

國立交通大學

資訊科學與工程研究所

博士論文

行動廣播網路之資料復原研究

The Data Recovery Issue in Mobile Broadcast Networks



研究生：楊文新

指導教授：曾煜棋、林寶樹 教授

中華民國一百年八月

行動廣播網路之資料復原研究

The Data Recovery Issue in Mobile Broadcast Networks

研究生：楊文新

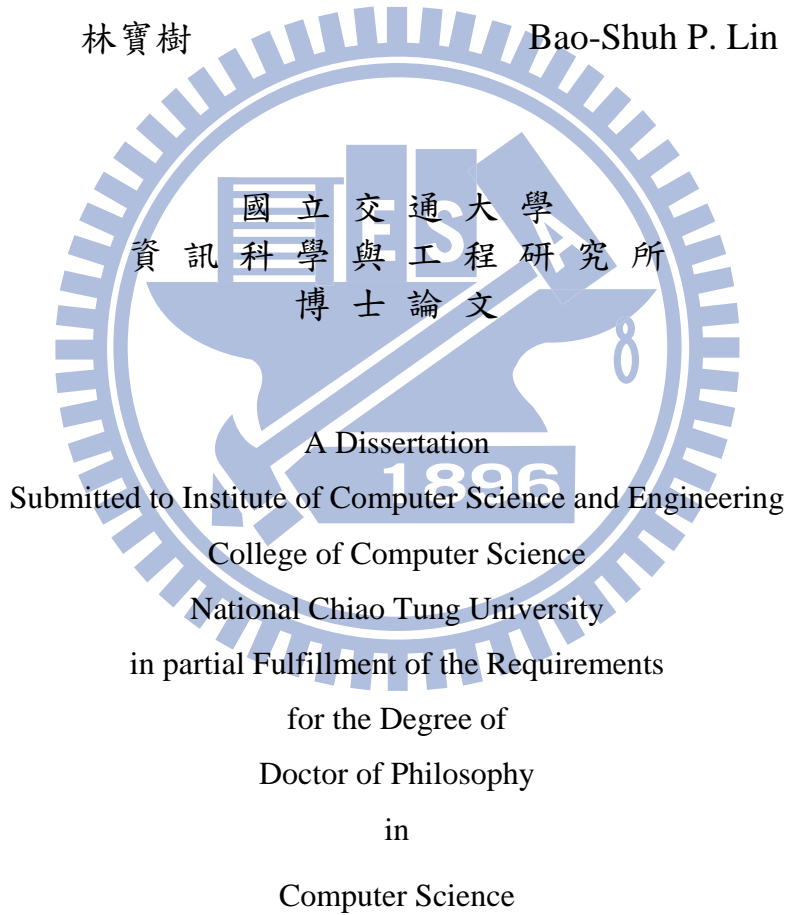
Student : Wen-Hsin Yang

指導教授：曾煜棋

Advisor : Yu-Chee Tseng

林寶樹

Bao-Shuh P. Lin



August 2011

Hsinchu, Taiwan, Republic of China

中華民國 一 百 年 八 月

行動廣播網路之資料復原研究

研究生：楊文新

指導教授：曾煜棋 博士
林寶樹 博士

國立交通大學資訊科學與工程研究所

摘 要

DVB-H (digital video broadcasting-handheld)及 DVB-IPDC (IP datacast over DVB-H) 近年來已發展成數位行動廣播服務的規格。其中 DVB-H 提供為手持行動裝置上進行數位視訊廣播，而 DVB-IPDC 則可結合另外的無線網路(如 IP 中繼網路)，進行 DVB-H 廣播資料遺失補償處理。本文以採用無線寬頻網路 WiMAX 及 WiFi 之 DVB-IPDC 架構基礎，由滿足行動使用者服務品質的需求觀點，研究行動廣播網路之資料復原議題。研究項目分為：一、討論透過復原網路進行行動廣播資料復原的方法；二、討論行動裝置在復原網路內移動產生換手的處理方法。以下分別說明。

首先透過復原網路進行行動廣播資料復原方法的研究，討論 DVB-H 廣播內容資料遺失或錯誤時，由一個 IP 中繼網路協助進行有效率的資料重送處理。提出 GPL (group packet loss)及 BDH (broadcast data handover)兩個問題。GPL 問題發生於重送相同的廣播資料需求，由大量的行動裝置提出，而這些行動裝置散佈於鄰近位置呈現地域性關係，或者重送廣播資料具有時間連續性的關係。BDH 問題則發生於行動裝置在之前所在的服務網路細胞範圍內提出重送廣播資料需求後，因為移動性換手到新的服務網路細胞範圍，而重覆提出重送廣播資料需求。為了解決 GPL 及 BDH 問題，本文提出透過發掘時間與空間相互關連性之 lazy wait 及 group acknowledgement 方法與機制，以減少重覆提出重送廣播資料的需求數量，也可以避免行動裝置因為換手重覆提出重送廣播資料需求。此外，系統根據行動裝置上的 DVB-H 接收訊號品質進行數學分析，動態調整系統參數 lazy wait 計時器(timer)及 group acknowledge 回應判斷條件臨界值。模擬結果顯示所提方法可以達到減少重送廣播資料需求及重送廣播資料數量的效果，並可減輕造成 IP 中繼網路擁擠的狀況。

其次討論 DVB-H 與 WiMAX 行動廣播網路整合架構下，利用網路編碼進行廣播資料隨需(on-demand)復原方法的研究。此時行動裝置可以保持接收 DVB-H 廣播視訊內容，同時使用 WiMAX 網路做為復原管道進行重送廣播資料。此復原管道建模成一個提送時段接著一個復原時段的重覆訊框，行動裝置在提送時段內提出重送廣播資料需求到

行動廣播伺服器，在復原時段內需重送的錯誤或遺失廣播資料則透過 WiMAX 網路傳遞到行動裝置。本文提出 XOR 編碼基礎的優先權網路編碼方法以達到復原更多具時效性廣播資料的目標，包括 RGS (recovery by greedy selection)及 RPS (recovery by prioritized selection)兩種方法。RGS 透過建構權重二分圖來表達編碼資料封包(coded packet)對資料復原的影響程度，採用貪婪策略找尋最大權重最小覆蓋組合(maximum weight minimum dominating set)的編碼資料封包組。RPS 延伸 RGS 方法，考量重送錯誤或遺失廣播資料與提出之重送廣播資料需求兩者之間數量的比重關係，以及編碼資料封包的重送順序兩項因素，以樹狀回溯模式進行權重調整並找尋最大權重最小覆蓋組合的編碼資料封包組。模擬結果顯示所提方法可以提升錯誤或遺失廣播資料之復原效率，並可減少需重送廣播資料因為逾越有效時限而無法復原的數量。

最後行動裝置在 WiMAX 與 WiFi 整合的復原網路內移動產生換手問題的研究。此復原網路中之 WiFi 網路雖然已經廣佈於大樓及公共空間區域提供無線上網，但其信號覆蓋區域有限，無法提供行動裝置於移動中之無縫連結要求。因此透過具有較大信號覆蓋區域的 WiMAX 網路來協助行動裝置進行換手處理，其中節省電能為一重要議題。本文根據行動裝置過去的換手紀錄，提出 HGMA (handover scheme with geographic mobility awareness)方法，透過三個方式達到節省電能目標。一、透過度量行動裝置對網路的接收訊號強度及其平均移動速度，避免行動裝置觸發不必要的換手需求。二、換手時利用 HCS (handover candidate selection)篩選候選網路組，以減少掃描網路接取點而節省電能。三、增加使用原先連結的 WiFi 或 WiMAX 網路類型接取點機會，減少因為使用不同類型網路切換通訊介面的頻率以節省電能。HGMA 也可以讓行動裝置換手時較偏好於使用 WiFi 網路及獲得頻寬保證。模擬結果為行動裝置執行換手時可以節省 59~80% 電量消耗，獲得 16~62% 的更多機會使用 WiFi 網路，同時增加 20~61% 的機率滿足應用頻寬需求的服務品質要求。

總結，本文以結合無線寬頻網路的 DVB-IPDC 架構，研究行動廣播內容資料遺失或錯誤的資料復原處理方法。透過復原網路進行有效率的廣播資料重送，同時行動裝置在 WiMAX 與 WiFi 整合的復原網路間換手時耗損較低的電能。

關鍵字：廣播網路、無線網路、復原網路、資料復原、網路編碼、網路選擇、換手、能源效率、DVB-H、DVB-IPDC、WiFi、WiMAX

The Data Recovery Issue in Mobile Broadcast Networks

Student: Wen-Hsin Yang

Advisor: Prof. Yu-Chee Tseng

Prof. Bao-Shuh P. Lin

Institute of Computer Science and Engineering

National Chiao Tung University

ABSTRACT

Recently, *DVB-H* (*digital video broadcasting–handheld*) and *DVB-IPDC* (*IP datacast over DVB-H*) have been developed to support mobile broadcasting services. *DVB-H* is designed to support digital video broadcast for handheld devices, while *DVB-IPDC* tries to cooperate with an wireless network such as IP-relay network to complement the data loss problem in *DVB-H*. Assuming that some wireless broadband networks such as WiMAX and WiFi networks are adopted to support *DVB-IPDC*, this dissertation will study the *data recovery* issue in the mobile broadcast network from the perspective of satisfying quality of service to mobile users. In particular, we break down two genres of tasks: one genre of tasks is to conduct the data recovery through a recovery network, the other genre of tasks is to discuss recovery network selection for the mobile devices.

For conducting the data recovery through a recovery network, we first discuss the message-efficient data recovery for *DVB-H* data losses through an IP-relay network, which points out two critical problems: *group packet loss (GPL)* and *broadcast data handover (BDH)*. *GPL* occurs when there is a burst of retransmission requests for the same pieces of data with high spatial or temporal correlation. *BDH* happens when some devices that made the above requests handover to new serving cells. To solve these problems, we propose a *lazy wait* and a *group acknowledgement* schemes to alleviate duplicate requests by exploiting their spatial and temporal correlation. This not only reduces the

requests submitted by neighboring devices in both space and time domains, but also avoids handovering devices from sending duplicate requests in new cells. Through mathematical analysis, we show how to adaptively adjust the timers of lazy wait and group acknowledgement based on channel quality. Simulation results prove that our schemes can efficiently reduce retransmission requests and retransmission packets, thus alleviating congestion in the IP-relay network.

Then, we consider the on-demand data recovery by network coding for mobile broadcasting systems, in which studies an integrated DVB-H and WiMAX mobile broadcasting architecture. DVB-H continually broadcasts videos to mobile devices. WiMAX serves as a recovery channel for mobile devices to request for retransmissions, which is modeled by repetitive frames, each with a *submission period* followed by a *recovery period*. Mobile devices submit their requests in the submission periods while the WiMAX network sends the lost packets in the recovery periods. To recover more urgent packets first, we develop two prioritized network coding schemes based on XOR coding. The first *recovery by greedy selection (RGS)* scheme constructs a weighted bipartite graph to reflect the influence of each coded packet and then adopts a greedy strategy to find a minimum dominating set of coded packets. By extending the similar concept to cope with the relation between requests submissions and packets losses as well as the retransmission sequence of coded packets, the second *recovery by prioritized selection (RPS)* scheme adopts a tree traversal approach to adjust weights and find a minimal set of coded packets from the graph. Simulation results verify that both RGS and RPS can improve recovery efficiency while reduce the number of packets being dropped due to missing deadlines.

For the recovery network selection for the mobile devices, we discuss the energy-efficient concern in an integrated WiMAX and WiFi network for the DVB-H mobile devices. To provide wireless Internet access, WiFi networks have been deployed in many regions such as buildings and campuses. However, WiFi networks are still insufficient to support ubiquitous wireless service due to their narrow coverage. One possibility to resolve this deficiency is to integrate WiFi networks with the wide-range WiMAX networks. Under such an integrated WiMAX and WiFi network, how to conduct energy-efficient

handovers is a critical issue. In this dissertation, we propose a *handover scheme with geographic mobility awareness (HGMA)*, which considers the historical handover patterns of mobile devices. HGMA can conserve the energy of handovering devices from three aspects. Firstly, it prevents mobile devices from triggering unnecessary handovers according to their received signal strength and moving speeds. Secondly, it contains a *handover candidate selection (HCS) method* for mobile devices to intelligently select a subset of WiFi access points or WiMAX relay stations to be scanned. Therefore, mobile devices can reduce their network scanning and thus save their energy. Thirdly, HGMA prefers mobile devices staying in their original WiMAX or WiFi networks. This can prevent mobile devices from consuming too much energy on interface switching. In addition, HGMA prefers the low-tier WiFi network over the WiMAX network and guarantees the bandwidth requirements of handovering devices. Simulation results show that HGMA can save about 59% to 80% of energy consumption of a handover operation, make mobile devices to associate with WiFi networks with 16% to 62% more probabilities, and increase about 20% to 61% of QoS satisfaction ratio to handovering devices.

In short, we pursue data recovery schemes to the mobile devices for DVB-H data losses by partnering with wireless broadband networks, which are able to do efficient data retransmissions for error recovery via a recovery network and achieve less energy consumption on the mobile devices for handovering between WiMAX and WiFi networks.

Keywords: broadcast, data recovery, DVB-H, DVB-IPDC, energy efficiency, handover, network coding, network selection, recovery network, WiFi, WiMAX, wireless network.

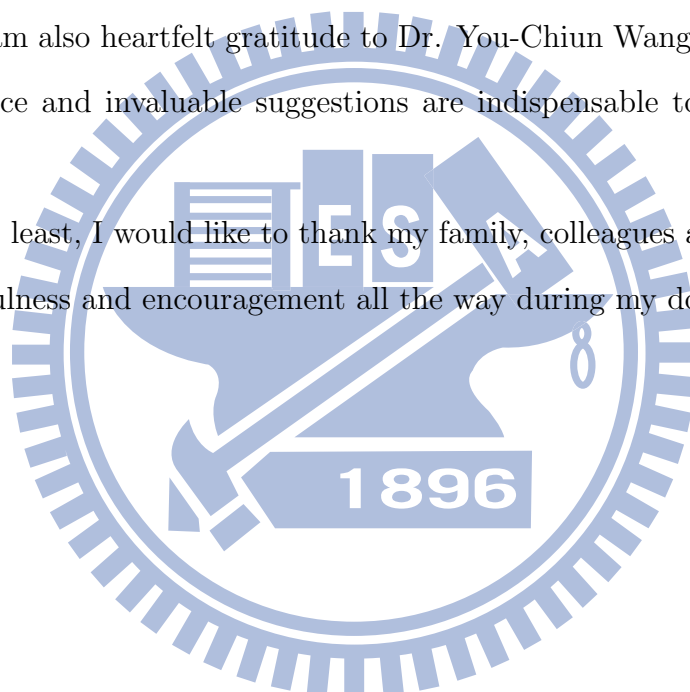
Acknowledgements

This dissertation could not be finished without the unconditional help and priceless support of many people and I will always bear them in mind.

At the very first, I am honored to express my deepest gratitude to my advisors, Prof. Yu-Chee Tseng and Prof. Bao-Shuh Paul Lin, with whose able guidance I could have worked out this dissertation. They have offered me valuable ideas, suggestions and criticisms with their profound knowledge and rich research experience. I am very much obliged to their efforts of helping me complete the dissertation.

In addition, I am also heartfelt gratitude to Dr. You-Chiun Wang, whose patient and meticulous guidance and invaluable suggestions are indispensable to the completion of this dissertation.

At last but not least, I would like to thank my family, colleagues and friends for their support, thoughtfulness and encouragement all the way during my doctoral study.

