Video Streaming over Cooperative Wireless Networks

Yi Liu
School of Computing Science
Simon Fraser University
Surrey, BC, Canada

Mohamed Hefeeda
School of Computing Science
Simon Fraser University
Surrey, BC, Canada

ABSTRACT

We study the problem of broadcasting video streams over a WMAN to many mobile devices. We propose to form a cooperative network among mobile devices that receive the same video stream, and share received video data over a WLAN. The proposed system significantly reduces the energy consumption and the channel switching delay. We design a distributed leader election algorithm for the cooperative system and analytically show that the proposed system outperforms current systems in terms of energy consumption and channel switching delay. We evaluate the proposed system in a real mobile video streaming testbed as well as in a trace driven simulator. Our experimental results show that the proposed system: (i) achieves as high as 70% of energy saving gain, (ii) outperforms current systems with only three cooperative mobile devices, (iii) reduces channel switching delay by up to 98%, (iv) is robust under device failures and quickly reacts to network dynamics, and (v) uniformly distributes load across all cooperative devices.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Wireless Communication*

General Terms

Design

1. INTRODUCTION

Mobile video streaming is starting to become a reality with the recent launch of 3G services and the significant consumer demand for multimedia content. Although music and gaming are currently dominant sources of revenue for mobile devices, the mobile TV service, which allows users to watch TV programs on the mobile devices, is catching up quickly. Furthermore, recent mobile devices in the market are equipped with fast processors, high resolution displays and multiple network interfaces so that they are powerful enough to natively receive, decode, and display videos sent over wireless networks. The capabilities of the mobile devices as well as the users' desire to access multimedia content anywhere and at anytime have created a strong demand for mobile TV services. Supporting mobile TV natively is also an attractive selling point for mobile device manufacturers.

Videos can be sent to mobile devices either over cellular networks or dedicated broadcast networks. Cellular networks cannot concurrently support a large number of mobile devices, as they are designed for unicast and have limited bandwidth. In addition, expanding a cellular network for higher capacity to accommodate more users is expensive. In contrast, in a Wireless Metropolitan Area Network (WMAN), a base station can *broadcast* videos to support a large number of mobile devices within its coverage range. Streaming videos over WMANs is promising as it supports more mobile devices with lower cost on network infrastructure. We consider video streaming over WMANs in this work.

Streaming videos to mobile devices is challenging, because mobile devices are battery powered and have stringent energy constraints. Therefore, energy saving on mobile devices is important since lower energy consumption leads to longer watch time and increases the user satisfaction. Current video streaming systems use WMANs to broadcast videos, in which each mobile device *independently* receives and decodes the video data from the base station. That is, every bit of data is used at most once before being discarded. In this paper, we explore the potential of better utilizing the received video data by sharing it among several mobile devices that watch the same video stream. More precisely, we propose a new video streaming system in which several mobile devices receive and cache video streams broadcast in a WMAN *and* share the received data over a Wireless Local Area Network (WLAN). Such sharing is possible as WLANs are very popular and many access points are readily available. Furthermore, our proposed cooperative system supports the ad-hoc mode of WLAN in case there is no access point available.

We explore the benefits of organizing mobile devices into cooperative networks in video broadcast systems. We propose a cooperative system that consists of a WMAN and a WLAN. We analytically compare this system against the current video broadcast systems that only use WMAN to distribute video streams. We show that the proposed cooperative system achieves better performance from several aspects. First, the energy consumption is *reduced*, because mobile devices can receive bursts from other mobile devices

MMSys'10, February 22–23, 2010, Phoenix, Arizona, USA.

over the WLAN, which consumes less energy compared to receiving bursts over the WMAN. Second, the delay of mobile devices switching to new video streams can be significantly reduced (almost *eliminated*), because these mobile devices can request an immediate burst transmission from other devices using unicast. Our experimental results from a real testbed indicate that the proposed cooperative system is promising, because it: (i) achieves as high as 70% of energy saving gain, (ii) outperforms current systems with only three (or more) cooperative mobile devices, (iii) reduces channel switching delay by up to 98%, (iv) is robust against device failures and quickly reacts to network dynamics, and (v) uniformly distributes load on all mobile devices.

The rest of the paper is organized as follows. Section 2 presents a brief background on WMAN networks and summarizes some existing strategies on energy saving and channel switching delay reduction. In Section 3, we present our cooperative strategy to save energy and reduce channel switching delay concurrently. We numerically analyze our cooperative system in Section 4. We present the experimental results from our testbed in Section 5, and we present the simulation based evaluation in Section 6. We conclude the paper in Section 7.

2. BACKGROUND AND RELATED WORK

2.1 Mobile TV in WMAN Networks

Several energy saving mechanisms have been proposed in the literature for broadcasting videos over WMANs. For example, in dedicated mobile TV networks, each TV channel is broadcast in bursts at a bit rate much higher than the encoding rate of that TV channel. Mobile devices can then receive a burst of traffic and turn off their radio frequency (RF) circuits until the next burst. This is called *time slicing*. One example of WMANs using time slicing is the DVB-H (Digital Video Broadcasting - Handheld) standard. DVB-H is an extension to the DVB-T (Digital Video Broadcast-Terrestrial) standard [1] tailored for mobile receivers. The DVB-H standard defines protocols below the network layer and uses the IP protocol as the interface with higher-layer protocols such as UDP and RTP. Another standard called IP Datacast [2, 3] complements DVB-H by defining a set of higher-level protocols for a complete end-to-end solution. The complete protocol stack [4] of video broadcasting over DVB-H networks is illustrated in 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 forms a sequence of MPEs. In addition, broadcasting videos over WMANs suffers from high data error rates caused by fading, shadowing, and interference. Since maintaining a large number of reverse connections to the base station for automatic repeat request (ARQ) is not feasible, many WMANs employ forward error correction (FEC) to mitigate the high data error rates. Thus 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 using the time slicing mechanism.

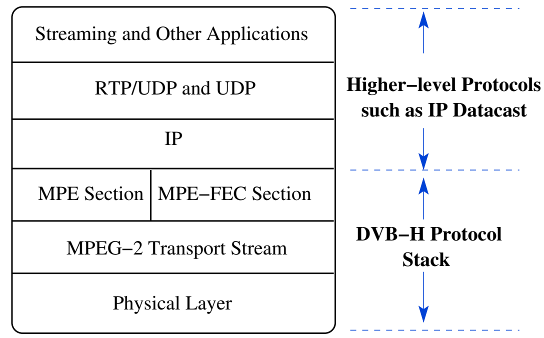

Figure 1: The DVB-H Protocol Stack.

2.2 Related Work

Some previous works explore the benefits of using cooperative network between WMANs and WLANs. Pei et al. [5] develop an analytical model in the cooperative network to evaluate the call block probability and throughput performance. They also propose two cooperative load-balancing strategies in order to improve the resource utilization. Niyato and Hossain [7] propose a hierarchical bandwidth management scheme using game theory to achieve fairness and efficiency of radio resource allocation. The optimal burst size for WMAN and WLAN connections can be determined by their proposed two-level hierarchical model. Pan et al. [8] propose a quantitative evaluation of the effect of integrating WiMAX and WiFi networks, as compared to WiMAX or WiFi only systems. Their cooperative network is used to evaluate the VoIP traffic as well as the corresponding resource management scheme on WiFi to WiMAX handover. However, to the best of our knowledge, none of the previous works addresses the video broadcast over the cooperative WMAN and WLAN networks.

