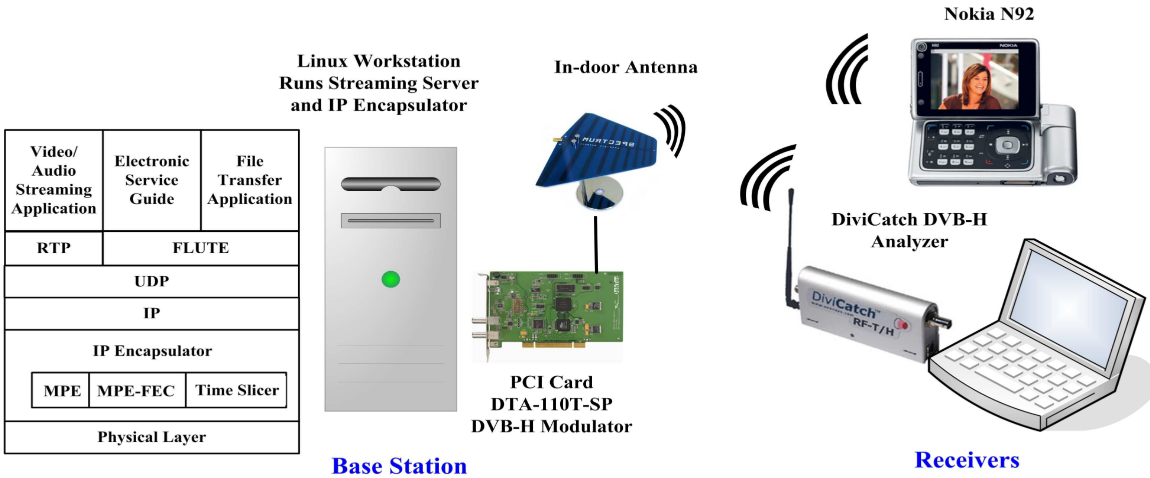

Figure 1: Setup of the Mobile TV (DVB-H) testbed.

implements a simple time slicing technique. As explained in the following section, we have extended this IP Encapsulator to support more sophisticated and optimal time slicing techniques. We have also re-designed the time slicing module to be well-structured with defined interfaces in order to facilitate implementing and comparing different current/future time slicing algorithms.

The IP Datacast standard [2] chose the Real-Time Transmission Protocol (RTP) for streaming audio and video data, and the File Delivery over Unidirectional Transport (FLUTE) protocol for delivery of other files. In addition, the Electronic Service Guide (ESG) specifications have also been developed by the IP Datacast standard. ESG is used to deliver detailed information about available services to mobile devices. The ESG data is delivered to mobile devices over the FLUTE protocol. We have implemented both the FLUTE protocol and a simple ESG, and integrated both of them into the whole stack.

Receivers. We use the Nokia N92 device as a receiver. This device is equipped with the receiver-side of the DVB-H protocol and video player. The operating system on this device (Symbian) provides several APIs, including APIs for measuring energy consumption. While the N92 device helps in assessing the visual quality of videos, it does 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 the DVB-H Analyzer available from [7] to the testbed. This analyzer can be attached to a PC via a USB port. The analyzer records traffic streams as well as provides a very detailed information on the RF signal, the MPEs, jitter, time slicing, and so on. It also comes with a visualization software that can run on the PC for analysis.

We note that the (academic) price of the RF Modulator is about $3,000 and of the Analyzer is about $6,000.

3. EXPERIMENTS TO BE DEMONSTRATED

To show the viability of the testbed, we will demonstrate several experiments on two of the most important parameters in mobile TV networks: energy consumption and channel switching delay. Minimizing the energy consumption of mobile devices is critical because they are battery-powered. Energy conservation is achieved by the time slicing mechanism, which determines a burst schedule for all TV channels. We implement the *heuristic* burst scheduling technique proposed in the standard documents [8, pp. 66], which can only work for TV channels that use the *same* bit rate.

In our recent work [9], we have formulated the general burst scheduling problem when TV channels can have arbitrary bit rates. We showed that this problem is actually NP-complete. Then we proposed a practical simplification of the general problem, which allows TV channels to be classified into multiple classes and each class has a different bit rate. The bit rate of class c, r_c, can take any value in the form of $r_c = 2^i \times r_1$, where $i \in \{0, 1, 2, 3, \dots\}$, and r_1 is the bit rate of the lowest class. r_1 can take any arbitrary bit rate. For example, the bit rates 800, 400, 200, and 100 kbps could make four different classes for encoding sports events, movies, low motion episodes, and talk shows, respectively. Using the above simplification, we developed an optimal burst scheduling algorithm, which has the time complexity is $O(S \log S)$, where S is the total number of TV channels.

We have implemented the burst scheduling algorithm in [9] in our testbed. Multiple experiments will be demonstrated to show that our algorithm provides better visual quality (as seen on the Nokia N92 device) and save energy (as shown in the detailed logs from the DVB-H Analyzer). Finally, we will also show how the channel switching delay can be controlled by encoding TV programs into multiple layers using H.264/SVC coders. The relationship between the scalable video streams, channel switching delay, and energy saving will be analyzed and demonstrated.

4. REFERENCES

[1] F. Hartung, U. Horn, J. Huschke, M. Kampmann, T. Lohmar, and M. Lundevall, "Delivery of broadcast services in 3G networks," *IEEE Transactions on Broadcasting*, vol. 53, no. 1, pp. 188–199, March 2007.

[2] M. Kornfeld and G. May, "DVB-H and IP Datacast – broadcast to handheld devices," *IEEE Transactions on Broadcasting*, vol. 53, no. 1, pp. 161–170, March 2007.

[3] "Digital Video Broadcasting - Handheld (DVB-H) home page," http://www.dvb-h.org/.

[4] "FLO technology overview," http://www.qualcomm.com/common/documents/brochures/tech_overview.pdf.

[5] "Dektec DTA-110T DVB-T/H PCI modulator product page," http://www.dektec.com/Products/DTA-110T/.

[6] "FATCAPS project home page," http://amuse.ftw.at/downloads/encapsulator.

[7] "Divi Catch RF-T/H transport stream analyzer product page," http://www.enensys.com/.

[8] European Telecommunications Standards Institute (ETSI) Standard EN 102 377 Ver. 1.3.1, "Digital video broadcasting (DVB); DVB-H implementation guidelines," May 2007.

[9] M. Hefeeda and C. Hsu, "On burst transmission scheduling in mobile TV broadcast networks," Simon Fraser University, Tech. Rep., July 2008.