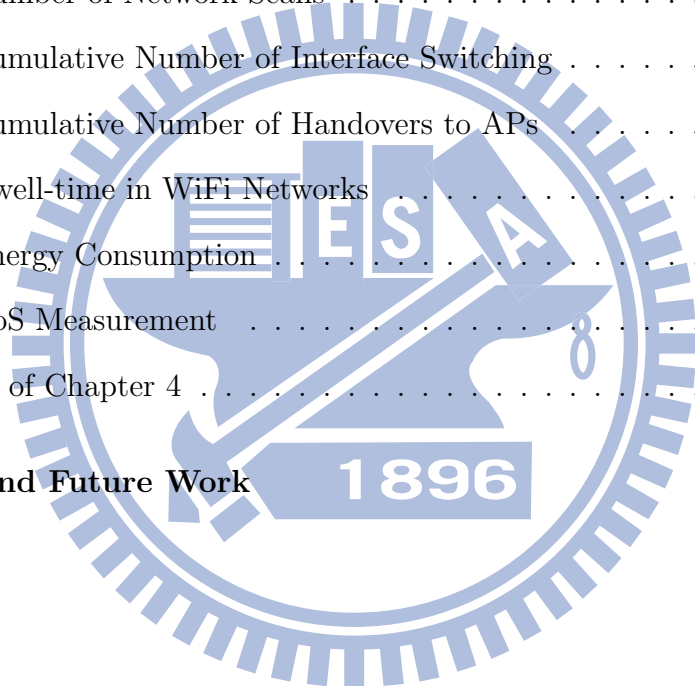


Contents

Abstract	i
Acknowledgements	vi
Contents	vii
List of Figures	x
List of Tables	xii
1 Introduction	1
1.1 Motivation	1
1.2 Background of the Dissertation	3
1.2.1 Digital Video Broadcasting – Handheld (DVB-H)	3
1.2.2 IP Datacast over DVB-H (DVB-IPDC)	4
1.2.3 Error Recovery in DVB-H Systems	8
1.2.4 Vertical Handover in Wireless Networks	13
1.3 Contributions of the Dissertation	14
1.4 Organization of the Dissertation	16
2 Data Recovery through an IP-Relay Network	18
2.1 Introduction	18
2.2 Preliminaries	21
2.2.1 Problem Statements	21

2.2.2	Related Work	22
2.3	Framework of the BR-LW Scheme	23
2.3.1	Operations of MDs	24
2.3.2	Operations of RSs	27
2.3.3	Operations of BSs	30
2.4	Determining System Parameters of BR-LW	30
2.4.1	Adjustment of lazy-wait timer t_{lazy}	30
2.4.2	Adjustment of threshold δ_{num}	33
2.4.3	Adjustment of threshold δ_{var}	35
2.5	Performance Evaluation	37
2.5.1	Ratio of Inhibited RREQs	39
2.5.2	Reduction of Packet Retransmissions	41
2.5.3	Adjustment of δ_{var}	43
2.5.4	Effect of Handovers	45
2.6	Summary of Chapter 2	46
3	Data Recovery by Using On-Demand Network Coding	48
3.1	Introduction	48
3.2	Related Work	50
3.2.1	Data Recovery in DVB-IPDC Systems	51
3.2.2	Network Coding for Content Distribution and Streaming Services	51
3.2.3	Network Coding for Error Recovery	52
3.3	Problem Definition	53
3.4	The Proposed Solutions	54
3.4.1	Recovery by Greedy Selection (RGS)	54
3.4.2	Recovery by Prioritized Selection (RPS)	58
3.5	Experiment Results	60
3.5.1	Simulation Model	60
3.5.2	Simulation Result	61

3.6	Summary of Chapter 3	64
4	Recovery Network Selection in WiMAX and WiFi Networks	66
4.1	Introduction	66
4.2	The Integrated WiMAX and WiFi Network Architecture	70
4.3	The HGMA Scheme	71
4.3.1	When to Trigger a Handover	74
4.3.2	The HCS Method	75
4.4	Experimental Results	77
4.4.1	Number of Network Scans	79
4.4.2	Cumulative Number of Interface Switching	80
4.4.3	Cumulative Number of Handovers to APs	81
4.4.4	Dwell-time in WiFi Networks	81
4.4.5	Energy Consumption	83
4.4.6	QoS Measurement	85
4.5	Summary of Chapter 4	86
5	Conclusion and Future Work	88
	Bibliography	90



List of Figures

1.1	A wireless broadband network overlaps to a mobile broadcasting network.	2
1.2	Structure of the research works in this dissertation.	3
1.3	The time slicing technique used in a DVB-H system.	4
1.4	An example of the coexistence of DVB-T and DVB-H data streams in a common channel.	5
1.5	The DVB-IPDC protocol stack.	6
1.6	The system architecture of DVB-IPDC.	7
1.7	The perspective of layered structure for DVB-IPDC and DVB-H.	9
1.8	The MPE-FEC frame used in the LL-FEC mechanism.	11
1.9	Handover in WiMAX and WiFi Networks.	14
2.1	The DVB-IPDC architecture with WiMAX networks as the datacast channel.	19
2.2	The flowchart of BR-LW.	25
2.3	The ratio of inhibited RREQs by BR-LW under different distributions of \mathcal{F} and \mathcal{L}	40
2.4	The amount of packet retransmissions by BR-LW under different distributions of \mathcal{F} and \mathcal{L}	43
2.5	The time when $c_{\text{var}} \geq \delta_{\text{var}}$ occurs and the hit rate of δ_{var} in the spatially-middle scenario of \mathcal{L}	44
2.6	The time when $c_{\text{var}} \geq \delta_{\text{var}}$ occurs and the hit rate of δ_{var} in the spatially-late scenario of \mathcal{L}	45
2.7	The hit rate of sending g_{ack} (due to $c_{\text{var}} \geq \delta_{\text{var}}$) when there are handovers.	46

3.1	A DVB-IPDC system with WiMAX networks as the recovery channel. . . .	49
3.2	An example of RGS: (a) the received RREQs, where ‘○’ means a successful reception while ‘×’ means an erroneous reception and (b) the weighted bipartite graph, where a rectangular vertex belongs to \mathcal{C} while a circular vertex belongs to \mathcal{Q}	56
3.3	The changes of overlapped recovery rates respecting to different retransmission time durations while the packet loss rate is 50%.	62
3.4	The changes of overlapped recovery rates respecting to different percentage of packet loss while the retransmission time is fixed at 10% of \mathcal{T}_{sub}	63
4.1	An integrated WiMAX and WiFi network.	67
4.2	The HGMA scheme.	72
4.3	Checking procedures to trigger a handover event.	75
4.4	Scenarios of three mobility models.	78
4.5	Average number of scanned APs in different models.	79
4.6	Cumulative number of interface switching in different models.	80
4.7	Cumulative number of handovers to APs in different models.	82
4.8	Average dwell-time of MDs in WiFi networks during one hour in different models.	83
4.9	Average energy consumption of MDs to conduct a handover operation in different models.	84
4.10	QoS-satisfied handover ratios in different models.	86

List of Tables

2.1	Notations used in Chapter 2.	24
3.1	Overlapped recovery on varying retransmissions time.	63
3.2	Overlapped recovery under different packet loss rates.	64
4.1	Notations used in Chapter 4.	71
4.2	Reduction of the number of scanned APs by HGMA.	80
4.3	Reduction of the number of interface switching by HGMA.	81
4.4	Improvement of the number of handovers to APs by HGMA.	82
4.5	Improvement of average dwell-time of MDs in WiFi networks by HGMA.	83
4.6	Reduction of energy consumption of MDs to conduct a handover operation by HGMA.	85
4.7	Improvement of QoS-satisfied handover ratios by HGMA.	86

Chapter 1

Introduction

1.1 Motivation

In the last decade, many mobile broadcasting technologies have been developed. Not only have a large number of broadcasting networks such as DVB-H [20], ATSC-M/H [7], DMB [19], and MediaFLO [56] been widely deployed, but many contemporary mobile devices such as smart phones are also powerful enough to receive the mobile broadcasting services such as mobile television (TV) and interactive social TV. Mobile broadcasting services are promising to become popular. Meanwhile, broadband Internet services (e.g., streaming, content delivery, and multicasting services) are wild reach for these mobile devices which have been equipped with other radio interfaces to access wireless broadband networks such as 3G cellular, WLAN, and WiMAX networks.

This dissertation aims at a *DVB-H (digital video broadcasting – handheld)* system, which is developed to support the broadcasting service of IP multimedia content such as digital TV programs to mobile devices [26, 55]. DVB-H is based on the successful DVB-T (terrestrial) system with some special designs for mobile devices. To deal with the quality-of-service requirements for DVB-H users, from the aspect of the mobile devices, we focus on both just-in-time data transmission and less power consumption requirements. So, the motive of this research work is to find out how to handle the erroneous reception on the

mobile devices in terms of DVB-H data losses. Specifically, with energy concerns to the mobile devices, how do we efficiently conduct the data recovery? We adopt the approach of benefiting by the support of the overlay wireless broadband networks to the mobile devices through which the data recovery can be conducted, as shown in Fig. 1.1. Thus, efficient transmissions for the data recovery will fit the need of just-in-time data transmission requirement. In addition, the user's mobility will be considered for consuming less energy on the mobile devices.

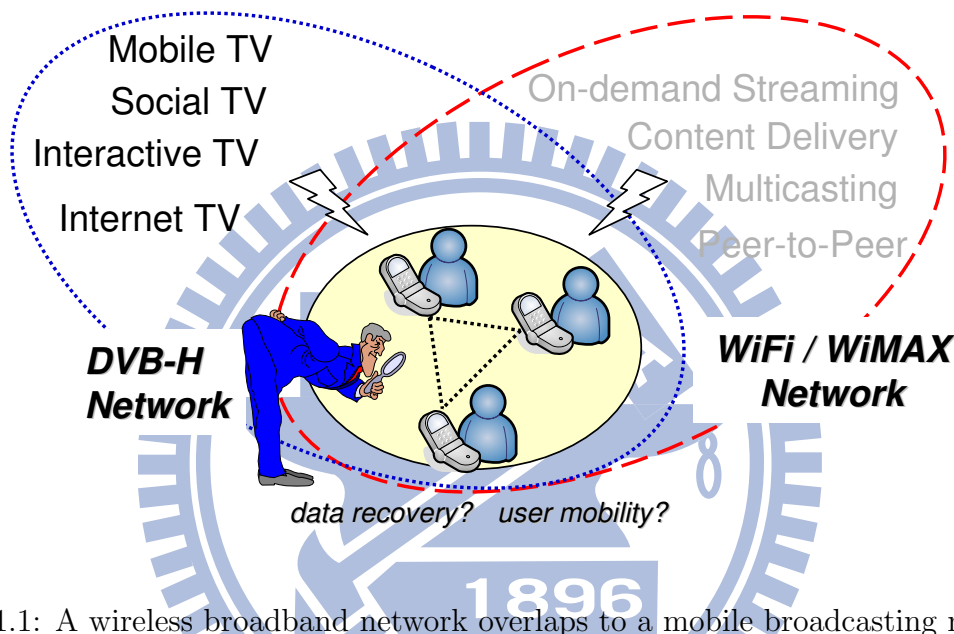


Figure 1.1: A wireless broadband network overlaps to a mobile broadcasting network.

To cope with data recovery for mobile broadcasting services by cooperating with wireless broadband networks, as shown as Fig. 1.2, we break down two genres of tasks in this dissertation. First, we conduct the data recovery through an IP-relay network in which includes two research works. The work 1 discusses the message-efficient data recovery for DVB-H data losses through an IP-relay network, while the work 2 considers the on-demand data recovery by network coding for mobile broadcasting systems. Second, we take the handover issue into consideration when the DVB-H mobile devices roam in multiple wireless networks. Thus, how to conduct network selection to the wireless broadband networks for the mobile devices is discussed, whereby the energy-efficient concern in an integrated WiMAX and WiFi network for the DVB-H mobile devices is done by the work 3 of this dissertation. In a nutshell, we pursue data recovery schemes to the mobile

devices for DVB-H data losses by partnering with wireless broadband networks, which are able to do efficient data retransmissions for error recovery via an IP-relay network and achieve less energy consumption on the mobile devices for handovering between WiMAX and WiFi networks.

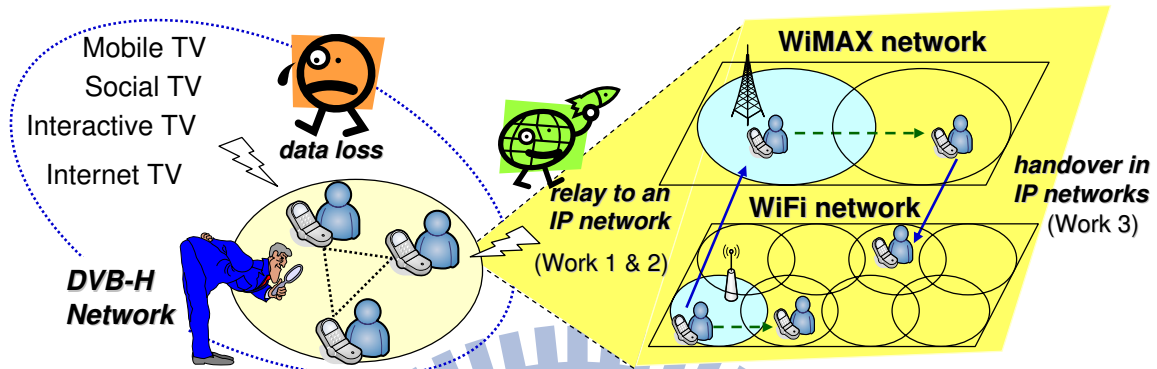


Figure 1.2: Structure of the research works in this dissertation.

1.2 Background of the Dissertation

1.2.1 Digital Video Broadcasting – Handheld (DVB-H)

DVB-H is developed from the existing DVB-T standard that is originally designed for the fixed and in-car reception of digital TV programs. In order to provide the same service at mobile devices, the DVB-H system employs a discontinuous transmission technique, called *time slicing*, to broadcast multimedia data to mobile devices. With the time slicing technique, multimedia data can be periodically transmitted in *bursts*, as shown in Fig. 1.3. Each burst also contains the information of the time interval between two bursts of the same service. In this way, mobile devices can synchronize to the bursts of their desired services (refer to the *burst duration*) and turn off their receivers when the bursts of other services are transmitting (refer to the *off-time* period). In spite of the discontinuous transmissions, mobile devices can still enjoy a constant data rate. The maximum allowable size of each burst is 2M bits. By the time slicing technique, the energy consumption of mobile devices can be significantly reduced, as compared with DVB-T systems. In

addition, the off-time periods can also provide smooth handover when a mobile device move from one service cell to another cell.

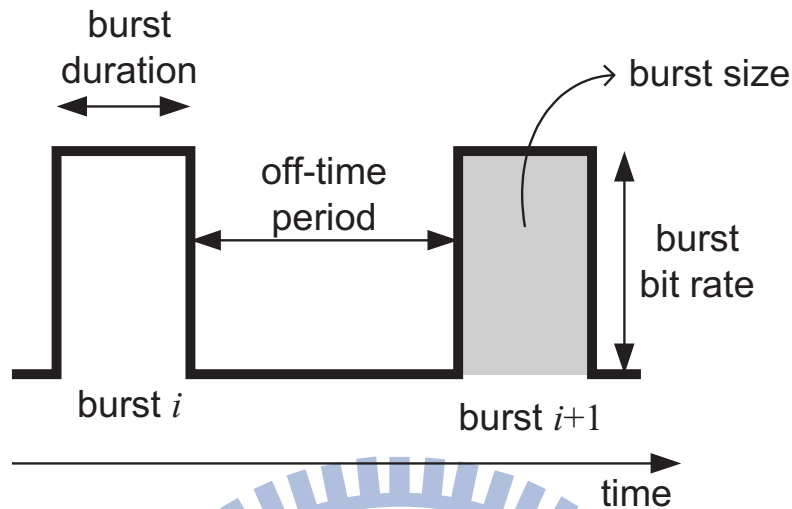


Figure 1.3: The time slicing technique used in a DVB-H system.

DVB-H is backward compatible to the original DVB-T system. In fact, both DVB-H and DVB-T data streams can be simultaneously transmitted through the same channel. Fig. 1.4 gives an example of the service multiplex in a common DVB-T/DVB-H channel. The total channel bandwidth is 13.27 Mbps and one quarter of the channel is used to transmit eight DVB-H data streams. The remaining channel bandwidth is then shared among three ordinary DVB-T data streams.

However, because bursts may be prone to error due to the nature of wireless broadcast, the DVB-H standard adopts some *forward error correction (FEC)* mechanisms to protect these bursts. On the other hand, the DVB-IPDC extension is also proposed to handle the situation when data retransmissions are necessary. Below, we introduce DVB-IPDC and error recovery in DVB-H systems.