A number of previous works address the energy saving problems in WMANs only network. Due to the time slicing nature of WMANs, energy saving of mobile devices can be done by burst scheduling on the base station side. Yang et al. [9] address the effectiveness of the time slicing technique for given burst schedules and calculate the corresponding energy saving level. This work does not solve the server side burst scheduling problem. Balaguer et al. [10] propose an energy saving strategy by not receiving some MPE-FEC (Multiprotocol Encapsulation - Forward Error Correction) sections once the sections received are good enough to reconstruct the data. Dropping some MPE-FEC sections can help mobile devices turn off their RF circuits earlier for additional energy saving. Hefeeda and Hsu [11] prove that the general burst scheduling problem with arbitrary bit rates of TV channels is NP-complete. They then propose a practical simplification of the general problem, which allows the bit rates of the TV channels to have power of 2 increments. The running time of this simplification is $O(S \log S)$ where S is the total number of TV channels. Hsu and Hefeeda [12] address the general scheduling problem where each TV channel can take any arbitrary bit rate, which enables finer-grained optimization by providing higher bit rate flexibility. All the above works are orthogonal to ours, because they assume all mobile devices receive video data from their WMAN interfaces only, while our work considers sharing video data over a WLAN in order to save energy *beyond* what can be saved by time slicing.

Some previous works address the channel switching delay problem in WMANs only network. Channel switching delay refers to the time period between when a user switches to a new video stream until his/her mobile device starts rendering that stream. Channel switching delay consists of several components [13], including time slicing delay, frame refresh delay, decoder buffering delay and reception delay. Time slicing delay is a dominant component among them, which is the time period between the mobile device tunes to a new video stream until the first burst of that video stream arrives. We only consider time slicing delay in this work, and assume other delay components are constant as they are not impacted by the proposed cooperative system. Hsu and Hefeeda [14] propose to control the channel switching delay by simulcasting each video over two streams. In contrast, our work uses unicast over a WLAN to almost *eliminate* the channel switching delay.

3. PROPOSED COOPERATIVE SYSTEM

3.1 Overview

We consider a video broadcast system with multiple mobile devices as receivers, where each mobile device joins a WMAN and a WLAN as illustrated in Fig. 2. The base station of this WMAN concurrently broadcasts multiple video streams to all mobile devices. These video streams are sent in *bursts* to save energy. Burst transmission, however, increases the channel switching delay. In WMAN only network, there exists a tradeoff between energy saving and channel switching delay because for a particular video channel, we save more energy by increasing the burst size in order to sleep for a longer time. However, as shown in Fig. 2(a) during the sleep time of a TV channel, no data is transmitted such that a new joiner of that channel needs to wait for a longer time to get the next burst data. This increases the channel switching delay for the new joiner. While the WMAN covers a much larger area compared to the WLAN, receiving video data from the WMAN may consume more energy than receiving from the WLAN. This is because a WLAN covers fewer mobile devices, and thus is less sensitive to path loss, shadowing, and interference. Therefore, broadcasting video streams over the WLAN achieves higher transmission rates than over the WMAN. As shown in Fig. 2(b), transmitting the same video stream over WLANs at higher rates leads to shorter transmission time, and allows mobile devices to turn off the wireless receivers for longer time, thus reduces energy consumption.

As shown in Fig. 2(c), we propose to form a *cooperative* network among all mobile devices that are viewing the same video stream. We let N be the size of the cooperative network. We then run a leadership protocol among these N mobile devices to elect one device that is *on-duty*. The on-duty device receives data bursts from the WMAN base station, and relays the data bursts to other mobile devices via WLAN. K *backups* devices are also elected, which monitor the status of the on-duty device and initiate a switch-over if the on-duty device fails or leaves the cooperative network. All other N - K - 1 devices are *off-duty* mobile devices, which receive the data bursts over the WLAN from the on-duty device. Last, the on-duty device offers the most recent burst to mobile devices that join this cooperative network. This allows new mobile devices to start playing out faster, as they do not need to wait for the next broadcast burst in WMAN.

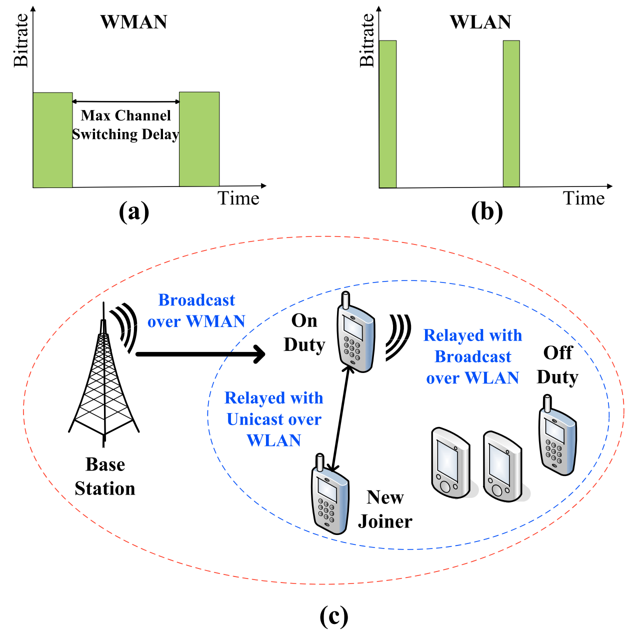

Figure 2: The proposed cooperative WMAN and WLAN video broadcast system. Mobile devices form a cooperative network over a WLAN. A device is elected to be on-duty on behalf of the group, which receives the video data from the base station over the WMAN then relays this data to other mobile devices over the much faster WLAN. A new joiner can contact the on-duty device to immediately receive video data without waiting for the next burst from the base station.

Both the infrastructure mode and ad-hoc mode in WLAN are supported in our cooperative system. The former mode requires an access point (AP). Many access points are readily available these days in places such as universities, airports, restaurants, and coffee shops. The latter mode does not require any infrastructure and it allows mobile devices to form a cooperative network in an ad-hoc manner. This ad-hoc mode is useful when no APs are available or are not freely accessible. For example, mobile users watching video streams while traveling on a bus or train could form a mobile ad-hoc network (MANET) to benefit from the proposed system to save energy and reduce channel switching delay. The support of the ad-hoc mode significantly lowers down the requirement of using our cooperative system. Additional operations such as unique IP assignment and multicast routing setup are required to form the cooperative group in MANET due to the lack of access point. We discuss these extra operations in [6] due to space limitations. We note that if the mobile devices already form a MANET to support other distributed applications, our cooperative video streaming system can easily use this MANET and there will be no need to duplicate the operations described below. In most cases, however, the MANET does not exist.

3.2 Details

We first discuss the operations under the WLAN infrastructure mode. These operations also apply to the WLAN ad-hoc mode.

3.2.1 Electing On-duty and Backup Devices

We propose a distributed algorithm to choose on-duty and backup devices. Each mobile device in our proposed cooperative system maintains a *contribution list* C[n], which contains the contribution value of each device in the cooperative group. The contribution list is updated in each WMAN cycle by the mobile device. A cycle starts with the beginning of a burst and ends with the start of the next burst, which is in the order of few seconds. The contribution value is defined as the total amount of data that has been relayed over the WLAN. Therefore, only the contribution value of the on-duty device is updated in any cycle. This also means that the contribution lists on all devices will be the same, because they are all updated at the same time based on the reception of a burst of data from the on-duty device. In order to elect the on-duty and K backup devices, devices in the cooperative group sort their contribution lists. The device with the smallest contribution value is elected as the on-duty device. This device notifies others by broadcasting an ON-DUTY message. Moreover, among the remaining devices, K devices with the smallest contribution values are elected as the backup devices and each broadcasts a BACKUP message.

Since only the on-duty device has its contribution value updated in one WMAN cycle, all devices in the group only need to update one entry in their contribution lists. The on-duty device can easily update its contribution list. For off-duty devices, they receive the data from the on-duty device such that they update their contribution list entries based on the amount of data they actually received. For K backup devices, even though they do not receive the data from the on-duty device, they receive the data from WMAN base station which is the same as the on-duty device. K backup devices can calculate the size of the data received from WMAN to update the contribution value of the on-duty device in their contribution lists because the data received from WMAN is the same as the data relayed over WLAN.