1.2.2 IP Datacast over DVB-H (DVB-IPDC)

DVB-IPDC provides an end-to-end architecture between service applications and mobile devices. In a DVB-IPDC system, multimedia content is delivered as either a *streaming* service or a *file-casting* service. In the streaming service, a continuous data flow carrying

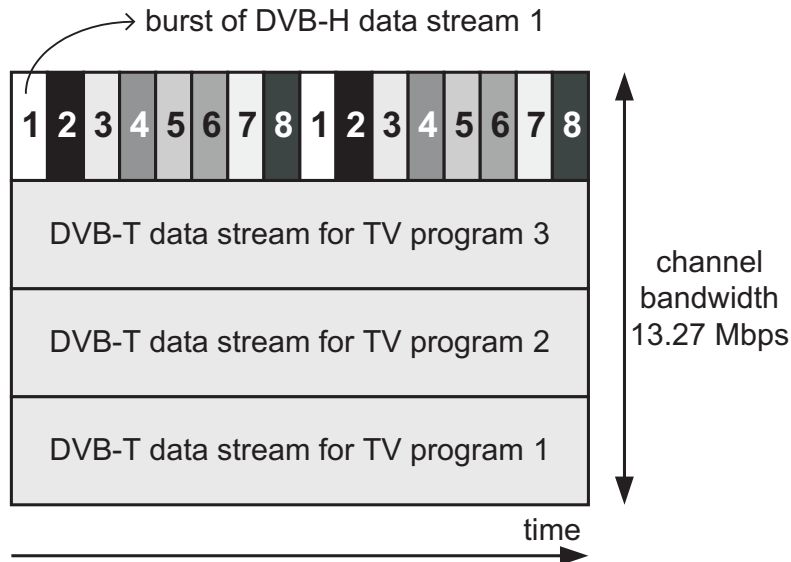


Figure 1.4: An example of the coexistence of DVB-T and DVB-H data streams in a common channel.

the audio, video, and subtitling content is delivered to the mobile device and will be directly consumed by the user. Occasional and slight data errors could be tolerated (even though they are not fixed or retransmitted). On the other hand, in the file-casting service, a finite amount of data are delivered and stored in the mobile device as a file. This file can be consumed either immediately or at a later time. However, the transmission of the file-casting service should be reliable, that is, any error in the data is not allowed.

In order to support both the streaming and the file-casting services, a DVB-IPDC protocol stack is developed, as illustrated in Fig. 1.5. The *user datagram protocol (UDP)* [70] is adopted as the transport protocol to provide the connectionless data transfer service. However, because UDP may not satisfy the requirements of the streaming and the file-casting services, DVB-IPDC requires other higher-layer protocols on the top of UDP. In particular, for the streaming service, the *real-time transport protocol (RTP)* [76] is adopted to guarantee the QoS (quality of service) requirements of the service. On the other hand, for the file-casting service, the *file delivery over unidirectional transport (FLUTE) protocol* [62] is adopted to provide flexible transmissions of data carousel sessions and single file transfers. In addition, DVB-IPDC adopts the *electronic service guide (ESG)* [23] to provide some detailed system information, such as the available services

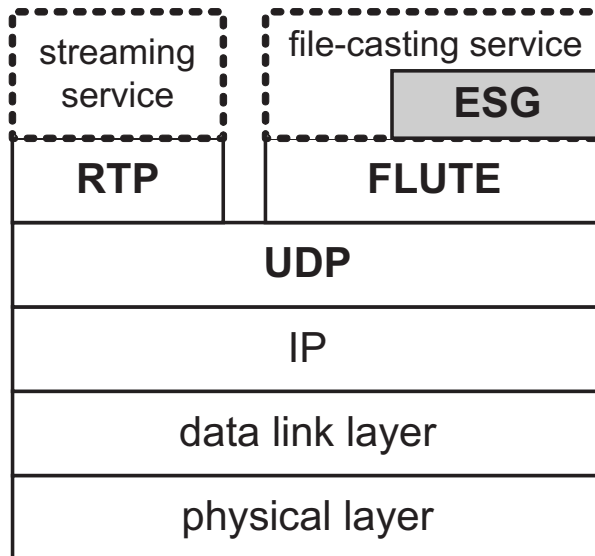


Figure 1.5: The DVB-IPDC protocol stack.

and the corresponding access information needed to use these services, for mobile devices. ESG can be considered as one type of the file-casting service and thus should be handled by the FLUTE protocol.

The system architecture of DVB-IPDC is illustrated in Fig. 1.6, which is composed of a number of *functional entities* and a set of *reference points*. These functional entities will cooperate together to provide some system capabilities:

- *Content creation*: The content creation entity provides the source of content for the streaming and the file-casting services.
- *Service application*: The service application entity offers a logical link between the content provider and the mobile device so that the service content can be delivered to the mobile device over multiple radio access technologies. In addition, the service application entity generates the meta-data for service description that will be used by ESG.
- *Service management*: The service management entity is used to control the services transmitted from the service application entity to the mobile device, which involves the following four functionalities:

- 1) Register the services to the system and allocate network bandwidth for these services.
 - 2) Collect and aggregate the meta-data of service description (from the service application entity) into ESG and then transmit ESG to the mobile device.
 - 3) Provide some security mechanisms to protect the services.
 - 4) Support some location services.
- *Broadcast network:* The broadcast network provides a unidirectional channel to deliver service content to the mobile device through point-to-multipoint communications.
 - *Interactive network:* The interactive network provides a bidirectional channel to allow the mobile device to interact with (or feed back to) the service management and the service application entities.

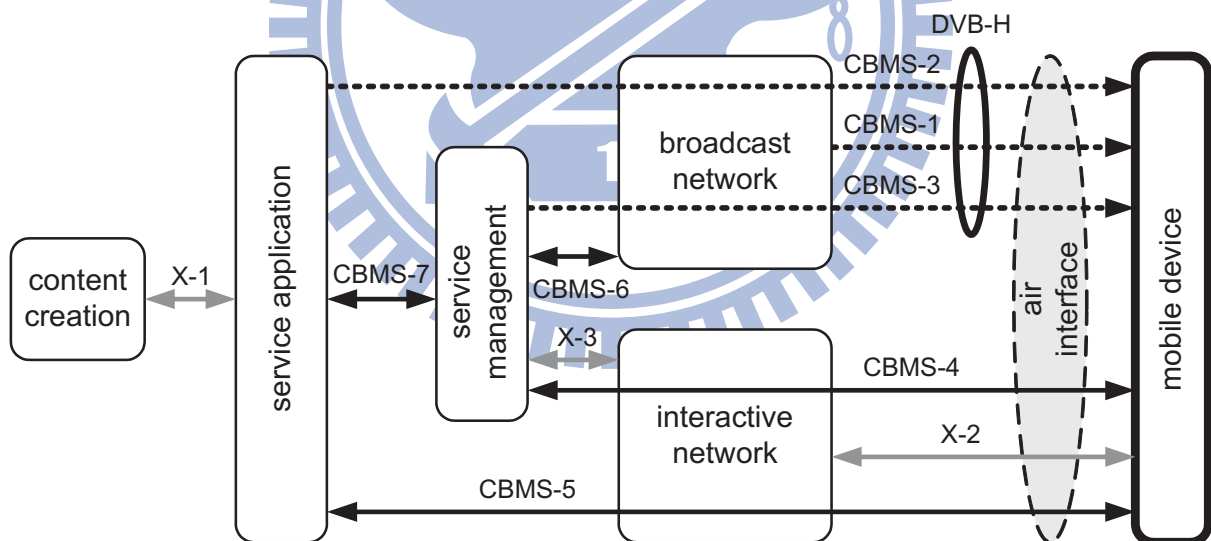


Figure 1.6: The system architecture of DVB-IPDC.

Two functional entities can communicate with each other (in one direction or two directions) through reference points, as shown in Fig. 1.6. These reference points include CBMS- i (convergence of broadcast and mobile services) and X- j ('X' stands for "extended"), $i = 1..7$ and $j = 1..3$, in which some specific data will be carried. For example,

reference point CBMS-1 delivers the program specific information and the service information to the mobile device through the broadcast network. Reference point CBMS-2 delivers audio and video streams and the necessary files from the service application entity to the mobile device. Reference point CBMS-3 transmits the ESG meta-data from the service management entity to the mobile device in a point-to-multipoint communication manner (that is, broadcast). Reference point CBMS-4 exchanges some data such as the access control information and the ESG meta-data between the service management entity and the mobile device in a point-to-point communication manner (that is, unicast). Finally, reference point CBMS-5 exchanges some point-to-point transport services such as SMS (short message service), MMS (multimedia message service), and IP connectivity between the service application entity and the mobile device.

From Fig. 1.6, it can be observed that the DVB-IPDC extension adopts an interactive network to support bidirectional communication between the system and mobile devices. In this way, when packets are lost in the broadcast network, mobile devices can ask the interactive network for data retransmissions. Many research efforts follow the similar architecture of DVB-IPDC that adopts another IP-based wireless network to handle the data recovery problem in a DVB-H system. In short, from the perspective of the layered structure, as illustrated as Fig. 1.7, the DVB-IPDC is designed for supporting applications over DVB-H on the basis of IP, while DVB-H provides the transmission protocol to support unidirectional and best effort IP packet delivery.

1.2.3 Error Recovery in DVB-H Systems

Since mobile devices are more vulnerable to packet loss in a wireless environment, it is necessary to provide some data recovery mechanisms [67]. We summarize three genres of mechanisms related to error recovery in DVB-H systems as follows.

- 1) To protect DVB-H packets against multipath fading or interference, some *error correction codes* can be appended in these packets. In this way, when few bits of a DVB-H packet are corrupted during transmission, the mobile device could fix

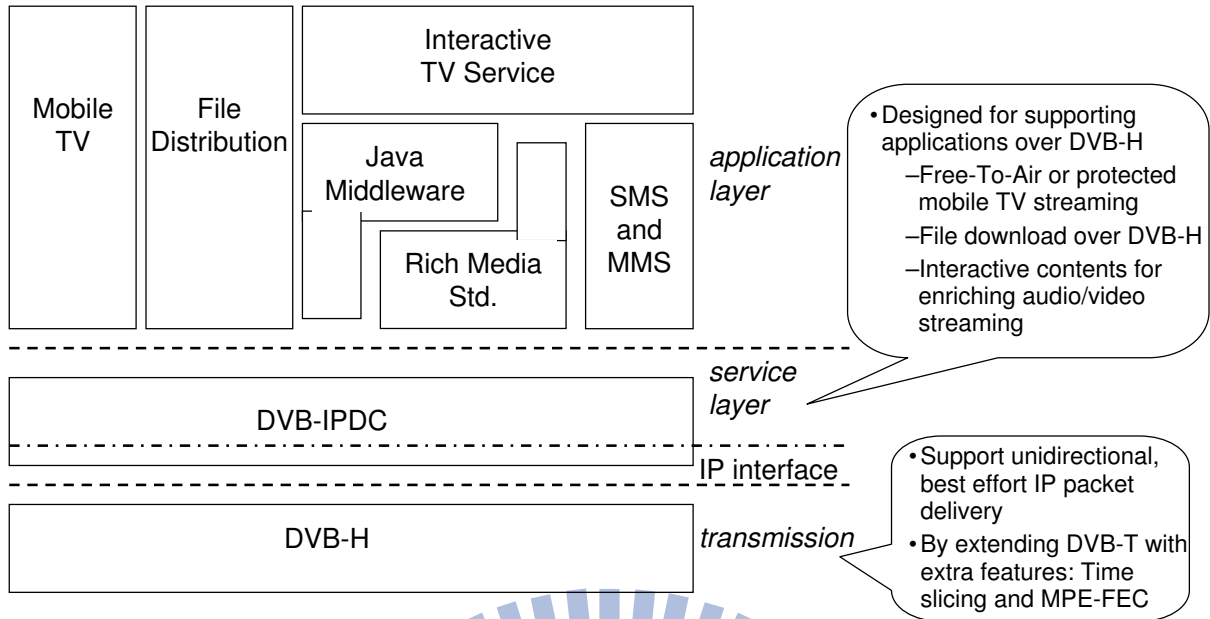


Figure 1.7: The perspective of layered structure for DVB-IPDC and DVB-H.

these bits by the corresponding error correction code. However, when the DVB-H packets are seriously corrupted, it is necessary to retransmit these packets. Besides, appending the error correction codes will increase the communication cost because the packet lengths are extended.

- 2) DVB-H provides a narrow *return channel* (e.g., via telecommunication network) for mobile devices to feed back some short control messages. Mobile devices may also use this return channel to notify the DVB-H server of packet loss. Then, the server can retransmit these DVB-H packets through the broadcasting channel. However, the return channel is quite narrow so that mobile devices may not be allowed to transmit a large number of requests. In addition, the DVB-H server could be overloaded because the server has to deal with both the regular broadcasting job and the retransmissions of lost packets [83].
- 3) The *DVB-IPDC (IP datacast over DVB-H)* standard [22, 45] is developed to support the interaction mechanism of a DVB-H system. DVB-IPDC relies on a separate IP-based wireless network to provide bidirectional communications between mobile devices and the DVB-IPDC server. Such a network can also support the delivery

of requests from mobile devices and the retransmissions of lost packets to mobile devices. In particular, there are two coexisting networks: one is the DVB-H broadcast network and the other is a recovery network. Mobile devices will continually receive the broadcasting content from the DVB-H network. However, when packets are lost in the DVB-H network, the mobile devices can request the recovery network to retransmit these DVB-H packets.

In order to increase the robustness of data transmission in a DVB-H system, two optional FEC mechanisms are proposed in the data link layer and the application layer. These two FEC mechanisms are adopted to recover erroneous data at mobile devices (without the need of data retransmission) but they cannot cooperate together. In particular, when the *data link layer FEC (LL-FEC)* is adopted, the *application layer FEC (AL-FEC)* should be disabled, and vice versa.

In the LL-FEC mechanism, bursts are protected by the *Reed-Solomon (RS) codes* [73], which are non-binary cyclic error-correcting codes used to describe a systematic manner to detect and correct multiple random symbol errors. Specifically, each burst consists of multiple *MPE (multi-protocol encapsulation) sections*, where an MPE section encapsulates one IP packet with a 12-byte header and a 4-byte CRC-32 (cyclic redundancy check) tail. In addition, the RS parity data are also encapsulated into MPE-FEC sections and formed as a part of one *MPE-FEC frame*. An MPE-FEC frame is composed of one *application data table* and one *RS data table*, as shown in Fig. 1.8. The application data table encapsulates the IP data packets (possibly with padding) while the RS data table stores the corresponding parity data. The number of rows in an MPE-FEC frame can be 256, 512, 768, or 1024, which is indicated in the service information. On the other hand, the number of columns in an MPE-FEC frame is 255, where the application data table and the RS data table occupy 191 and 64 columns, respectively. The maximum size of an MPE-FEC frame is 2M bits to fit into one burst. Starting from the upper left corner, the IP data packets (of the same burst) are inserted column by column in the application data table. When the IP data packets of a burst cannot completely fill with the application

data table, the remaining table space is padded with zero bits. On the other hand, by using the Reed-Solomon RS(255,191) code, the 64-byte parity data in each row of the RS data table can be calculated from the 191-byte IP data in each row of the application data table. Because writing and reading the IP data in the application data table is conducted in column-wise direction while the parity data are calculated in row-wise direction, the MPE-FEC frame structure can result in a block interleaving effect over the whole frame data and thus provide a certain degree of robustness.

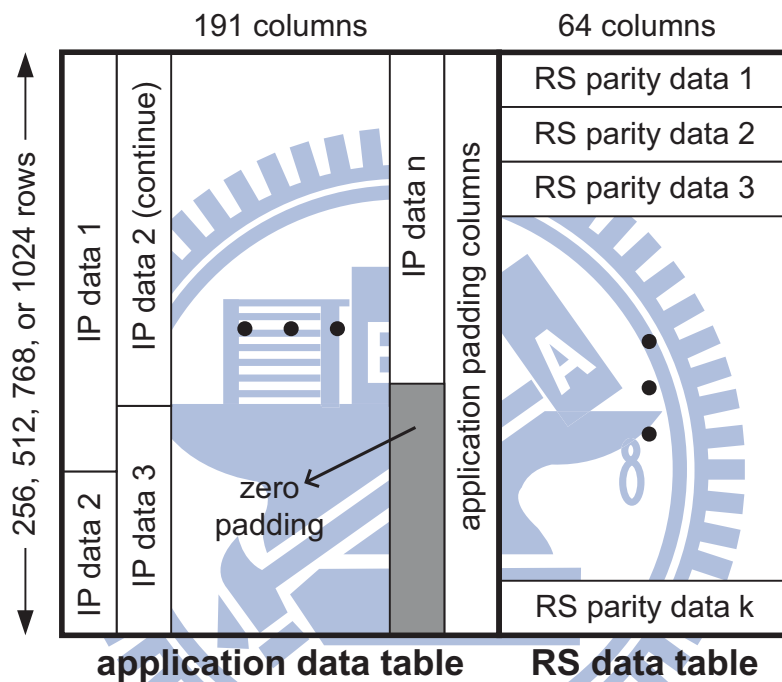


Figure 1.8: The MPE-FEC frame used in the LL-FEC mechanism.