This election algorithm is totally distributed and very simple; hence it can be implemented in practice. The algorithm also maintains the election truthfulness. A device that lies about its contribution value can be detected by other devices in the group since the contribution value is calculated independently by each device. The lier device has no control of contribution lists C[n] of other devices. And the lier device once detected is kicked out from the cooperative group. In addition, the contribution list contains two other fields: join time and MAC address of each device. The join time is the *first* time a device participated in the cooperative group. In case that multiple devices have the same lowest contribution value, the device with the earliest join time will be chosen as the on-duty device, because the device that joined earlier has benefited from the cooperative network more than other devices. This scheme also does not penalize or discourage new joiners by making them on-duty right after they join. In case multiple devices have the same contribution value and join time, their MAC addresses are used to break the tie, since the MAC addresses are globally unique. The MAC address also provides another useful feature: it prevents devices from the whitewashing type of cheating. That is, a cheating device can participate in the cooperative group and save energy by receiving data from the on-duty device. When it comes its turn to serve as on-duty, it leaves the group. Then the cheating device joins a little latter with a new ID and a recent join time. Using the MAC addresses as IDs and the earliest join time a device appears in the cooperative group solve this cheating problem.

We note that the proposed algorithm provides a *practical* protection level against cheating that is sufficient for the proposed cooperative video streaming system over mobile devices with limited resources. This algorithm, however, may not be strong enough for other cooperative systems with more powerful devices. For example, a malicious device may spoof its MAC address and use a different (fake) address each time it joins the system. This MAC address spoofing may not be easily performed on mobile devices (e.g. cell phones), because most of them have light weight and secure operating systems (e.g. Symbian). In addition, the spoofing process and the continuous changing of the MAC address may impose high energy consumption on the malicious device, which may equal to or even exceed the energy spent in the legitimate cooperative process. Thus, a malicious device may be better off cooperating with other devices.

The proposed algorithm is also very efficient in terms of computation and communication. There is no computation in the algorithm except the periodic update and sorting of the contribution list at each device, which happens once in each WMAN time slicing cycle (few seconds). Also since the number of devices in the cooperative group is small (at most tens of devices), sorting the contribution list composes negligible CPU load on mobile devices. For communication overhead, there is only one message broadcast from the on-duty device to the cooperative group, and up to K messages from the backup devices. The messages are short as they only contain a single field indicating the message type, e.g., ON-DUTY or BACKUP.

3.2.2 Receiving and Relaying Bursts

After one on-duty and K backup devices are elected, the cooperative system starts to transmit data. The on-duty and backup devices start to work slightly earlier before the WMAN burst period to make sure the cooperative group can be reliably organized such that devices can successfully receive the data from the WMAN base station. This is done by setting a timeout timer to receive the ON-DUTY messages announced by the current on-duty device. The timer for the ith backup device ($1 \leq i \leq k$) is set as $i \times \delta$, where δ is the timeout period. In the worst case, all on-duty and backup devices fail, then all off-duty devices need to turn on their interfaces to receive from the WMAN directly. This is achieved by setting the total preparation period to organize the cooperative group before the start of the WMAN burst to be $\Delta = (K + 1)\delta$. Therefore, if this period passes and none of the on-duty or K backup devices announces an ON-DUTY message, all devices will use their WMAN interfaces to obtain the video data. Therefore, by using this simple scheme, we can make sure that there is no data lost in our cooperative system even in presence of device failures. This is important to ensure high quality and continuous video playback.

The on-duty device turns off its WMAN interface after receiving the WMAN burst data and starts to send data packets via WLAN. After sending, the on-duty device switches to WLAN idle state to monitor potential new joiner requests. If there exists a new joiner during this period, the on-duty device forwards a fraction of the data packets to the new joiner based on the exact join time. Thus, the new joiner no longer needs to wait until the next burst period from the WMAN base station. Due to the high speed of the WLAN, the data packet exchange is fast and the channel switching delay is significantly reduced. The on-duty device finishes its duty after a WMAN time slicing cycle.

In addition, K backup devices are used to maintain a fail-safe mechanism. They receive data packets from the WMAN directly but do not forward the data via the WLAN. Their main task is to monitor the on-duty device status. If the on-duty device fails, one of the backup devices will take over the on-duty role and will start to forward data to all off-duty devices. For the remaining off-duty devices, they do not need to turn on their WMAN interfaces and only need to turn on their WLAN interfaces to receive data from the on-duty device. From the whole system point of view, for a particular video program, instead of all N devices turn on their WMAN interfaces, only 1 + K devices need to turn on the WMAN interfaces in our cooperative network. Therefore, the more users concurrently watching that video program, the more energy we can save in our cooperative network.

3.2.3 Handling Network Dynamics

Mobile devices may join, leave, or fail at any time during the cooperation period. Our cooperative system handles these dynamics as follows.

Device Joining. Instead of waiting for the next broadcast data burst sent from the WMAN, a newly joined mobile device can send a JOIN message to the on-duty mobile device. The on-duty mobile device, upon receiving the JOIN message, updates its contribution list by adding a new entry for the new joiner. Then the on-duty device transmits the *most recent* video data burst together with the latest contribution list information of the whole cooperative group to the newly joined mobile device using unicast. This allows the new joiner to start playing out immediately after joining the cooperative network. Thus reducing the channel switching delay. By receiving the most recent contribution list of the whole group, the new joiner can quickly take part in the election of the cooperative group in the following WMAN cycle. The on-duty device also notifies all existing members of the group such that they can all add an entry for the new joiner device. There are only three messages involved in the new join procedure. One message sent from the new joiner and received by the on-duty via unicast, another message replied from the on-duty and received by the new joiner via unicast and the third message broadcast from the on-duty to all devices.

Device Leaving. If an off-duty or backup mobile device gracefully leaves the network (i.e, not suddenly fails), a LEAVE message is sent to the on-duty device such that in the next cycle to elect on-duty and backup devices, the current leaving device will no longer be included. When the on-duty device receives the LEAVE message, it deletes the leaving device entry in its contribution list and broadcasts this leave information to all members in the cooperative group. All devices will thus know about the leaving event and will update their contribution lists. There are two messages involved in this procedure. One message sent from the leaving device and another message broadcast by the on-duty device. If the on-duty mobile device leaves the network, a LEAVE message is broadcast to the whole group such that all devices remove the leaving on-duty device entry from their contribution lists. The backup device with the least contribution value among all backup devices takes over and acts as the new on-duty device. It will then notify all group members by sending an ON-DUTY message. There are two messages involved in this procedure. One message broadcast to the whole group by the leaving on-duty device and another message broadcast by the new on-duty device.

Device Failing. In the proposed cooperative system, the on-duty device periodically broadcasts small ON-DUTY messages when it is not sending actual video data. This message is frequently broadcast during the WMAN cycle. The backup devices check the status of the on-duty device by monitoring the ON-DUTY messages. Whenever the on-duty device fails, the device with the least contribution value among all backup devices takes over and acts as the new on-duty device. Since backup devices receive and cache all video data bursts, the video streaming service is not interrupted by the failure of the on-duty device. The new on-duty device will inform the whole group to remove the failed on-duty device entry in their contribution lists. If an off-duty or backup device encounters failure, no message is sent. But this is not harmful to the cooperative system and the failed devices can be detected in future on-duty election periods. This is because if the failed devices are elected as on-duty or backup devices but do not send any messages as expected, they are treated as failed and their corresponding entries are automatically removed from the contribution lists of all devices in the group. This failure detection process is effective yet simple. Thus it is suitable for the proposed cooperative system in which devices are quite dynamic and failures are common.

3.2.4 Handling Time Synchronization

Mobile devices need to know the start of bursts such that they can properly schedule their wake up timers. The time synchronization problem is solved by the nature of WMAN time slicing. The WMAN burst data packets are made of MPEG-2 Transport Stream (TS). With time slicing, MPEG-2 TS packets are sent in short bursts at the full bitrate of the MPEG-2 TS. In order for a receiver to wake up on time to receive the next burst data, the header of the TS packets contains a time parameter determining the time offset between the start time of the current burst and the next burst. Any data packets received on the receiver side contain such time offset information to wake up the receiver reliably in the next burst period. In our WMAN and WLAN cooperative system, even though off-duty mobile devices do not receive the relative time parameter directly from WMAN, this time parameter is forwarded by the on-duty mobile device via the WLAN to all off-duty mobile devices. This enables all mobile devices in the cooperative system to get the time parameter to determine the next wake up time in order to elect on-duty and backup mobile devices for WMAN reception. Therefore, the proposed cooperative system enables mobile devices to properly and efficient receive data busts on time. Our system does not employ any clock synchronization algorithms, which impose significant overhead especially if fine-grain clock synchronization is needed.

3.2.5 Initializing Cooperative Group

Whenever a new device comes to the cooperative network, it first listens to a default multicast IP address to determine if any on-duty device already exists. If there already

Parameter	Description
P_m^r	Power spent on WMAN receive
P_m^i	Power spent on WMAN idle
P_l^r	Power spent on WLAN receive
P_l^s	Power spent on WLAN send
P_l^i	Power spent on WLAN idle
T_m^b	WMAN receive period
T_m^i	WMAN sleep period
T_m^o	WMAN receive overhead
T_l^b	WLAN transmit period
T_l^i	WLAN sleep period
Δ	Cooperative group organization period

Table 1: Symbols used in the analysis.

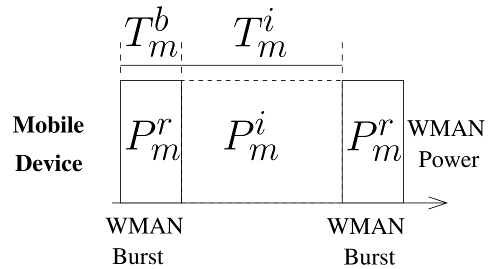

Figure 3: Burst transmission in current WMAN networks.

exists an on-duty device, the new joiner contacts the on-duty device directly to receive data bursts and the group contribution list. If nothing heard after a period of time, the new joiner itself becomes the on-duty device and sends packets to the common multicast IP address to guide future coming devices. If several devices join the cooperative network at the same time, since they all have zero contribution value, the device with the smallest MAC address takes the on-duty role for the first WMAN cycle. Again, this group initialization scheme is simple and practically sufficient for our proposed cooperative video streaming system over mobile devices. More sophisticated, and much more expensive, protocols would have been needed if we were to consider malicious devices with powerful resources. For example, authentication schemes may be needed to only allow legitimate devices. Also encryption of some fields such as burst start time would prevent malicious mobile devices from free riding by passively listening on the multicast group without contributing anything back to the group. Encryption and decryption of a few fields (only few bytes) do not impose too much overhead. There is no need to encrypt the video data itself.

4. ALGORITHM ANALYSIS

In this section, we analytically analyze the energy consumption of the proposed cooperative system and compare it against that of the current system. We use values obtained from actual devices to numerically analyze the derived equations under different network conditions. Table 1 contains all the symbols used in the analysis.

4.1 Analysis of Energy Consumption

Energy Consumption of Current Systems. We first study the energy consumption of mobile devices in current

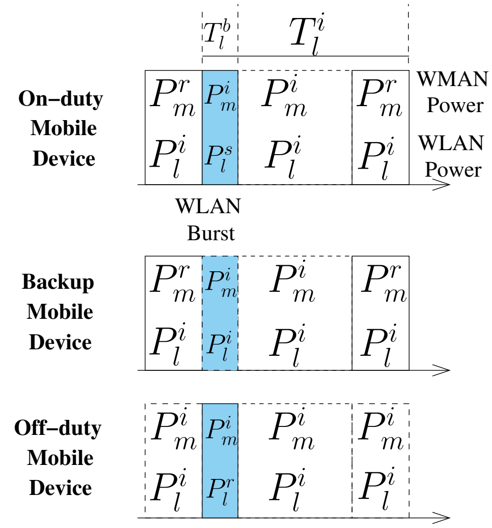

Figure 4: Burst transmission in the proposed cooperative WMAN and WLAN networks.

video broadcast systems, in which video streams are sent in bursts over the WMAN. Fig. 3 shows two bursts of the same video stream broadcast over a WMAN to mobile devices. Each mobile device receives a WMAN burst with length T_m^b sec, and then puts its WMAN interface into idle mode for T_m^i sec before reaching the next data burst. The WMAN interface has a receiving power consumption of P_m^r, and an idle power consumption of P_m^i. We note that the activation of the WMAN interface is not instantaneous because the WMAN interface needs to turn on its radio frequency (RF) circuit a little bit earlier to search for and tune into the broadcast frequency. We let T_m^0 sec be the WMAN burst overhead duration. Even though no data is transmitted during this period, the WMAN interface has to be on to get ready for the coming data. We compute the energy consumption of all mobile devices in the current system between two WMAN bursts as:

$$E_e = N\left(P_m^r(T_m^b + T_m^0) + P_m^i T_m^i\right) \text{ Joule} \quad (1)$$

Energy Consumption of the Proposed Cooperative System. We next study the energy consumption of mobile devices in the proposed cooperative system, where each mobile device has a WMAN interface as well as a WLAN interface. Fig. 4 shows the video data transmission over the WMAN and WLAN. Notice that the main difference between the proposed cooperative system and the current system is that the on-duty mobile device in the former relays the video data burst received from the WMAN over the WLAN to the off-duty mobile devices. These relayed bursts are shaded in Fig. 4. We let T_l^b sec be the burst time period and T_l^i sec be the idle time period in the WLAN. The WLAN interface has receiving power consumption of P_l^r, sending power consumption of P_l^s, and idle power consumption of P_l^i. In WMAN only network, all receivers experience the WMAN overhead T_m^0 sec. However, in the cooperative network, only the selected on-duty and backup devices need to spend energy on WMAN receiving preparation, which improves the energy saving metric. We also note that in our cooperative network, extra energy needs to be spent on the

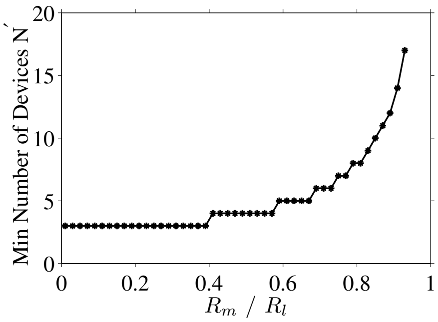

Figure 5: Minimum number of devices needed under different network speeds.

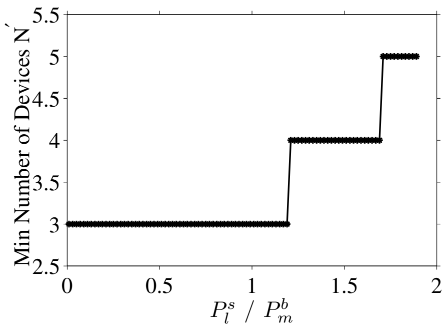

Figure 6: Minimum number of devices needed under different energy costs.