The AL-FEC mechanism is based on the digital fountain coding scheme [10], which can generate an infinite amount of parity data on the fly (that is, *rateless*). Theoretically, with an ideal fountain code, the source file can be reliably constructed after receiving a certain amount of encoded data. The digital fountain codes allow efficient asynchronous file downloading over broadcasting channels without any feedback mechanism, and they can also be used in data delivery in wireless broadcast systems. The Raptor coding scheme [78] is one practical implementation of the digital fountain codes and its systematic version is also standardized in DVB-H.

AL-FEC is more efficient than LL-FEC when transmitting large-sized files that are distributed over more than one burst[61]. The reason is as follows. With the LL-FEC mechanism, each burst must be decoded successfully in order to recover the whole file. When the mobile device loses one burst, it has to wait until the mobile device can receive that burst again (via retransmission). Meanwhile, the bursts containing data that have already been received will be discarded by the mobile device. On the contrary, in the AL-FEC mechanism, all source data and parity data received correctly are useful to the mobile device, which can accelerate the delivery of the file. On the other hand, when transmitting small-sized files that can fit into one burst, the system performance of the LL-FEC mechanism will be almost the same as that of the AL-FEC mechanism.

Many research efforts follow the similar architecture of DVB-IPDC to realize data recovery in a DVB-H system by adopting an IP-based wireless network to serve as the recovery network. The following three types of data recovery schemes for the DVB-H system are discussed.

- 1) In the first type of data recovery schemes, a cellular network such as 3G or UMTS (universal mobile telecommunications system) is adopted to support bidirectional communications between mobile devices and the DVB-IPDC server. However, because the bandwidth of a cellular network is usually narrow, the parity data or the lost DVB-H packets may be (re)transmitted through either the DVB-H network or the cellular network. With the uplink channel of the cellular network, mobile devices can feed back the current network situation to the DVB-IPDC server for deciding which network (DVB-H or cellular) should be adopted to conduct data recovery so that the system performance can be improved.
- 2) In the second type of data recovery schemes, the DVB-H system is integrated with a wireless broadband network. Since the wireless broadband network such as WLAN or WiMAX possesses a wider bandwidth compared with a cellular network, both the request submission from mobile devices and the retransmission of lost DVB-H packets can be conducted in the wireless broadband network. In other words, the

DVB-H network is only responsible for broadcasting multimedia content.

- 3) Unlike the first two types of data recovery schemes that adopt an infrastructure network (for example, cellular or WiMAX networks) for data recovery purpose, in the third type of data recovery schemes, mobile devices will organize an ad hoc network to exchange DVB-H data. In particular, when a mobile device loses some DVB-H packets, this mobile device will query its neighboring devices for sharing these packets. Then, packet retransmissions of lost DVB-H data can be realized in a peer-to-peer communication manner.

1.2.4 Vertical Handover in Wireless Networks

When a mobile device roams through different wireless networks, e.g., WiMAX and WLAN, how to maintain a consistent connection and the seamless handover in these wireless networks is important because they are varied in terms of frequency band, bandwidth, data transmission latency, coverage range and other features. Traditionally, handoff process is performed between two base stations within same technology. This kind of handoff is defined as horizontal handoff. For example, in Fig. 1.9 a mobile device move around in the WiFi network or the WiMAX network. In contrast to horizontal handoff, vertical handoff is defined as the handoff between base stations with different wireless technologies, which is likely as the mobile device mobile between the WiMAX and WiFi networks in Fig. 1.9. The fast and seamless vertical handover has three main requirements. First, the vertical handoff process should be done automatically, without user's intervention. Second, the mobile device with multiple interfaces for different wireless networks can automatically select the most appropriate wireless network. Third, the dataflow should be smoothly redirected to the new wireless network when the mobile device is roaming between different service providers.

The vertical handoff process is generally divided into three steps, in which many technical challenges exist. First, *system discovery*: the mobile device must know which wireless networks are reachable. Second, *handover decision*: the mobile device then evaluates the

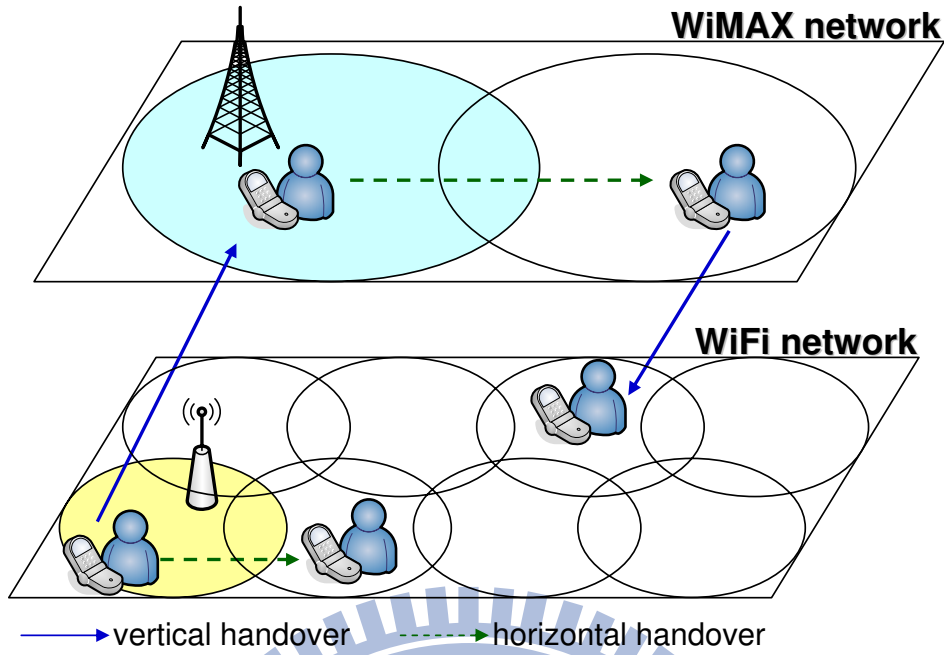


Figure 1.9: Handover in WiMAX and WiFi Networks.

reachable wireless networks to make the handover decision. Third, *handover execution*: if the mobile device decides to perform vertical handover, it executes the vertical handover procedure to be associated with the new wireless network. Since a mobile device with multiple interfaces must activate the interface to know a wireless network is reachable if it wants to associate with, the simplest way to do system discovery is always keeping all the interfaces on. However, activating the interface consumes power even without receiving/sending any packet. Moreover, the mobile should observe whether performing handover into the candidate network is worthwhile, that is finding the better wireless network and performing handoff into it. Besides, the frequency of triggering vertical handover which arouse switching wireless interfaces should be considered when maintaining the connection smoothly.

1.3 Contributions of the Dissertation

In this dissertation, we study the *data recovery* issue in a mobile broadcast network, which considers conducting the data recovery for DVB-H data losses through a recovery network

and the recovery network selection for mobile devices. For the issue of conducting the data recovery for DVB-H data losses through a recovery network, we first take an IP-relay network as the recovery network, in which discusses the message-efficient data recovery for DVB-H data losses through the IP-relay network, and then discuss how to apply network coding to the on-demand data recovery for mobile broadcasting systems. For the mobile devices roaming in the recovery network which composes of different wireless networks, we discuss how to conduct energy-efficient network selection to an integrated WiMAX and WiFi network for the mobile devices. The contributions of this dissertation are detailed as follows:

- For the message-efficient data recovery for DVB-H data losses through an IP-relay network, major contributions are three-fold. First, we point out the new GPL and BDH problems under the DVB-IPDC environment. Second, we propose the lazy wait and group acknowledgement schemes to solve these problems by exploiting the spatial and temporal correlation of requests. Simulation results show that our schemes can significantly reduce the amounts of retransmission requests and retransmission packets, thereby alleviating network congestion. Third, with mathematical analysis, we discuss how to adaptively adjust the timers of lazy wait and group acknowledgement according to the channel conditions.
- For the on-demand data recovery by network coding for mobile broadcasting systems, WiMAX serves as a recovery channel for mobile devices to request for retransmissions, which is modeled by repetitive frames, each with a *submission period* followed by a *recovery period*. Mobile devices submit their requests in the submission periods while the WiMAX network sends the lost packets in the recovery periods. To recover more urgent packets first, we develop two prioritized network coding schemes based on XOR coding. The first *recovery by greedy selection (RGS)* scheme constructs a weighted bipartite graph to reflect the influence of each coded packet and then adopts a greedy strategy to find a minimum dominating set of coded packets. By extending the similar concept to cope with the relation between

requests submissions and packets losses as well as the retransmission sequence of coded packets, the second *recovery by prioritized selection (RPS)* scheme adopts a tree traversal approach to adjust weights and find a minimal set of coded packets from the graph. Simulation results verify that both RGS and RPS can improve recovery efficiency while reduce the number of packets being dropped due to missing deadlines.

- For the energy-efficient network selection in an integrated WiMAX and WiFi network, the major contributions are three-fold. Firstly, we identify the GM feature of MDs in an integrated WiMAX and WiFi network. Secondly, we propose an energy-efficient HGMA handover scheme by adopting the GM feature. Our HGMA scheme can eliminate unnecessary handovers, reduce the number of network scanning, and avoid switching network interfaces too frequently. Extensive simulation results show that HGMA can conserve about 59% to 80% energy consumption to conduct a handover operation, save about 70% of network scanning, and reduce about 31% of interface switching. Thirdly, HGMA prefers the low-tier WiFi network over the WiMAX network, and guarantees the bandwidth requirements of handovering MDs. It is verified by simulations that HGMA can make MDs to associate with WiFi networks with 16% to 62% more probabilities, and increase 20% to 61% of possibility that WiFi APs can afford the demanded bandwidths of MDs after they handover to.

1.4 Organization of the Dissertation

The rest of this dissertation is organized as follows. In Chapter 2, we solve the group packet loss and broadcast data handover problems in conducting the data recovery for DVB-H data losses through an IP-relay network. In Chapter 3, we discuss how to provide on-demand data recovery for mobile broadcasting systems by applying network coding methods. In Chapter 4, we discuss how to do network selection in an integrated WiMAX and WiFi network with energy-efficient awareness. In Chapter 5, we give the conclusions

and future directions of this dissertation.



Chapter 2

Data Recovery through an IP-Relay Network

2.1 Introduction

DVB-H (digital video broadcasting–handheld) [26, 55] is developed from the successful *DVB-T* (terrestrial) system with special designs for handheld, battery-powered devices. *DVB-H* adopts a time-slicing mechanism to reduce the energy consumption of *mobile devices* (MDs). Since MDs are vulnerable to packet loss in a wireless environment [21, 67], a feedback mechanism to request retransmissions is necessary. There are two possible solutions. One is to use the *DVB-H* return channel to request the lost packets. However, not only the return channel is quite small, but also the *DVB-H* server can be easily overloaded [83]. The other solution is to adopt *DVB-IPDC* (*IP datacast over DVB-H*) [22, 45, 37], which relies on cooperating with a separate wireless network, such as an IP-relay wireless network, to allow MDs to submit requests and receive lost packets.

This work supports *DVB-IPDC* by adopting broadband WiMAX networks [52, 91] as an example to serve as the datacast channel. We consider MDs roaming inside the coverage of both *DVB-H* and WiMAX networks, as shown in Fig. 2.1. The *DVB-H* server

continuously broadcasts digital videos to MDs. When a MD detects any packet¹ loss, it requests a nearby WiMAX *relay station (RS)* for retransmission. Under this architecture, we point out two critical problems: *group packet loss (GPL)* and *broadcast data handover (BDH)*. GPL occurs when there are bursty requests for retransmissions of the same pieces of data with high spatial or temporal correlation. Thus, some RSs may be seriously congested by incoming requests and outgoing retransmissions. On the other hand, BDH occurs when some MDs handover to neighboring RS cells after submitting their requests. Thus, duplicate requests and retransmissions could further congest the WiMAX network.

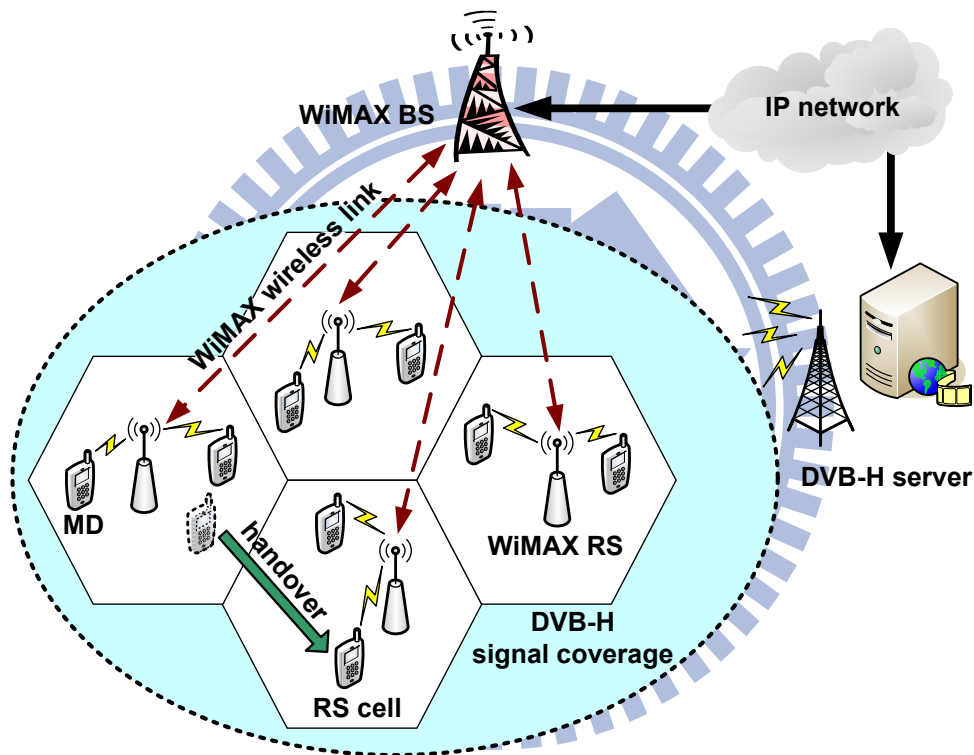


Figure 2.1: The DVB-IPDC architecture with WiMAX networks as the datacast channel.

To solve these problems, we should prevent MDs from sending duplicate requests and deliver lost packets effectively. In fact, many requests could exhibit *spatial* or *temporal* correlation. In the space domain, neighboring MDs may send similar requests because they are interfered by the same noise or obstacles. Moreover, the handover behavior of MDs may extend this spatial correlation to neighboring cells. In the time domain, some MDs may lose a sequence of packets since the interference sources often exist for a spell.

¹In this chapter, a “packet” means a DVB-H packet.

If redundant requests and retransmissions can be merged, the network efficiency can be greatly improved.

This work proposes a *bulk recovery with lazy wait (BR-LW)* scheme to deal with the GPL and BDH problems by exploiting the spatial and temporal correlation of retransmission requests. The idea is to adopt the *lazy wait* and *group acknowledgement* mechanisms. Specifically, when a MD detects any packet loss, it “lazily” waits a random period and only sends the request to its associated RS if needed. Then, each RS accumulates the requests from its MDs and broadcasts a group acknowledgement g_{ack} to its cell to announce the requests it has received. On receiving g_{ack} , each MD cancels those requests that appear in g_{ack} but have not been sent out yet. Also, the RS packs its accumulated requests in bulk and sends it to the WiMAX *base station (BS)*. The BS queries the DVB-H server for the lost packets and then sends these packets to MDs through the corresponding RSs.