WLAN interfaces to organize the cooperative group a little bit earlier before the WMAN burst. As discussed in Section 3.2.2, we use Δ sec to denote this period. Using Fig. 4, we compute the energy consumption of on-duty, backup, and off-duty mobile devices between two WMAN bursts as follows:

An on-duty mobile device consumes:

$$P_m^r(T_m^b + T_m^0) + P_m^i T_m^i + P_l^s(T_l^b + \Delta) + P_l^i T_l^i \text{ Joule}$$

A backup mobile device consumes:

$$P_m^r(T_m^b + T_m^0) + P_m^i T_m^i + P_l^i T_l^b + P_l^i T_l^i + P_l^r \Delta \text{ Joule}$$

An off-duty mobile device consumes:

$$P_m^i T_m^b + P_m^i T_m^i + P_l^r T_l^b + P_l^i T_l^i \text{ Joule}$$

Then, we write the total energy consumption of the proposed system as:

$$E_c = \Bigg(P_m^r(T_m^b + T_m^0) + P_m^i T_m^i + P_l^s(T_l^b + \Delta) + P_l^i T_l^i\Bigg) + \Bigg(P_m^r(T_m^b + T_m^0) + P_m^i T_m^i + P_l^i T_l^b + P_l^i T_l^i + P_l^r \Delta\Bigg)K + \Bigg(P_m^i T_m^b + P_m^i T_m^i + P_l^r T_l^b + P_l^i T_l^i\Bigg)\Bigg(N - 1 - K\Bigg)\text{Joule} \quad (2)$$

Comparison of Energy Consumption. We derive the sufficient condition for the proposed cooperative system to outperform current systems. Observe that, since the energy consumption of the on-duty device is higher than that of the off-duty device, more off-duty devices lead to lower energy consumption. Therefore, we let N' be the minimum number of mobile devices for the proposed system to outperform current systems. Combining Eqs. (1) and (2), and rearranging the inequality, we get N' as:

$$N' = \Bigg\lceil \frac{(P_l^s - P_l^r + KP_l^i - KP_l^r)T_l^b + (P_l^s + KP_l^r)\Delta}{(P_m^r - P_m^i)T_m^b + P_m^r T_m^0 - P_l^r T_l^b - P_l^i T_l^i} + \frac{((K+1)(P_m^r T_m^b - P_m^i T_m^b + P_m^r T_m^0)}{(P_m^r - P_m^i)T_m^b + P_m^r T_m^0 - P_l^r T_l^b - P_l^i T_l^i} \Bigg\rceil \quad (3)$$

Note that keeping N' small is important, otherwise the proposed cooperative system may lead to worse performance than current systems *if* there are very few devices in the cooperative network.

Description	Sample value
WMAN receive	400 mW
WMAN idle	10 mW
WLAN send	592 mW
WLAN receive	375 mW
WLAN idle	2 mW

Table 2: Sample energy usage from [17] and [18].

4.2 Numerical Analysis

To analyze the energy usage of WMANs and WLANs, we consider the data sheets released by a popular DVB-H chip manufacturer [17], which indicate that the recent WMAN chips have a receiving power consumption of 400 mW, and the power consumption drops to 10 mW in idle mode. For the WLAN energy consumption, we consider the data sheets released by the Philips Low Power 802.11 WLAN solution [18], which indicate that the lowest power consumption in the standby mode is less than 2 mW. The receiver power consumption in the IEEE 802.11g WLAN is 375 mW. The transmit power at 15 dBm is 592 mW for the 802.11g WLAN. These sample values are listed in Table 2. Multiple parameters affect the value of N' and we evaluate their impacts in the following.

Transmission Speed Difference. The transmission speed difference in WMAN and WLAN can affect the values of T_m^b, T_m^i, T_l^b and T_l^i. We use R_m to denote the transmission speed of WMAN and R_l to denote the transmission speed of WLAN. To transmit the same amount of data, we have $R_m T_m^b = R_l T_l^b$. Since WMAN and WLAN use the same cycle period, we must have $T_m^b + T_m^i = T_l^b + T_l^i$. We vary the value of R_m/R_l and we use Eq. (3) to compute N' with the sample energy value listed in Table 2. We plot the results in Fig. 5. This figure shows that N' is fairly small for practical value of R_m/R_l. For example, when the WLAN rate is *only* two times faster than the WMAN rate ($R_m/R_l = 0.5$), the proposed cooperative system only needs 4 devices ($N' = 4$) to outperform current systems. This requirement is easy to meet in the real world.

Energy Consumption Difference. It is expected that hardware manufacturers will keep improving the energy consumption of their chips. P_m^b, P_l^s and P_l^r are the dominant energy parameters. We vary the value of P_m^b/P_l^s. We assume $P_l^r/P_l^s = 0.6$ which is similar to the sample ratio in Table 2. We also assume $R_m/R_l = 0.5$. We use Eq. (3) to compute N' with common network parameters and plot

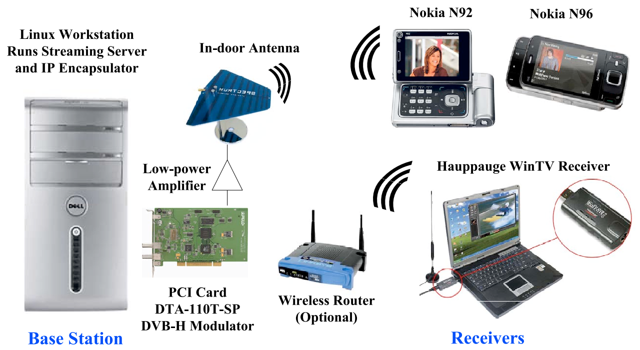

Figure 7: The DVB-H and WLAN Cooperative Testbed.

the results in Fig. 6. This figure shows that if the WLAN chip can reduce its energy cost level to about the same as WMAN chip, the proposed cooperative system only needs 3 devices ($N' = 3$) to outperform current system. Even with existing technology that the WLAN chip consumes about 50 percent more energy than WMAN chip, only 4 devices are needed, which is also easy to meet in practice.

5. EVALUATION IN A TESTBED

In this section, we first describe our WMAN and WLAN cooperative testbed setup. Then we use this testbed to evaluate the performance of our proposed system. We also show the robustness and the fairness of our system. In our test environment, we choose the DVB-H standard to stand for WMAN and the IEEE 802.11g to standard for the WLAN. All mobile devices in the cooperative network have both DVB-H and WLAN interfaces and use the same wireless access point.

5.1 Testbed Setup

We setup a DVB-H and WLAN cooperative testbed, which is shown in Fig. 7. This testbed generates valid DVB-H video streams and provides an IEEE 802.11g WLAN environment. It includes a base station, a data analyzer, several receivers and management tools. We describe each component in the following. Table. 3 lists all hardware components used in the testbed.

WMAN Base Station. We use a commodity Linux PC as our base station, in which we install an RF signal modulator card [19] to implement the physical layer of the DVB-H stack. This modulator card can modulate the MPEG-2 packets and transmit DVB-H signals. The RF output level of the modulator, however, is quite low ($\sim$ -29 dBm) and can only reach up to 1-meter broadcast range for receivers with 6 dB gain antenna. Mobile devices typically have antenna gains much lower than 6 dB. We use the amplifier available from Enensys [23] to boost the signal to about 0 dBm, which gives us approximately 20-meter range for cell phones in our lab environment. The amplifier is connected to the DVB-H compatible antenna LP49-DTV [24]. This PCI card comes with a software tool to control the OFDM modulator.

WLAN Access Point. The IEEE 802.11g based wireless router Linksys WRH54G is used to provide the WLAN environment. We enable the WLAN encryption to make sure only our selected test client devices can use the wireless bandwidth.

Hardware Components	Usage
Linux PC	DVB-H base station
DTA-110T card [19]	Physical layer of DVB-H
Enensys amplifier [23]	Boost the DVB-H signal
LP49-DTV [24]	DVB-H antenna
Linksys router	WLAN environment
DiviCatch analyzer [22]	Data analyzer
Nokia phones	Verify DVB-H streams
Hauppauge receiver [20]	USB based receiver

Table 3: Hardware components used in the mobile TV testbed.

Receivers. We use Nokia N96 and N92 cell phones to verify the output stream of our DVB-H testbed as well as assessing the video quality. We also use several USB based low-cost WinTV-NOVA-T receivers from Hauppauge [20] in our testbed. Each WinTV-NOVA-T USB receiver is connected to a PC to receive the DVB-H steams. The WinTV-NOVA-T receiver is designed for DVB-T reception but can still receive DVB-H steams by using several utilities from the Linux TV Project [21]. Thus we can perform DVB-H signal scan and tune into the right frequency of the DVB-H video stream. Some system environment parameters such as bandwidth, modulation scheme, PID of the MPEG-2 TS stream can also be obtained. By using these parameters, we develop an open-source application in Java that runs on Linux to handle data transmission. This application uses the libpcap library to capture the video stream based on its PID from the DVB-H network and can forward the data packets received from DVB-H via WLAN to other mobile devices in the same multicast group. Furthermore, when we capture the IP packets for a specific DVB-H channel, each packet is associated with a receiving timestamp. Based on the timestamps, together with DVB-H bursts information, we can get T_m^i, T_m^b, T_l^i and T_l^b for each cycle. Precise channel switching delay can also be detected. We implement our leader election algorithm on each receiver to determine the device role. The system self-forms cooperation group and calls the Java transmit application for data exchange based on the device role.

Data Analyzers. As the mobile TV application on the Nokia cell phone is proprietary, we can not use the cell phones to evaluate the performance metric in the testbed implementation. To address this shortcoming, we added the DVB-H Analyzer called DiviCatch available from [22] to the testbed. This analyzer can be attached to a PC via a USB port, and comes with a visualization software that runs on Microsoft Windows. The DiviCatch software also provides detailed real-time information on the RF signals, MPE frames, burst schedules, burst jitters and so on.

To conduct our experiments, we configure the testbed as follows. We setup an 8 MHz radio channel to broadcast four 5-minute long TV programs coded at 250 kbps. We use the QPSK modulation scheme together with the convolution coding rate at 2/3 and guard interval at 1/8. We have 4 WinTV-NOVA-T DVB-T receivers [20] in total.

5.2 Empirical Results

Potential Energy Saving. We first study the potential energy saving gain that can be achieved by using the proposed cooperative system. We define the energy saving

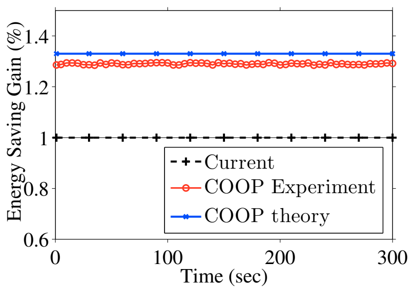

Figure 8: Energy saving gain in testbed experiment.

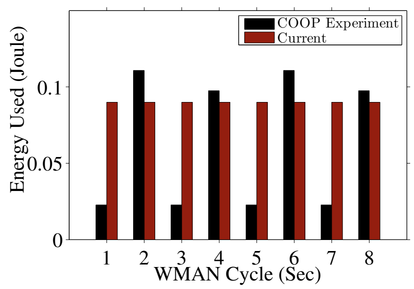

Figure 9: Energy consumption of one mobile device.

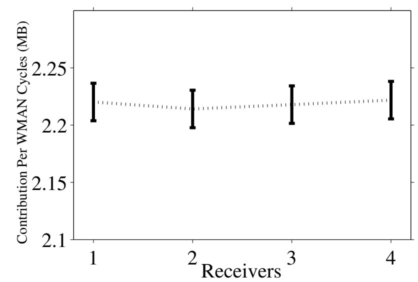

Figure 10: Contribution value per WMAN cycle of four receivers.

gain as $E_g = (E_e - E_c)/E_e$, where E_c is the total energy consumption of the proposed cooperative system, and E_e is that of current WMAN only system. We use 1 on-duty device, 1 backup device and 2 off-duty devices in the WMAN and WLAN cooperative system. We broadcast a 5-minute long video stream via the DVB-H testbed. Since we can get accurate T_m^i, T_m^b, T_m^0, Δ, T_l^i and T_l^b for each WMAN cycle as shown in Eqs. (1) and (2) on the receivers, after the video stream finishes, we can measure the total energy spent in the WMAN and WLAN cooperative system based on the sample energy consumption listed in Table 2. We broadcast the same video stream again from the DVB-H testbed and this time all 4 receivers are in the WMAN only mode. After the video stream finishes, we measure the total energy spent in the WMAN only system in the same way. We compare the experimental results against the results we get in theory by using Eqs. (1) and (2). The comparison plot is shown in Fig. 8. In theory, with 4 devices, we can save around 33%, and in the experiment we can save around 29%. This performance difference may due to the extra energy spent on turning on and off the network interfaces and handling all overhead messages composed in the real system. We have only four devices in our testbed, because of the limited hardware. Yet, our experiments show a gain of up to 29% in the energy saving. The gain will be much larger with more cooperative devices.

Energy Consumption of Individual Mobile Device. Next, we analyze the energy consumption of individual client devices. Using the above setup, we present sample results of client 1; all other results are similar. We calculate the energy used by the client in both cooperative mode and WMAN only mode in each WMAN time slicing cycle and plot the results in Fig. 9. We draw two observations on this figure. First, in our cooperative system, because of different roles taken by the client device in WMAN cycles, its energy usage in each cycle is different. When the device is on-duty, its energy consumption is about 23% more than in WMAN only mode. When the device serves as backup, its energy consumption is about 8% more than in WMAN only mode. However, when the device is off-duty, it can save about 75% of its energy, leading to the overall energy saving gain. Second, the client device switches roles with 25% possibility of being on-duty, 25% possibility of being backup and 50% possibility of being off-duty. Knowing that the total number of devices used in this experiment is 4, these results demonstrate the fairness of our proper functioning algorithm. This is because the high energy consumption which corresponds to the device being on-duty occurs every second WMAN cycle. Similarly, the low energy consumption which corresponds to the device being off-duty occurs every first and third WMAN cycles.

Fairness of the Proposed System. Finally, we consider the fairness of the proposed cooperative system in details. Fairness is important to mobile devices because they have stringent energy constraints, and any imbalance in the load may drive users away from the cooperative network. We measure the contribution of each device in the cooperative network throughout the whole experiments. The contribution value of a device is the number of bytes relayed by that device to other devices over the WLAN. The total contribution values for the four devices are 222.02 MB, 221.41 MB, 221.79 MB and 222.18 MB. These values confirm that all four receivers contributed almost equally during the experiments. Next we compute the average contribution in each WMAN cycle for all devices. We also compute the 95% confidence interval, which is computed as the average value plus/minus 1.96 × the value of the standard deviation for each device (1.96 is the ratio to determine 95% confidence interval). We plot the results in Fig. 10. The figure shows that for each device, the average contribution value per WMAN cycle falls into a small range. In addition, the range of each device is similar, which shows that the proposed cooperative system evenly balances the load on mobile devices.

6. EVALUATION USING TRACE-DRIVEN SIMULATION

In this section, we analyze the proposed cooperative system using simulation. The simulation has more devices than our testbed and it enables us to rigorously study the performance of the proposed system from several angles.

6.1 Simulation Setup

We have implemented a trace driven simulator that concurrently supports the WMAN and WLAN. We have implemented the proposed cooperative system in the simulator. For comparisons, we have also implemented the current system, which only uses the WMAN for broadcast. For realistic simulations, we have studied a transport stream generated by a Nokia mobile TV base station. This transport stream consists of four video channels for five minutes each, where each channel is encoded at 450 kbps using H.264/AVC video

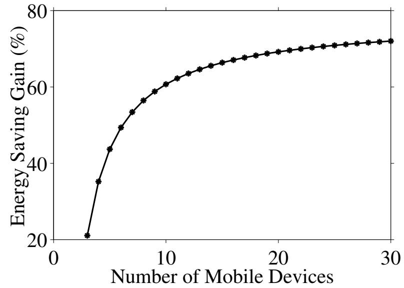

Figure 11: Potential energy saving gain in simulator.