In BR-LW, we adopt three special designs to reduce duplicate requests and improve network efficiency:

- With lazy wait, each MD can pack many small requests together and sends one bulk request to the RS. Thus, not only is the amount of message transmissions reduced but network contention is also alleviated. The waiting period should be adaptively adjusted based on the DVB-H channel condition. When the MD experiences a bad channel, a shorter period is adopted to quickly recover the lost packets; otherwise, a longer period is adopted to accumulate more requests.
- The group acknowledgement g_{ack} confirms not only the receipt of requests by the RS but also prevents MDs from sending duplicate requests. The timing to send g_{ack} should consider the spatial and temporal correlation of requests. When most requests exhibit high spatial or temporal correlation, g_{ack} is sent immediately to avoid further duplication. Otherwise, the RS can defer g_{ack} to collect more requests.
- The BS also takes advantage of the spatial and temporal correlation of retransmission requests to improve the transmission efficiency. In the space domain, we divide RSs into several multicast groups according to their positions, so that the BS can

multicast the requested packets to adjacent RSs altogether to reduce the amount of transmissions. In the time domain, the BS proactively fetches correlated packets from the DVB-H server in advance to reduce the downloading latency.

Major contributions of this work are three-fold. First, we point out the new GPL and BDH problems under the DVB-IPDC environment. Second, we propose the lazy wait and group acknowledgement schemes to solve these problems by exploiting the spatial and temporal correlation of requests. Simulation results show that our schemes can significantly reduce the amounts of retransmission requests and retransmission packets, thereby alleviating network congestion. Third, with mathematical analysis, we discuss how to adaptively adjust the timers of lazy wait and g_{ack} according to the channel conditions.

The rest of this chapter is organized as follows. Section 2.2 gives the problem statements and surveys related work. Section 2.3 presents the BR-LW scheme while Section 2.4 discusses how to determine its system parameters. Simulation results are shown in Section 2.5. Conclusions are drawn in Section 2.6.

2.2 Preliminaries

2.2.1 Problem Statements

We consider a DVB-H system with WiMAX networks as the datacast channel (refer to Fig. 2.1). Each MD is equipped with a DVB-H data receiver and a WiMAX wireless interface. The WiMAX networks consist of multiple BSs connected to the DVB-H server through an IP-based network. Each BS supports multiple RSs. MDs continuously receive the DVB-H broadcast data and ask their associated RSs to retransmit the lost packets. Each packet has a deadline and a packet missing its deadline will be dropped. We assume that each MD maintains a small playback buffer and there is an upper-layer protocol to manage the buffer to guarantee some video QoS (quality of service) requirements, such as jitter and PSNR (peak signal-to-noise ratio). Therefore, in this work we only focus on

the retransmission of lost packets to avoid missing their deadlines.

For each BS, it collects the requests from its RSs, packs and forwards these requests to the DVB-H server, and then sends the requested packets to MDs through the corresponding RSs. In the above architecture, the GPL problem occurs when MDs send duplicate requests for the same packets (that is, the requests exhibit spatial correlation) or multiple requests for consecutive packets (that is, the requests exhibit temporal correlation). The BDH problem occurs when a MD sends its requests to a RS, handovers to another RS, and then resends the same requests to this new associated RS again. Our goal is to reduce the above redundancy to improve retransmission efficiency and avoid potential congestion in the WiMAX networks.

2.2.2 Related Work

DVB-H is developed from the DVB-T system, which relies on uni-directional broadcast. Prior studies thus focus on integrating DVB-T with other wireless networks to provide two-way communications. For example, [82] suggests adopting GSM networks for this purpose. In [72], a GPRS network is adopted to support near-media-on-demand service such as MP3 delivery. References [47, 46] address the integration of 3G, WLAN, and DVB-T networks, where the objective is to manage the composite network resources. A seamless, cross-layer interworking scheme between DVB-T and IEEE 802.11 WLANs is proposed in [18] to provide interactive mobile TV services, where a gateway is deployed in each access network to conduct media adaptations. Similarly, [74] proposes a testbed for mobile interactive television, where DVB-H streams are accessed through WiFi networks. In [75], the concept of mobile social television is proposed, where the DVB-H content is shared among mobile users through peer-to-peer interactivity. Automatic configuration of an IP-based interworking model among WLAN, 3G, and DVB-H is addressed in [63].

Following the DVB-IPDC architecture, several research efforts consider recovering DVB-H data through an IP-relay wireless network. Reference [6] suggests multicasting a DVB-H stream to multiple receivers through a WiFi access point. This work relies on

maintaining multicast group membership to avoid CSMA/CA contention; it differs from our work in that we use the datacast channel only for retransmissions. The study of [31] adopts a cellular network to transmit repair data for the receivers temporarily affected by noise, interference, or fading. Similarly, [32] integrates DVB-H with a 3G network to transmit additional parity data for the recovery purpose. Reference [33] proposes an application-layer FEC (forward error correction) scheme and a multi-burst coding scheme to deliver DVB-H data to improve transmission robustness. However, none of the above work addresses the GPL and BDH problems. References [24, 25] propose a *content delivery protocol (CDP)* for the DVB-IPDC architecture to recover lost or corrupted packets, and [35] evaluates the cost of CDP’s repair mechanism. Note that CDP relies on a point-to-point mechanism, while ours adopts a more efficient point-to-multipoint mechanism. In [79], MDs are organized as an ad hoc network and each MD encountering packet loss will query the lost packets from other MDs. [38] proposes a recovery mechanism based on a peer-to-peer basis connecting Mobile TV clients via UMTS to recover packets that have been lost during the Mobile TV DVB-H transmission. Different from [79, 38], our work follows the DVB-IPDC architecture to retransmit through an infrastructured network.

2.3 Framework of the BR-LW Scheme

In this section, we describe our BR-LW scheme. How to adaptively adjust its system parameters will be discussed in Section 2.4. Table 2.1 summarizes our notations. Fig. 2.2 illustrates the flowchart of BR-LW. We assume that each MD has a small buffer to store the received DVB-H packets before playback, so a MD can tolerate a reasonable delay d_i for each packet p_i . Our goal is to reduce the amount of message transmissions between MDs and RSs and that between RSs and BSs to preserve wireless bandwidth and avoid network congestion. Below, we present the operations conducted by MDs, RSs, and BSs.

Table 2.1: Notations used in Chapter 2.

notation	definition
T_{MD}	the table stored in each MD to record the information of the lost packets
p_i	a lost packet with index i
d_i	the deadline of packet p_i
t_i^L	the lazy-wait timer of packet p_i
t_{lazy}	the threshold value to determine the lazy-wait timers
T_{RS}	the table stored in each RS to record the information of the received RREQs
n_i	the number of RREQs that request packet p_i
g_{ack}	the group acknowledgement
c_{num}	the number of entries in T_{RS}
c_{var}	the coefficient of variation of all n_i s in T_{RS}
δ_{num}	the threshold value of c_{num} for the RS to trigger a g_{ack}
δ_{var}	the threshold value of c_{var} for the RS to trigger a g_{ack}
ζ	the threshold value of the sum of all n_i s in T_{RS} for the RS to trigger a g_{ack}
μ_E	the expected number of successive lost packets by a MD
\mathcal{R}	the set of lost packets requested by RREQs during the $t_{g_{\text{ack}}}$ period
\mathcal{N}_R	the size of \mathcal{R}
ε_i	the minimum value of n_i for the RS to trigger a g_{ack}
\mathcal{T}	the period of simulation time
\mathcal{F}	the distribution of packet loss in the simulations
\mathcal{L}	the distribution of correlation feature of RREQs in the simulations

2.3.1 Operations of MDs

Each MD continuously receives the packets sent from the DVB-H server, and maintains a table T_{MD} to record the information of the lost packets. Specifically, for each lost packet p_i , the MD inserts an entry (p_i, d_i, t_i^L) into its T_{MD} , where d_i is the deadline and t_i^L is the *lazy-wait timer* of packet p_i . Initially, the entry (p_i, d_i, t_i^L) is set as “*unqueried*”. The

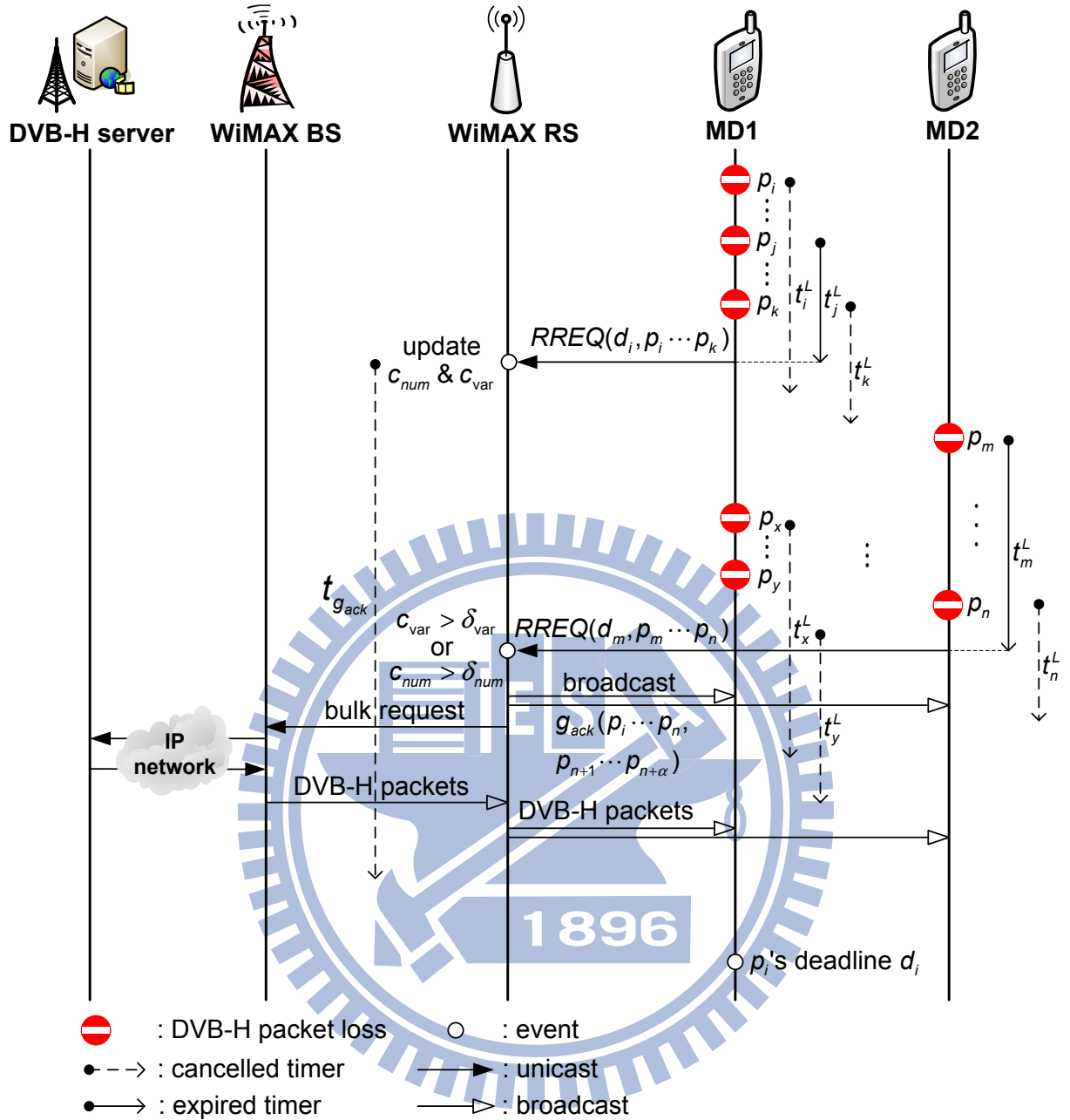


Figure 2.2: The flowchart of BR-LW.

value of timer t_i^L is randomly selected from $(0, t_{lazy}]$, where t_{lazy} is a system parameter to be selected according to the current DVB-H channel condition. Intuitively, when the MD experiences a bad channel, a smaller t_{lazy} is set to enforce the MD to quickly react to packet loss; otherwise, a larger t_{lazy} is set to allow the MD to accumulate more requests. (We will discuss how to adaptively adjust t_{lazy} in Section 2.4.1.) Note that the order that these lazy-wait timers expired is not necessarily the same as the order that the

corresponding packets were lost. Fig. 2.2 gives an example. Although packet p_i is lost earlier than packet p_j , timer t_i^L expires later than timer t_j^L .

When any lazy-wait timer expires, the MD will send to its associated RS a *recovery request* $RREQ(d_i, p_i, \dots, p_k)$ containing the earliest packet deadline d_i and the indices of all lost packets p_i, \dots, p_k whose entries are “unqueried” in its T_{MD} . We assume that the transmissions of RREQs are more *reliable* (guaranteed by an acknowledgement mechanism). Then, the MD marks all entries in its T_{MD} as “queried” and cancels all active lazy-wait timers. For example, in Fig. 2.2, since timer t_j^L expires first, MD₁ will send a RREQ and cancel timers t_i^L and t_k^L .

Each MD continuously repeats the above actions (that is, inserts entries for lost packets into T_{MD} and marks entries as “queried” by sending RREQs). However, when the MD receives a *group acknowledgement* g_{ack} from its associated RS while its T_{MD} is non-empty, it will mark those entries in its T_{MD} that have the same packet indices in g_{ack} as “queried” and then cancel the corresponding lazy-wait timers. Fig. 2.2 shows an example. Supposing that MD₁ receives g_{ack} that contains the packet indices p_x and p_y , MD₁ will mark the entries (p_x, d_x, t_x^L) and (p_y, d_y, t_y^L) as “queried” and cancel the active timers t_x^L and t_y^L . When either deadline d_i expires or the MD receives packet p_i , the entry (p_i, d_i, t_i^L) is removed from T_{MD} .

To deal with the BDH problem, when a MD handovers to a new RS cell, two cases will be considered:

1. If all entries of the MD’s T_{MD} are “unqueried”, which means that there are no *pending* RREQs in its old RS, this handovering MD behaves the same as a non-handovering MD.
2. If there are some entries marked as “queried” in T_{MD} , which means that the handovering MD did not receive the requested packets from the *old* RS,² the MD will set all entries in its T_{MD} as “unqueried” and initialize a lazy-wait timer t_{HO}^L . The value of t_{HO}^L is also randomly selected from $(0, t_{lazy}]$. This lazy-wait timer is not

²That is, the MD sends RREQs to the RS but the RS has not returned the requested packets yet.

associated with any packet. Its purpose is to ensure that this handovering MD still has a chance to send a RREQ to let the new RS know its (previously) lost packets if the MD does not experience any packet loss after handovering to the new RS cell.

When any of the lazy-wait timers expires, this handovering MD will send a RREQ to the new RS and cancel all active lazy-wait timers (including t_{HO}^L).

The above lazy wait design has three advantages. First, we can prevent MDs from sending a large number of *small* RREQs. Instead, a MD can combine all requests together and send a bulk RREQ. Thus, not only the overhead of packet headers is reduced but the network contention is also alleviated. Second, the *random* waiting periods help differentiate RREQs' transmission time, and thus reduce potential packet collisions. Third, by deferring RREQs, the probability to receive g_{ack} before sending RREQs is increased, thus avoiding MDs from sending duplicate requests to the RS. Since a handovering MD will also conduct lazy wait in the new RS cell, some of its duplicate requests could be eliminated if packet loss in both new and old RS cells exhibit spatial correlation.

2.3.2 Operations of RSs

Each RS should keep the RREQs received from its associated MDs in a table T_{RS} . Each entry (p_i, n_i) in T_{RS} contains the index of a lost packet p_i and the number n_i of MDs that request packet p_i . Every time when T_{RS} changes from empty to non-empty, a timer $t_{g_{ack}}$ is initiated to determine the deadline to broadcast g_{ack} . The expiration of $t_{g_{ack}}$ should be earlier than the earliest packet deadline in each received RREQ. For example, in Fig. 2.2, $t_{g_{ack}}$ should expire earlier than the earliest packet deadline d_i .

In addition, the RS should check two parameters c_{num} and c_{var} whenever the content of its T_{RS} changes, where c_{num} is the total number of entries in T_{RS} and c_{var} is the *coefficient of variation* of all n_i s in T_{RS} which is defined as the ratio of the standard deviation of all n_i s to their mean. The RS will broadcast a g_{ack} immediately to its associated MDs if any of the following conditions is true:

1. $c_{num} \geq \delta_{num}$: Since the RS will clear c_{num} and T_{RS} whenever it sends out g_{ack} (this

will be discussed later), the functionality of T_{RS} can be viewed as a “sliding window” to record which packets are lost among a *small* set of packets broadcasted from the DVB-H server in each round. Therefore, when the number of entries in T_{RS} (that is, c_{num}) exceeds a threshold δ_{num} , there is a high probability that consecutive or semi-consecutive packets are lost. In other words, these RREQs could exhibit temporal correlation.