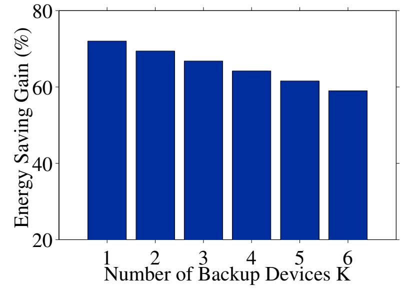

Figure 12: Implications of different number of backup devices on energy saving.

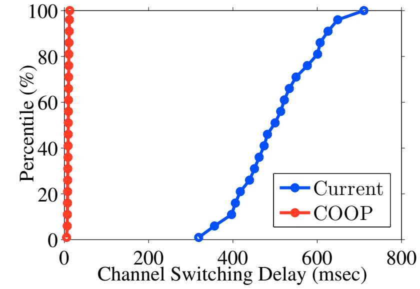

Figure 13: Channel switching delay gain in simulator.

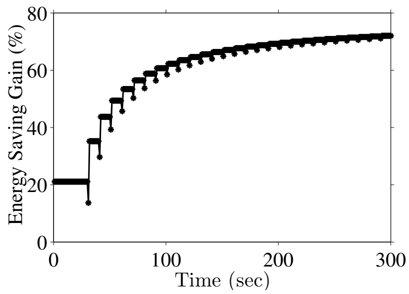

Figure 14: Energy saving gain under network dynamics: new devices join.

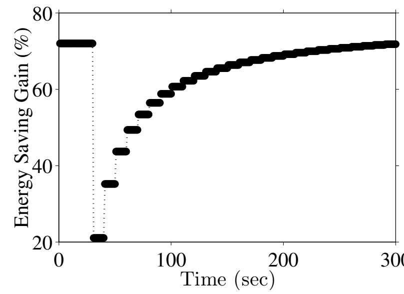

Figure 15: Energy saving gain under network dynamics: failed and rejoined devices.

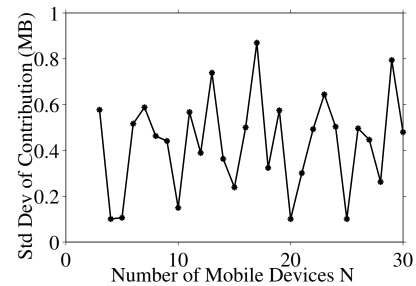

Figure 16: Fairness of the proposed system.

codec and AAC audio codec. We have developed a software utility to analyze the transport stream and create a log file for each video channel. The log files indicate the start time of each burst and its size. We have conducted the simulations with all four video streams. We present sample results of video stream 1; all other results are similar. We still use the sample energy consumption listed in Table 2.

6.2 Simulation Results

Potential Energy Saving. We vary the total number of mobile devices from $N = 3$ to 30. We compute the average energy saving gain E_g and plot the results in Fig. 11. We draw two observations on this figure. First, the energy saving gain increases as the number of mobile devices increases and can be as high as 70% with a total of 30 devices. Second, the proposed system outperforms the current systems if $N \geq 3$. For example, with only three mobile devices, the proposed cooperative system saves more than 21% energy compared to current systems. Note that the energy saving gain is relative to what can be achieved by the current system and not absolute. Since the current systems employ time slicing schemes to save energy, the absolute energy saving achieved by our proposed cooperative system will be substantial. This is improtant to prolong the battery life.

Impact of Backup Devices on Energy Saving. We turn on 30 mobile devices that view the same video stream for five minutes. We vary the number of backup devices from 1 to 6. We compute the average energy saving gain E_g under different K values and plot the results in Fig. 12. The figure shows that while more backup devices lead to higher robustness, more backup devices also reduce the energy saving gain achieved by the proposed system. Despite the *tradeoff* between robustness and energy saving, the energy saving gain achieved by the proposed system does *not* dramatically drop when the number of backup devices increases. As illustrated in this figure, even when devoting 1/5 of mobile devices to be backup devices, the energy saving gain is still higher than 58%.

Switching Delay Reduction. Next, we study the channel switching delay under user dynamics. We broadcast four video streams for five minutes. For each video stream we choose 1 on-duty mobile device and 1 backup mobile device. We turn on 120 mobile devices to watch one of the four video streams randomly. Initially, each video stream has 30 mobile devices. We generate random channel switching events using Bernoulli trials, and we set the probability of success in a way that users stay with one video stream for 30 sec on average. For each channel switching event, the mobile device randomly selects a new video stream other than the currently viewed one. We derive the channel switching delay by measuring the time difference between when a user decides to switch channel until he/she gets the data for the new video stream. We then compute the average channel switching delay of each mobile device, and we plot the CDF curves for the current and the proposed systems in Fig. 13. This figure illustrates that the proposed cooperative system effectively eliminates the channel switching delay: from up to 700 msec in current systems to at most 13 msec in the proposed system. That is to say, our cooperative system can reduce the channel switching delay by up to 98%.

Impact of New Joiners on Energy Saving. We study the impact of energy saving when mobile devices dynamically join the cooperative network. Initially we turn on 3 mobile devices that view the same video stream for five minutes. We configure the system to have 1 on-duty and 1 backup mobile devices. After the first 30 seconds, we gradually add mobile devices to the group in order to emulate that new users choose to watch this video stream. More precisely, we insert a mobile device every 10 seconds such that by the end of the 5-minute video stream, we have 30 mobile devices in total. We compute the average energy saving gain E_g, and we plot the results in Fig. 14. We draw two observations on this figure. First, for the particular round when new device joins in the group, extra energy are used to provide the data packets to the new joiners. Second, after that round, the cooperative group can reorganize itself properly to achieve even better energy saving due to the bigger group size.

Fail-safe Mechanism under Device Failures. We study the effectiveness of the fail-safe mechanisms under mobile device failure. We turn on 30 mobile devices that view the same video stream for five minutes. We configure the system to have 1 on-duty and 1 backup mobile devices, and instruct the simulator to fail 90% of the mobile devices after 30 seconds. In addition, we gradually add these failed mobile devices back in order to emulate that the users either reboot their mobile devices or adjust the antennas for better reception. We insert a mobile device every ten seconds. We compute the average energy saving gain E_g, and we plot the results in Fig. 15. The figure shows that the proposed cooperative system is robust, because it survives a sudden loss of 90% of mobile devices, despite an energy saving hit of about 51%. The proposed cooperative system also quickly adapts to the network dynamics, because once the failed mobile devices rejoin the cooperative network, the energy saving increases and finally can reach the original level with the same number of devices in the group.

Fairness of the Proposed System. Next, we consider the fairness of the proposed cooperative system. We turn on N mobile devices that view the same video stream for five minutes. We vary the value of N from 3 to 30. We measure the contribution (in terms of number of bytes) of each mobile device. For each N value, we compute the standard deviation of the contribution among all mobile devices, and we plot the results in Fig. 16. This figure shows that the standard deviation of contribution is always less than 0.6 MB. The figure clearly shows that the proposed cooperative system evenly balances load on mobile devices, given that the total contribution is in the order of several hundred MBs.

6.3 Evaluation of the Ad-hoc Mode

The simulations in the previous section use the infrastructure mode of the WLAN in the proposed cooperative system. In this section, we evaluate the cooperative system under the WLAN ad-hoc mode. We implement another simulator to do this. We use the video stream from Nokia and the sample energy consumption values in Table 2. We setup a 10m × 10m area for all devices. The WLAN coverage for each device is limited to a radius of 2m. We assume that within this WLAN coverage, the WLAN transmission rate remains the same. Two devices that are more than 2m apart cannot cooperate directly. Each device can move with a speed of 0.2m/s in any direction within the defined area. We make this estimation as in real world users using WLAN ad-hoc mode to watch video programs on mobile devices usually are not moving fast while watching video programs.