2. $c_{\text{var}} \geq \delta_{\text{var}}$: When the coefficient of variation c_{var} exceeds a threshold δ_{var} , it means that *some* of the packets may be lost by a large number of MDs. In this case, these RREQs could exhibit spatial correlation.
3. $\sum_{(p_i, n_i) \in T_{\text{RS}}} n_i \geq \zeta$: When both $c_{\text{num}} < \delta_{\text{num}}$ and $c_{\text{var}} < \delta_{\text{var}}$, but the RS has collected a sufficient number ζ of RREQs, this also implies some sort of spatial correlation of received RREQs.
4. Timer $t_{g_{\text{ack}}}$ expires: If none of the above three conditions is met, it means that most RREQs are not correlated. Thus, the RS defers g_{ack} until timer $t_{g_{\text{ack}}}$ expires to collect more RREQs.

Parameters δ_{num} , δ_{var} , and ζ are thresholds to determine how quickly the RS should broadcast g_{ack} before timer $t_{g_{\text{ack}}}$ expires. The threshold ζ can be calculated by past statistics. We will discuss how to adaptively adjust δ_{num} and δ_{var} in Section 2.4. If any condition above is met, the RS broadcasts a g_{ack} containing all packet indices in its T_{RS} .

As to the content of the g_{ack} , our design will try to exploit both spatial and temporal correlation of RREQs to inhibit potential future RREQs. There are two rules.

1. In the space domain, g_{ack} will contain all packet indices that have received so far. This prevents nearby MDs from sending duplicate requests. Specifically, when a MD sends the request of a lost packet p_i to the RS, other neighboring MDs that also lose packet p_i but have not sent their requests yet will cancel their submissions after receiving g_{ack} . In Fig. 2.2, suppose that MD₂'s RREQ contains packet indices p_x

and p_y . If the g_{ack} from the RS also contains p_x and p_y , MD_1 will remove (p_x, d_x, t_x^L) and (p_y, d_y, t_y^L) from its T_{MD} .

2. In the time domain, the RS tries to *predict* future lost packets. Specifically, if it finds a sequence of packet indices $p_{n-\beta}, p_{n-\beta+1}, \dots, p_n$ in its T_{RS} , the g_{ack} will include α more packet indices $p_{n+1}, p_{n+2}, \dots, p_{n+\alpha}$. So as to exploit temporal correlation of RREQs. Here α and β are system parameters.

After sending g_{ack} , the RS cancels timer $t_{g_{\text{ack}}}$ if it has not expired yet, resets its c_{num} and c_{var} to zeros, sends a bulk request containing all of the packet indices in g_{ack} to the BS, and clears its T_{RS} . Note that to reduce the overhead of packet headers and alleviate network contention, each RS also packs small requests together and then sends one bulk request to the BS.

For the BDH problem, we suggest dividing RSs into multiple *multicast groups* according to their relative positions in deployment, where adjacent RSs will be grouped together [81, 36]. Since MDs usually handover to adjacent RS cells, this behavior could make the received RREQs exhibit spatial correlation in each multicast group of RSs. Therefore, the BS can take advantage of multicast to alleviate BDH by simultaneously sending the requested packets to the RSs in the same multicast group. This issue will be further discussed later.

After obtaining the packets from the BS, the RS will send these packets to all its MDs through broadcasting. Using broadcasting has two advantages over using unicasting. First, the RS does not need to record which packets are lost by which MD, thus reducing storage and maintenance costs (especially when there are many handovering MDs). Second, the RS can significantly alleviate the amount of transmissions if most MDs lose similar packets. In addition, even when there are many handovering MDs, broadcasting can help reduce the cost of sending *orphan* packets (that is, those packets no longer requested by any MD).

2.3.3 Operations of BSs

Recall that RSs will be divided into multiple multicast groups according to their positions. For each multicast group of RSs, the BS will collect (and pack) their requests and send a bulk request to the DVB-H server to ask for retransmissions of lost packets. Then, the BS will multicast the requested packets to the member RSs in the multicast group. The above multicast takes advantage of spatial correlation of requests since MDs in some adjacent RS cells may lose similar packets. Besides, the handovering behavior of MDs extends this spatial correlation to neighboring RS cells. To take advantage of temporal correlation of requests, the BS can prefetch a small number of future correlated packets from the DVB-H server. In this way, the downloading latency can be reduced.

Note that the goal of this work is to reduce redundant requests to avoid bandwidth waste and network congestion. How to guarantee video QoS requirements, such as jitter and PSNR, is out of the scope of this work. Therefore, BR-LW adopts c_{num} and c_{var} to help RSs decide when to broadcast g_{ack} to alleviate redundant requests. For the delay concern, since packets have deadlines and MDs will drop those out-of-date packets, we consider a packet with *no delay* if the packet can be received by the MD before passing its deadline. Therefore, BR-LW makes MDs to notify the RS of packet deadlines so that these lost packets can be sent to MDs before they become out of date.

2.4 Determining System Parameters of BR-LW

In this section, we discuss how to adaptively adjust the system parameters of BR-LW according to the network situation.

2.4.1 Adjustment of lazy-wait timer t_{lazy}

The system parameter t_{lazy} determines how long a MD *waits* to send its RREQ after it incurs any packet loss. Intuitively, a shorter length of t_{lazy} is adopted if the MD incurs a bad DVB-H channel. In this case, since the MD is expected to lose a large number of

packets, the MD has to send its RREQ immediately to recover the lost packets. Otherwise, a longer length of t_{lazy} is adopted to make the MD accumulate more requests into its RREQ. Therefore, we adjust the length of t_{lazy} in a *binary exponential manner* as follows:

$$t_{\text{lazy}} = (d_i - t_{\text{current}}) \times 2^{-\mu_E}, \quad (2.1)$$

where t_{current} is the current time and μ_E is the expected number of *successive* lost packets by the MD.

To calculate μ_E , we adopt a four-state *aggregated Markov process (AMP)*, which can approximate the behavior of a DVB-H channel (even when the MD is moving) [68]. AMP considers a finite-state channel error model, where the states are divided into two groups \mathcal{S}_C and \mathcal{S}_E that correspond to the *correct* and *erroneous* reception of packets, respectively. Each group is associated with an output symbol that emits at each state transition and ends at the given group. In particular, let $\mathcal{X} = \{\mathcal{X}_0, \mathcal{X}_1, \dots\}$ be a time-homogenous Markov chain with a four-state space $\mathcal{S} : \{1, 2, 3, 4\}$. We can divide \mathcal{S} into two groups $\mathcal{S}_C : \{1, 2\}$ and $\mathcal{S}_E : \{3, 4\}$. Let $\Phi : \mathcal{S} \rightarrow \{c, e\}$ be an *emission function*, where $\Phi(i) = c$ if $i \in \mathcal{S}_C$ and $\Phi(i) = e$ if $i \in \mathcal{S}_E$. If we denote the output process of \mathcal{X} as $\mathcal{Y} = \{\mathcal{Y}_0, \mathcal{Y}_1, \dots\}$ with state space (c, e) , then \mathcal{Y} is an AMP.

The transition probability matrix of \mathcal{X} is defined as

$$\mathcal{M}_{\mathcal{X}} = [P_{ij}]_{4 \times 4} = \begin{bmatrix} \rho_1 & 0 & (1 - \rho_1)\omega_3 & (1 - \rho_1)\omega_4 \\ 0 & \rho_2 & (1 - \rho_2)\omega_3 & (1 - \rho_2)\omega_4 \\ (1 - \rho_3)\omega_1 & (1 - \rho_3)\omega_2 & \rho_3 & 0 \\ (1 - \rho_4)\omega_1 & (1 - \rho_4)\omega_2 & 0 & \rho_4 \end{bmatrix},$$

where P_{ij} is the transition probability from state i to state j , $1 \leq i, j \leq 4$; the condition $0 \leq \rho_i \leq 1$ holds for $i = 1, 2, 3, 4$; ω_i s, $i = 1, 2, 3, 4$, are the weights such that $\omega_1 + \omega_2 = 1$ and $\omega_3 + \omega_4 = 1$.

Let $\pi^{(n)} = [\pi_1^{(n)} \pi_2^{(n)} \pi_3^{(n)} \pi_4^{(n)}]$ be the vector of state probability distribution, where each $\pi_i^{(n)}$, $i = 1, 2, 3, 4$, is the unconditional probability of being in state i at time n . Let

$\pi = \lim_{n \rightarrow \infty} \pi^{(n)} = [\pi_1 \ \pi_2 \ \pi_3 \ \pi_4]$. If the limit exists, π is a stationary distribution. Thus, we can calculate π by $\pi \cdot \mathcal{M}_{\mathcal{X}} = \pi$ and $\pi \cdot \mathbf{1} = 1$. Therefore, we can obtain that

$$\pi_i = \frac{\omega_i}{(1 - \rho_i) \sum_{j=1}^4 \frac{\omega_j}{1 - \rho_j}}, \quad \text{for } i = 1, 2, 3, 4.$$

With π , we can calculate the probability that a MD incurs erroneous reception of packets by

$$\begin{aligned} Prob(\mathcal{Y} = e) &= \sum_{i \in \mathcal{S}_E} \pi_i \\ &= \pi_3 + \pi_4 \\ &= \frac{\omega_3}{(1 - \rho_3) \sum_{j=1}^4 \frac{\omega_j}{1 - \rho_j}} + \frac{1 - \omega_3}{(1 - \rho_4) \sum_{j=1}^4 \frac{\omega_j}{1 - \rho_j}}. \end{aligned} \quad (2.2)$$

On the other hand, the value of $Prob(\mathcal{Y} = e)$ can be measured by the average *packet error rate* of the MD during a fixed observation window. Therefore, the moving behavior of the MD is also considered because it will impose an effect on the received signal strength. By assigning weights $\omega_1, \omega_2, \omega_3$, and ω_4 , we can adopt the Levenberg-Marquardt algorithm[54] to calculate the variables ρ_1, ρ_2, ρ_3 , and ρ_4 in Eq. (2.2).

Now, Let \mathcal{D}_E be the discrete random variable of the dwell time in state group \mathcal{S}_E . The probability mass function of \mathcal{D}_E is defined by [68]

$$\begin{aligned} P_{\mathcal{D}}(n) &\triangleq Prob(\mathcal{D}_E = n) \\ &= Prob(\mathcal{X}_{j+2} \in \mathcal{S}_E, \dots, \mathcal{X}_{j+n} \in \mathcal{S}_E, \\ &\quad \mathcal{X}_{j+n+1} \in \mathcal{S}_C \mid \mathcal{X}_j \in \mathcal{S}_C, \mathcal{X}_{j+1} \in \mathcal{S}_E) \\ &= \sum_{i=3}^4 \omega_i \rho_i^{n-1} (1 - \rho_i), \quad \text{for } n = 1, 2, \dots. \end{aligned}$$

Then, the probability-generating function of \mathcal{D}_E is calculated by

$$\begin{aligned}
G_{\mathcal{D}}(z) &= \sum_{n=1}^{\infty} P_{\mathcal{D}}(n)z^n \\
&= \sum_{n=1}^{\infty} \left(\sum_{i=3}^4 \omega_i \rho_i^{n-1} (1 - \rho_i) \right) \cdot z^n \\
&= \sum_{n=1}^{\infty} \left(\sum_{i=3}^4 \frac{\omega_i (1 - \rho_i)}{\rho_i} \right) \cdot \rho_i^n z^n \\
&= \sum_{i=3}^4 \frac{\omega_i (1 - \rho_i)}{\rho_i} \cdot \sum_{n=1}^{\infty} (\rho_i z)^n \\
&= \sum_{i=3}^4 \frac{\omega_i (1 - \rho_i)}{\rho_i} \cdot \frac{\rho_i z}{1 - \rho_i z}
\end{aligned} \tag{2.3}$$

Let us define $f_i(z) = \frac{\rho_i z}{1 - \rho_i z}$. Then, Eq. (2.3) is written as

$$G_{\mathcal{D}}(z) = \sum_{i=3}^4 \frac{\omega_i (1 - \rho_i)}{\rho_i} \cdot f_i(z).$$

The k th derivative of $f_i(z)$ is calculated by

$$\begin{aligned}
f_i^{(k)}(z) &= \frac{d^{k-1}}{dz^{k-1}} (f_i'(z)) \\
&= \frac{d^{k-1}}{dz^{k-1}} \left(\frac{\rho_i (1 - \rho_i z) - \rho_i z (-\rho_i)}{(1 - \rho_i z)^2} \right) \\
&= \rho_i \cdot \frac{d^{k-1}}{dz^{k-1}} \left(\frac{1}{(1 - \rho_i z)^2} \right) \\
&= \frac{k! \rho_i^k}{(1 - \rho_i z)^{k+1}}
\end{aligned}$$

Therefore, the k th derivative of $G_{\mathcal{D}}(z)$ is calculated by

$$G_{\mathcal{D}}^{(k)}(z) = \sum_{i=3}^4 \frac{k! \rho_i^{k-1} \omega_i (1 - \rho_i)}{(1 - \rho_i z)^{k+1}}, \quad \text{for } k = 1, 2, \dots \tag{2.4}$$

By Eq. (2.4), we can calculate the expectation of \mathcal{D}_E as

$$\mu_E = G_{\mathcal{D}}^{(1)}(1) = \frac{\omega_3}{1 - \rho_3} + \frac{1 - \omega_3}{1 - \rho_4}. \tag{2.5}$$

By substituting Eq. (2.5) into Eq. (2.1), we can obtain t_{lazy} .

2.4.2 Adjustment of threshold δ_{num}

The threshold δ_{num} determines how quickly the RS will broadcast g_{ack} . When $c_{\text{num}} \geq \delta_{\text{num}}$, the RS sends a g_{ack} because the received RREQs may exhibit temporal correlation. Let

$\mathcal{R} = \{p_1, p_2, \dots, p_{|\mathcal{R}|}\}$ be the set of the lost packets requested by RREQs that the RS collects during the $t_{g_{\text{ack}}}$ period. The size of \mathcal{R} is

$$\mathcal{N}_R \leq \gamma \times e_{MD} \times t_{g_{\text{ack}}},$$

where γ is the packet transmission rate of the DVB-H server and e_{MD} is the average packet error rate of a MD. Then, we adopt the *Zipf's law* [101] to calculate δ_{num} . According to the Zipf's law, many types of data studied in the physical and social sciences such as on-demand broadcasting and video-on-demand services [8] can be approximated by *Zipfian distributions*. Thus, we model the arrival of RREQs by a Zipfian distribution. Besides, due to the lazy wait scheme, we assume that a continuous or near-continuous sequence of packet indices are put together in a single RREQ if the packet loss exhibits temporal correlation.

Suppose that the packets in \mathcal{R} are ranked *decreasingly* by the number of RREQs that query them. The Zipf's law predicts that out of these \mathcal{N}_R packets, the frequency of a packet with rank k that appears in RREQs is

$$f(p_k) = \frac{\frac{1}{k^\theta}}{\sum_{i=1}^{\mathcal{N}_R} \frac{1}{i^\theta}},$$

where θ is an exponent characterizing the Zipfian distribution. For instance, supposing that $\mathcal{N}_R = 5$ and $\theta = 0.6$, the frequencies of packets p_1, p_2, p_3, p_4 , and p_5 appearing in RREQs are 0.33, 0.22, 0.17, 0.15, and 0.13, respectively. In other words, about 33%, 22%, 17%, 15%, and 13% RREQs request packets p_1, p_2, p_3, p_4 , and p_5 , respectively. Therefore, we calculate δ_{num} by

$$\sum_{i=1}^{\delta_{\text{num}}} f(p_i) \geq F_{th},$$

where F_{th} is the expected cumulative frequency of the first δ_{num} packets that most frequently appear in RREQs. Clearly, we have $F_{th} > 0.5$ (that is, the first δ_{num} packets are requested by more than half of all RREQs). For instance, we can set $F_{th} = 0.7$ and thus calculate $\delta_{\text{num}} = 3$ in the previous example. In this case, since the first three packets are requested by at least 70% RREQs, the RS can broadcast g_{ack} without waiting the remaining 30% RREQs.

2.4.3 Adjustment of threshold δ_{var}

The threshold δ_{var} also determines how quickly the RS will broadcast g_{ack} . When $c_{\text{var}} \geq \delta_{\text{var}}$, the RS sends a g_{ack} because the received RREQs may exhibit spatial correlation.

Let $\varepsilon_1, \varepsilon_2, \dots$, and $\varepsilon_{\mathcal{N}_R}$ be the minimum values of n_1, n_2, \dots , and $n_{\mathcal{N}_R}$ that trigger the RS to broadcast a g_{ack} , respectively, where $\varepsilon_i > 0$ for all $i = 1..\mathcal{N}_R$. In other words, if $n_i \geq \varepsilon_i$ for *any* $i = 1..\mathcal{N}_R$, the RS sends a g_{ack} . Then, δ_{var} is calculated by the coefficient of variation of all ε_i s, $i = 1..\mathcal{N}_R$:

$$\delta_{\text{var}} = \frac{\sqrt{\frac{1}{\mathcal{N}_R} \sum_{i=1}^{\mathcal{N}_R} (\varepsilon_i - \varepsilon_{\text{avg}})^2}}{\varepsilon_{\text{avg}}}, \quad (2.6)$$

where $\varepsilon_{\text{avg}} = \frac{1}{\mathcal{N}_R} \sum_{i=1}^{\mathcal{N}_R} \varepsilon_i$. Note that the sum of all ε_i s, $i = 1..\mathcal{N}_R$, is smaller than ζ (otherwise, by case 3 in Section 2.3.2, the RS sends a g_{ack} due to $\sum_{i=1}^{\mathcal{N}_R} n_i \geq \zeta$). Thus, we have

$$\sum_{i=1}^{\mathcal{N}_R} \varepsilon_i \leq \sum_{i=1}^{\mathcal{N}_R} n_i < \zeta. \quad (2.7)$$

The values of $\varepsilon_1, \varepsilon_2, \dots$, and $\varepsilon_{\mathcal{N}_R}$ can be calculated by an optimization problem whose objective is to minimize the total communication cost between the RS and its MDs. In particular, let $\Theta(\zeta)$ be the number of messages exchanged between the RS and its MDs during the $t_{g_{\text{ack}}}$ period, under the constraint in Eq. (2.7). Therefore, we have

$$\Theta(\zeta) = \mathcal{N}_{RREQ} + \mathcal{N}_{g_{\text{ack}}},$$

where \mathcal{N}_{RREQ} is the number of RREQs sent by MDs and $\mathcal{N}_{g_{\text{ack}}}$ is the number of g_{ack} sent by the RS. Since the RS sends only one g_{ack} during the $t_{g_{\text{ack}}}$ period, we have $\mathcal{N}_{g_{\text{ack}}} = 1$. Thus, the above equation is written as

$$\Theta(\zeta) = \mathcal{N}_{RREQ} + 1. \quad (2.8)$$

Our goal is to find the values of $\varepsilon_1, \varepsilon_2, \dots$, and $\varepsilon_{\mathcal{N}_R}$ such that $\Theta(\zeta)$ is minimized.

Since $\sum_{i=1}^{\mathcal{N}_R} n_i \geq \zeta$ will trigger the RS to send a g_{ack} , the maximum number of RREQs received by the RS before sending g_{ack} is ζ . Therefore, before the condition $c_{\text{var}} \geq \delta_{\text{var}}$ is

satisfied, the expectation of \mathcal{N}_{RREQ} is

$$\begin{aligned}
& E[\mathcal{N}_{RREQ}] \\
&= \zeta \times \text{Prob}(\text{(no } g_{\text{ack}} \text{ is sent due to } c_{\text{var}} < \delta_{\text{var}}) \mid (\text{no } g_{\text{ack}} \\
&\quad \text{is sent due to } \sum_{i=1}^{\mathcal{N}_R} n_i < \zeta)) \\
&= \zeta \times \text{Prob}((n_i < \varepsilon_i, \forall p_i \in \mathcal{R}) \mid (\sum_{i=1}^{\mathcal{N}_R} n_i < \zeta)) \\
&= \zeta \times \frac{\text{Prob}((n_i < \varepsilon_i, \forall p_i \in \mathcal{R}) \cap (\sum_{i=1}^{\mathcal{N}_R} n_i < \zeta))}{\text{Prob}(\sum_{i=1}^{\mathcal{N}_R} n_i < \zeta)}. \tag{2.9}
\end{aligned}$$

We assume that each MD sends its RREQs *independently* with other MDs. By Eq. (2.7), we obtain that

$$\begin{aligned}
& \text{Prob}((n_i < \varepsilon_i, \forall p_i \in \mathcal{R}) \cap (\sum_{i=1}^{\mathcal{N}_R} n_i < \zeta)) \\
&= \text{Prob}(n_i < \varepsilon_i, \forall p_i \in \mathcal{R}) \times \text{Prob}(\sum_{i=1}^{\mathcal{N}_R} n_i < \zeta) \\
&= \prod_{i=1}^{\mathcal{N}_R} \text{Prob}(n_i < \varepsilon_i) \times \text{Prob}(\sum_{i=1}^{\mathcal{N}_R} n_i < \zeta). \tag{2.10}
\end{aligned}$$

By substituting Eq. (2.10) into Eq. (2.9), we have

$$E[\mathcal{N}_{RREQ}] = \zeta \times \prod_{i=1}^{\mathcal{N}_R} \text{Prob}(n_i < \varepsilon_i). \tag{2.11}$$

Thus, by Eq. (2.11), Eq. (2.8) is written as

$$\Theta(\zeta) = \zeta \times \prod_{i=1}^{\mathcal{N}_R} \text{Prob}(n_i < \varepsilon_i) + 1. \tag{2.12}$$

Since we have $\varepsilon_i > 0$ for each $i = 1.. \mathcal{N}_R$, we can obtain that $\text{Prob}(n_i < \varepsilon_i) > 0$. Besides, since ζ is a given constant, the only way to minimize $\Theta(\zeta)$ in Eq. (2.12) is to minimize the value of $\text{Prob}(n_i < \varepsilon_i)$ for each $i = 1.. \mathcal{N}_R$.

Recall that each n_i , $i = 1.. \mathcal{N}_R$, is the number of RREQs received by the RS. Thus, a combination of all n_i s reflects the statistical quantity of RREQs that should follow some distribution (*e.g.*, Zipfian distribution). Based on the definition, ε_i is the minimum value of n_i , $i = 1.. \mathcal{N}_R$, that triggers the RS to broadcast a g_{ack} . Thus, the combination of all ε_i s also follows the same distribution. Therefore, the value of $\text{Prob}(n_i < \varepsilon_i)$ is expected

to increase when n_i is close to ε_i for each $i = 1..N_R$. Based on the above observation, we define a *distance function* of n_i and ε_i as follows:

$$H(n_i, \varepsilon_i) = \begin{cases} \frac{\max(n_i, \varepsilon_i)}{\min(n_i, \varepsilon_i)} & \text{if } n_i < \varepsilon_i \\ 1 & \text{otherwise.} \end{cases}$$

When $n_i < \varepsilon_i$, we have $H(n_i, \varepsilon_i) > 1$ and its value decreases when n_i is more close to ε_i . On the other hand, when $n_i \geq \varepsilon_i$, we define the value of $H(n_i, \varepsilon_i)$ to be one. By transforming the probability $Prob(n_i < \varepsilon_i)$ to the distance function $H(n_i, \varepsilon_i)$, our objective (to minimize $\Theta(\zeta)$ in Eq. (2.12)) becomes

$$\text{minimize } \prod_{i=1}^{N_R} H(n_i, \varepsilon_i). \quad (2.13)$$

By taking logarithm in Eq. (2.13), we calculate the optimal values of $\varepsilon_1, \varepsilon_2, \dots$, and ε_{N_R} to minimize $\Theta(\zeta)$ by the following optimization equations:

$$\begin{aligned} & \text{minimize} \\ & \sum_{i=1}^{N_R} \log H(n_i, \varepsilon_i), \\ & \text{subject to} \\ & n_i > 0 \quad \text{for } i = 1..N_R, \\ & \varepsilon_i > 0 \quad \text{for } i = 1..N_R, \\ & 0 < \sum_{i=1}^{N_R} \varepsilon_i \leq \zeta. \end{aligned} \quad (2.14)$$

We can use the Karush-Kuhn-Tucker (KKT) algorithm [15] to solve Eq. (2.14). Then, by applying the values of all ε_i s, $i = 1..N_R$, into Eq. (2.6), we can calculate δ_{var} .

2.5 Performance Evaluation

In this section, we verify the effectiveness of BR-LW by simulations. In our simulations, there is one BS supporting four RSs, as shown in Fig. 2.1. These four RSs belong to the same multicast group, and each RS serves ten MDs initially. We measure the amounts of

RREQ submissions from the MDs and packet retransmissions from the RS under different distributions of *DVB-H packet loss* \mathcal{F} and *correlation feature of RREQs* \mathcal{L} . The two distributions \mathcal{F} and \mathcal{L} are normal distributions and independent with each other. For distribution \mathcal{F} , we consider three scenarios:

- **Left-skew:** The peak of \mathcal{F} appears in the left-hand side.

$$\mathcal{F}(\mathcal{X}_{\mathcal{F}} = 0 \sim \mathcal{T}, \mu_{\mathcal{F}} = \frac{1}{4}\mathcal{T}, \sigma_{\mathcal{F}} = \frac{1}{5}\mathcal{T}).$$

- **Balanced:** The peak of \mathcal{F} appears in the middle.

$$\mathcal{F}(\mathcal{X}_{\mathcal{F}} = 0 \sim \mathcal{T}, \mu_{\mathcal{F}} = \frac{1}{2}\mathcal{T}, \sigma_{\mathcal{F}} = \frac{1}{5}\mathcal{T}).$$

- **Right-skew:** The peak of \mathcal{F} appears in the right-hand side.

$$\mathcal{F}(\mathcal{X}_{\mathcal{F}} = 0 \sim \mathcal{T}, \mu_{\mathcal{F}} = \frac{3}{4}\mathcal{T}, \sigma_{\mathcal{F}} = \frac{1}{5}\mathcal{T}).$$

Here, $\mathcal{X}_{\mathcal{F}}$, $\mu_{\mathcal{F}}$, and $\sigma_{\mathcal{F}}$ are the random variable, mean value, and standard deviation of \mathcal{F} , respectively, and \mathcal{T} is the period of simulation time. Distribution \mathcal{F} models the needs of sending RREQs by MDs. The three scenarios, namely *left-skew*, *balanced*, and *right-skew*, indicate that a mass of packet loss occur at time around $0.25\mathcal{T}$, $0.5\mathcal{T}$, and $0.75\mathcal{T}$, respectively.

For distribution \mathcal{L} , we consider four scenarios:

- **Spatially-middle:** Spatially-correlated RREQs are generated in the middle of the simulation.

$$\mathcal{L}(\mathcal{X}_{\mathcal{L}} = 0 \sim \mathcal{T}, \mu_{\mathcal{L}} = \frac{1}{2}\mathcal{T}, \sigma_{\mathcal{L}} = \frac{1}{10}\mathcal{T}).$$

- **Spatially-late:** Spatially-correlated RREQs are generated in the late portion of the simulation.

$$\mathcal{L}(\mathcal{X}_{\mathcal{L}} = 0 \sim \mathcal{T}, \mu_{\mathcal{L}} = \frac{3}{4}\mathcal{T}, \sigma_{\mathcal{L}} = \frac{1}{10}\mathcal{T}).$$

- **Temporally-middle:** Temporally-correlated RREQs are generated in the middle of the simulation.

$$\mathcal{L}(\mathcal{X}_{\mathcal{L}} = 0 \sim \mathcal{T}, \mu_{\mathcal{L}} = \frac{1}{2}\mathcal{T}, \sigma_{\mathcal{L}} = \frac{1}{5}\mathcal{T}).$$

- **Temporally-late:** Temporally-correlated RREQs are generated in the late portion of the simulation.

$$\mathcal{L}(\mathcal{X}_{\mathcal{L}} = 0 \sim \mathcal{T}, \mu_{\mathcal{L}} = \frac{3}{4}\mathcal{T}, \sigma_{\mathcal{L}} = \frac{1}{5}\mathcal{T}).$$

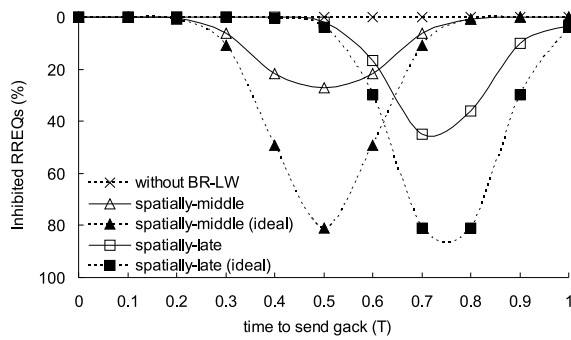
Here, $\mathcal{X}_{\mathcal{L}}$, $\mu_{\mathcal{L}}$, and $\sigma_{\mathcal{L}}$ are the random variable, mean value, and standard deviation of \mathcal{L} , respectively. Distribution \mathcal{L} models the inhibited submissions of RREQs after the RS broadcasts a g_{ack} . To model the spatial correlation of RREQs, the two scenarios, namely *spatially-middle* and *spatially-late*, indicate that a mass of correlated RREQs are generated at time around $0.5\mathcal{T}$ and $0.75\mathcal{T}$, respectively. The similar scenarios (that is, *temporally-middle* and *temporally-late*) are also applied to model the temporal correlation of RREQs.

2.5.1 Ratio of Inhibited RREQs

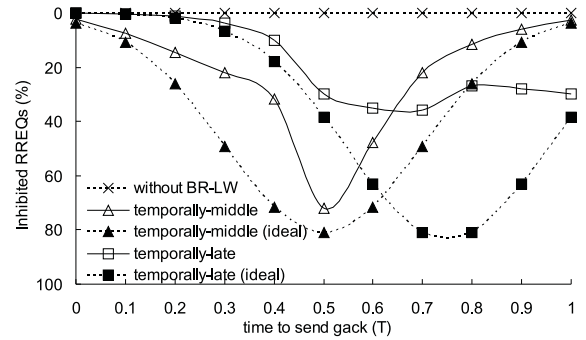
We first evaluate the ratio of inhibited RREQs by BR-LW. To show the effect of g_{ack} timing, we do not apply the adjustment mechanisms in Sections 2.4.2 and 2.4.3. Instead, we let the RS send a g_{ack} once at *fixed* time $0.1\mathcal{T}$, $0.2\mathcal{T}$, \dots , and \mathcal{T} . Note that the RS sends only one g_{ack} and MDs still submit their RREQs after receiving a g_{ack} .

Fig. 2.3 shows the ratio of inhibited RREQs under different distributions of \mathcal{F} and \mathcal{L} . For comparison, we consider an *ideal* case where only one MD sends RREQs and other MDs are silent before the RS sending g_{ack} . As can be seen, g_{ack} timing significantly affects the ratio of inhibited RREQs. The peak (that is, the maximum ratio of inhibited RREQs) of the ideal case in the spatially-middle (respectively, temporally-middle) scenario of \mathcal{L} locates at time $0.5\mathcal{T}$ because a mass of spatially (respectively, temporally) correlated RREQs are generated at time $0.5\mathcal{T}$. In this case, most MDs can find similar requests from the g_{ack} and thus avoid sending duplicate RREQs. Similarly, the peak of the ideal case in the spatially-late (respectively, temporally-late) scenario of \mathcal{L} appears at time between $0.7\mathcal{T}$ and $0.8\mathcal{T}$ since a mass of spatially (respectively, temporally) correlated RREQs are generated at time $0.75\mathcal{T}$. However, since the RS will put its predicted future requests in the g_{ack} if RREQs exhibit temporal correlation, the effect of g_{ack} timing is

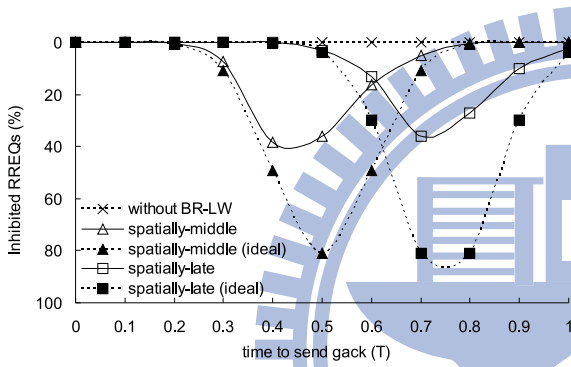
less significant in the temporally-middle and temporally-late scenarios compared to the spatially-middle and spatially-late scenarios of \mathcal{L} .