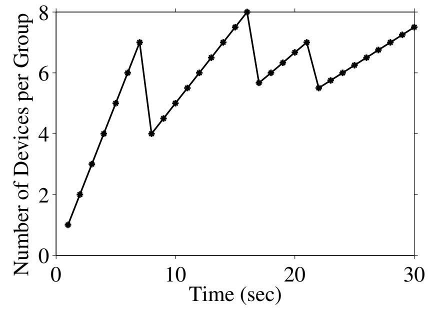

Figure 17: Average number of devices per group.

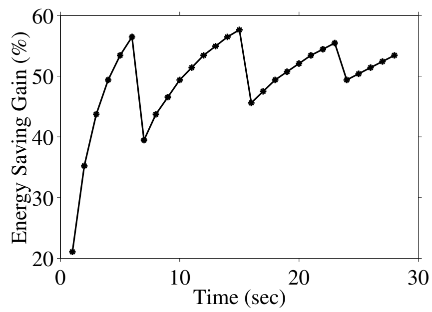

Figure 18: Energy saving gain in ad-hoc mode.

We first study the dynamics of group formation and the average number of devices in each group as devices move and change their locations. We use N_g to denote the number of groups in the defined area and N_d to denote the total number of devices in the defined area. We calculate N_d/N_g every second. We configure the system to add 1 device per second to the center of the defined area until there are 30 devices in total. Each device can move at any direction with the speed of 0.2m/s. We plot the results in Fig. 17. The figure clearly shows the dynamics of the system, as the number of groups changes because of mobility. These results confirm that our cooperative system can properly function in the ad-hoc mode and it can adapt to user mobility.

Next, we study the energy saving gain in the ad-hoc mode. We initially put 3 devices in the center of the defined area. We configure the system to add 1 device per second to the center of the defined area until there are 30 devices in total. Each device can move at any direction with the speed of 0.2m/s. Every second, we calculate the potential energy saving gain. In each cooperative group, we use 1 on-duty and 1 backup devices. We plot the results in Fig. 18. The figure clearly shows that in the ad-hoc mode, our cooperative system can still have energy saving gain as high as 57% with a total of 30 devices. Notice that the fluctuations in the energy saving are due to device mobility and re-formation of the cooperative groups.

7. CONCLUSIONS

We studied the problem of broadcasting video streams over a WMAN to mobile devices. We proposed a cooperative system in which several mobile devices *share* received video data bursts over a WLAN. The proposed system reduces the energy consumption and significantly reduces the channel switching delay. We analytically showed that the proposed cooperative system outperforms current system in terms of the energy consumption with only few cooperative devices.

We presented a simple distributed leader election algorithm to choose the on-duty and backup devices in the cooperative network. The proposed algorithm imposes very little computation and communication overheads. Moreover the proposed algorithm enables truthful cooperation among devices because it distributes the load by rotating the on-duty role across all devices. The on-duty device is elected based on previous contributions, where the contribution of a device is not computed by itself but computed independently by all other devices based on the actual data received from the on-duty device. Therefore the room for a device to cheat or free-ride is very limited.

Our experimental results from the real testbed showed that the proposed cooperative system is promising, because it achieves high energy saving, significantly reduces the channel switching delay and uniformly distributes load on all mobile devices. Furthermore, we complemented our evaluation in the testbed by several simulations that address larger networks, device failures, departures and mobility. Our simulations confirm the viability of the proposed cooperative system and the significant energy saving potential that can be achieved by using our simple algorithm.

8. REFERENCES

[1] "Digital Video Broadcasting (DVB); framing structure, channel coding and modulation for digital terrestrial television. European Telecommunications Standards Institute (ETSI) Standard EN 300 744 Ver. 1.5.1," June 2004.

[2] G. May, "The IP Datacast system - overview and mobility aspects," in *Proc. of IEEE International Symposium on Consumer Electronics (ISCE'04)*, Reading, UK, September 2004, pp. 509–514.

[3] 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.

[4] "Digital Video Broadcasting (DVB); IP datacast over DVB-H: Content delivery protocol. European Telecommunications Standards Institute (ETSI) Standard EN 102 472 Ver. 1.1.1," November 2006.

[5] X. Pei, G. Zhu, Q. Wang, G. Liu, and W. Yuan, "Cooperative load-balancing strategy based on N-hops routing algorithm of ad hoc network in heterogeneous overlapping networks," *Annals of Telecommunications*, pp. 597–606, November 2008.

[6] Y. Liu, "Video Streaming over Cooperative Wireless Network," Master's thesis, Simon Fraser University, Canada, December 2009.

[7] D. Niyato and E. Hossain, "A hierarchical model for bandwidth management and admission control in integrated IEEE 802.16/802.11 wireless networks," in *Proc. of IEEE Wireless Communications and Networking Conference (WCNC'07)*, Hong Kong, China, March 2007, pp. 3763–3767.

[8] H. Pan, S. Yamano, and H. Tani, "On the effectiveness of integrating WiMAX and Wi-Fi," in *International Journal of Communication Networks and Distributed Systems*, vol. 2, November 2009, pp. 198–213.

[9] X. Yang, Y. Song, T. Owens, J. Cosmas, and T. Itagaki, "Performance analysis of time slicing in DVB-H," in *Proc. of Joint IST Workshop on Mobile Future and Symposium on Trends in Communications (SympoTIC'04)*, Bratislava, Slovakia, October 2004, pp. 183–186.

[10] E. Balaguer, F. Fitzek, O. Olsen, and M. Gade, "Performance evaluation of power saving strategies for DVB-H services using adaptive MPE-FEC decoding," in *Proc. of IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'05)*, Berlin, Germany, September 2005, pp. 2221–2226.

[11] M. Hefeeda and C. Hsu, "Energy optimization in mobile TV broadcast networks," in *Proc. of IEEE Innovations in Information Technology (Innovations'08)*, Al Ain, United Arab Emirates, December 2008, pp. 430–434.

[12] C. Hsu and M. Hefeeda, "Time slicing in mobile TV broadcast networks with arbitrary channel bit rates," in *Proc. of IEEE INFOCOM'09*, Rio de Janeiro, Brazil, April 2008.

[13] M. Rezaei, I. Bouazizi, and M. Gabbouj, "Joint video coding and statistical multiplexing for broadcasting over DVB-H channels," *IEEE Transactions on Multimedia*, vol. 10, no. 7, pp. 1455–1464, December 2008.

[14] C. Hsu and M. Hefeeda, "Bounding switching delay in mobile TV broadcast networks," in *Proc. of ACM/SPIE Multimedia Computing and Networking (MMCN'09)*, San Jose, CA, January 2009.

[15] S. Deering, D. Estrin, D. Farinacci, V. Jacobson, C. Liu, and L. Wei, "The pim architecture for wide-area multicast routing," in *IEEE/ACM Transactions on Networking*, April 1996.

[16] S. Deering and D. Cheriton, "Multicast routing in datagram internetworks and extended lans," in *ACM Transactions on Computer Systems*, Hong Kong, China, April 1990, pp. 85–110.

[17] "Evolution of DVB-T front-end receivers through integration," June 2007, `http://www.dibcom.info/Images/Upload/pdf/whitePaper2_integration_MD_V2.pdf`.

[18] "Philips BGW211 low-power 802.11 WLAN chip," April 2005, `http://www.mt-system.ru/documents/bgw211.pdf`.

[19] "Dektec DTA-110T PCI modulator," 2009, `http://www.dektec.com/Products/DTA-110T/`.

[20] "WinTV-NOVA-T USB dvb-t receiver," 2009, `http://www.hauppauge.co.uk/`.

[21] "Linux TV project, television with linux," 2009, `http://www.linuxtv.org/`.

[22] Divi Catch RF-T/H transport stream analyzer, 2009. `http://www.enensys.com/`.

[23] Enensys 1-watt RF power amplifier, 2009. `http://www.enensys.com/`.

[24] Spectrum LP49 DTV indoor antenna, 2009. `http://spectrum.co.kr/`.