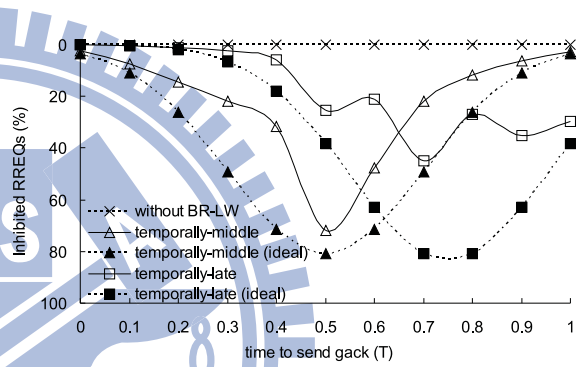
(a) \mathcal{F} : left-skew, \mathcal{L} : spatial correlation



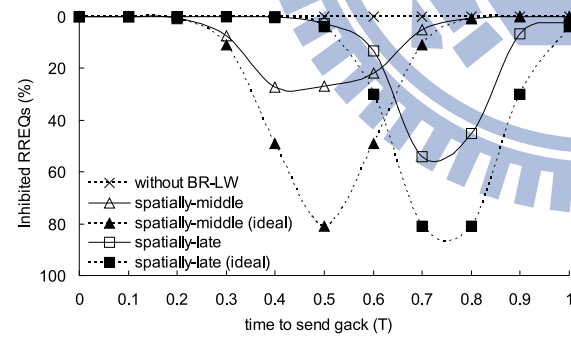
(b) \mathcal{F} : left-skew, \mathcal{L} : temporal correlation



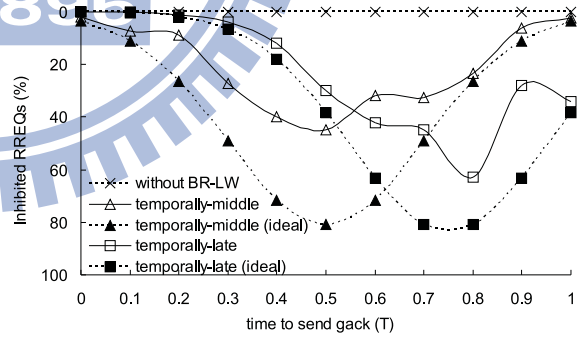
(c) \mathcal{F} : balanced, \mathcal{L} : spatial correlation



(d) \mathcal{F} : balanced, \mathcal{L} : temporal correlation



(e) \mathcal{F} : right-skew, \mathcal{L} : spatial correlation



(f) \mathcal{F} : right-skew, \mathcal{L} : temporal correlation

Figure 2.3: The ratio of inhibited RREQs by BR-LW under different distributions of \mathcal{F} and \mathcal{L} .

We then discuss the effect of \mathcal{F} on inhibited RREQs. From Fig. 2.3(a), (c), and (e), we observe that the peak of spatially-middle (respectively, spatially-late) scenario of \mathcal{L} locates at time between $0.4T$ and $0.5T$ (respectively, $0.7T$ and $0.8T$), which is close to

that of the ideal case. In the spatially-middle scenario of \mathcal{L} , the balanced scenario of \mathcal{F} has the highest ratio of inhibited RREQs (refer to Fig. 2.3(c)). The reason is that most MDs lose a mass of packets at time around $0.5\mathcal{T}$ and these packets exhibit high spatial correlation. On the other hand, in the spatially-late scenario of \mathcal{L} , the right-skew scenario of \mathcal{F} has the highest ratio (refer to Fig. 2.3(e)). Similarly, since most MDs incur serious packet loss at time around $0.75\mathcal{T}$ and these packets also exhibit high spatial correlation, sending a g_{ack} at time $0.75\mathcal{T}$ can reduce the most duplicate RREQs.

From Fig. 2.3(b), (d), and (f), we observe that the peak of temporally-middle scenario of \mathcal{L} appears at time $0.5\mathcal{T}$, which is the same as that of the ideal case. The peak of temporally-late scenario of \mathcal{L} appears at time $0.7\mathcal{T}$ and $0.8\mathcal{T}$, which is close to that of the ideal case. The ratio of inhibited RREQs in the temporally-late scenario of \mathcal{L} oscillates because a mass of temporally-correlated RREQs are generated late (at time $0.75\mathcal{T}$) and thus the RS may not predict many future requests. Comparing Fig. 2.3(b), (d), and (f), in the temporally-middle scenario of \mathcal{L} , the right-skew scenario of \mathcal{F} has the lowest ratio. The reason is that most MDs lose their packets in the late portion of the simulation and these packets may be uncorrelated (since a mass of temporally-correlated RREQs are generated in the middle of the simulation). However, when most MDs lose their packets in the late portion of the simulation (that is, the right-skew scenario of \mathcal{F} in Fig. 2.3(f)), the temporally-late scenario of \mathcal{L} has the highest ratio because the peaks of distributions \mathcal{F} and \mathcal{L} overlap.

2.5.2 Reduction of Packet Retransmissions

Using the same setting in the previous section, we then measure the reduced amount of packet retransmissions by BR-LW. Recall that the RS will broadcast the packets after sending g_{ack} . Thus, the g_{ack} timing can be approximately viewed as the timing to broadcast packets. To show the effect of such broadcasting timing, after the RS broadcasts the packets (at the time of sending g_{ack}), we allow MDs to receive unicast packets and submit RREQs simultaneously.

Fig. 2.4 shows the amount of packet retransmissions under different distributions \mathcal{F} and \mathcal{L} . Without BR-LW, the RS always unicasts the requested packets to MDs after receiving RREQs. In the *ideal* case, we assume that the RS can know the lost packets by *all* MDs before sending g_{ack} . From Fig. 2.4, g_{ack} timing significantly affects the amount of packet retransmissions. The peak (that is, the minimum amount of packet retransmissions) of the ideal case in the spatially-middle (respectively, temporally-middle) scenario of \mathcal{L} locates at time $0.5\mathcal{T}$ because a mass of spatially (respectively, temporally) correlated RREQs are generated at time $0.5\mathcal{T}$. In this case, most of the retransmitted packets can meet the requirements of most MDs. Similarly, the peak of the ideal case in the spatially-late (respectively, temporally-late) scenario of \mathcal{L} appears at time between $0.7\mathcal{T}$ and $0.8\mathcal{T}$ since a mass of spatially (respectively, temporally) correlated RREQs are generated at time $0.75\mathcal{T}$. Again, the effect of g_{ack} timing is less significant in the temporally-middle and temporally-late scenarios compared to the spatially-middle and spatially-late scenarios of \mathcal{L} , since the RS predicts the future lost packets if RREQs exhibit temporal correlation.

Next, we discuss the effect of \mathcal{F} on packet retransmissions. In Fig. 2.4(a), the peaks of the three scenarios of \mathcal{F} appear at time $0.5\mathcal{T}$, which are the same as that of the ideal case. The right-skew scenario of \mathcal{F} has the smallest peak, because most MDs experience serious packet loss at time $0.75\mathcal{T}$ but most RREQs exhibit high spatial correlation at time $0.5\mathcal{T}$. In this case, a g_{ack} can only reflect the spatial correlation of a small number of RREQs, as it is sent early at time $0.5\mathcal{T}$. On the other hand, in Fig. 2.4(b), the peaks of the three scenarios of \mathcal{F} locate at time $0.7\mathcal{T}$ or $0.8\mathcal{T}$, which are close to that of the ideal case. The left-skew scenario of \mathcal{F} has the smallest peak. The reason is that most MDs experience serious packet loss at early time $0.25\mathcal{T}$ but most spatially-correlated RREQs are generated at late time $0.75\mathcal{T}$.

For the temporally-middle and temporally-late scenarios of \mathcal{L} (that is, Fig. 2.4(c) and (d)), the peaks of different scenarios of \mathcal{F} may not necessary be the same. The reason is that the RS will put its prediction of future lost packets in a g_{ack} . However, similar to Fig. 2.4(a) and (b), the right-skew (respectively, left-skew) scenario of \mathcal{F} in Fig. 2.4(c) (respectively, Fig. 2.4(d)) has the smallest peak, because g_{ack} can only reflect the temporal

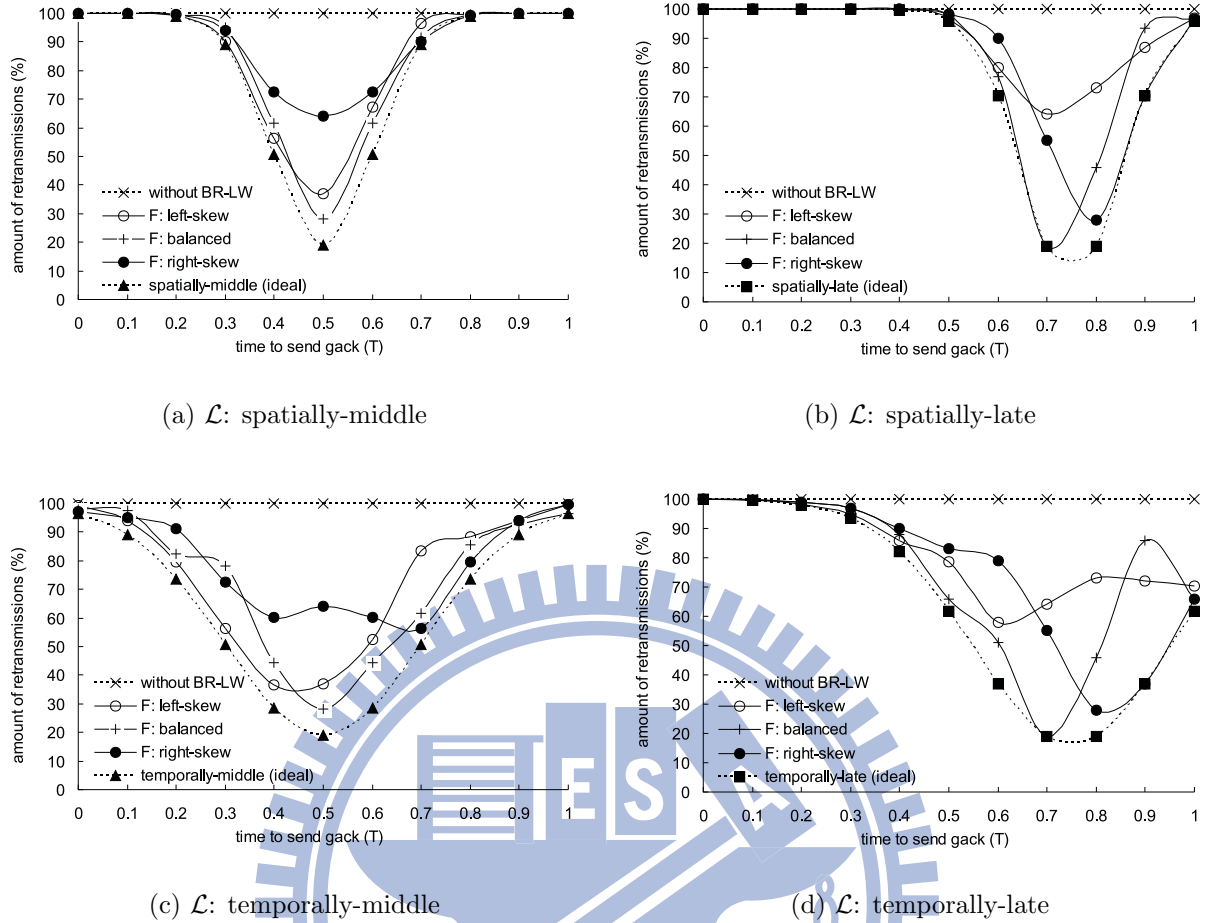


Figure 2.4: The amount of packet retransmissions by BR-LW under different distributions of \mathcal{F} and \mathcal{L} .

correlation of a small number of RREQs.

2.5.3 Adjustment of δ_{var}

Recall that the RS will send a g_{ack} when $c_{\text{var}} \geq \delta_{\text{var}}$, which means that the received RREQs may exhibit high spatial correlation. We also analyze how to adjust δ_{var} in Section 2.4.3. In this section, we verify the correctness of our analysis on δ_{var} . The simulation consists of 2000 iterations, each with a period of \mathcal{T} . In an iteration, we divide the time into 10 segments and check when the condition $c_{\text{var}} \geq \delta_{\text{var}}$ occurs. We adopt the spatially-middle and spatially-late scenarios of \mathcal{L} .

Fig. 2.5(a) shows when the condition $c_{\text{var}} \geq \delta_{\text{var}}$ occurs in the spatially-middle scenario of \mathcal{L} . From Figs. 2.3 and 2.4, we find that the optimal time to send a g_{ack} is $0.5\mathcal{T}$. We

define that the calculation of δ_{var} *hits* the optimal value if the time when $c_{\text{var}} \geq \delta_{\text{var}}$ occurs locates between time $0.4\mathcal{T}$ and $0.6\mathcal{T}$ (that is, we tolerate an error of $0.1\mathcal{T}$). Fig. 2.5(b) shows the hit rate of δ_{var} , where the *exact hit* means that $c_{\text{var}} \geq \delta_{\text{var}}$ occurs exactly at time $0.5\mathcal{T}$ and the *nearby hit* means that $c_{\text{var}} \geq \delta_{\text{var}}$ occurs at time either $0.4\mathcal{T}$ or $0.6\mathcal{T}$. As can be seen, in the spatially-middle scenario of \mathcal{L} , the overall hit rate (that is, the sum of exact hits and nearby hits) is always higher than 80%. Besides, the exact hit rate is always higher than 40%. The above result shows the correctness of our analysis on δ_{var} in the spatially-middle scenario of \mathcal{L} .

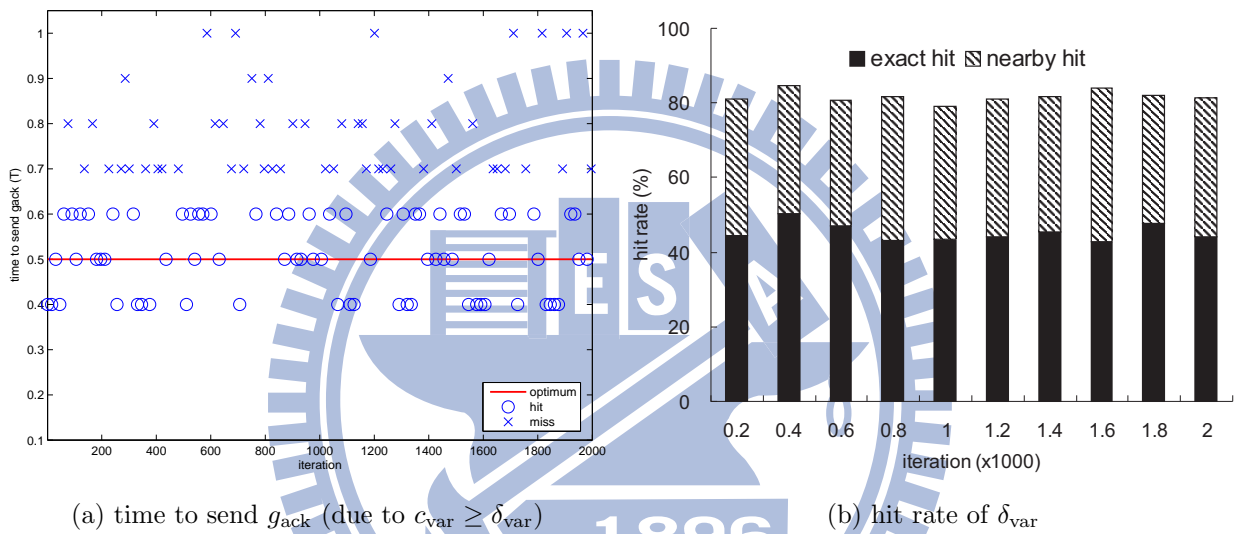


Figure 2.5: The time when $c_{\text{var}} \geq \delta_{\text{var}}$ occurs and the hit rate of δ_{var} in the spatially-middle scenario of \mathcal{L} .

Fig. 2.6(a) shows when the condition $c_{\text{var}} \geq \delta_{\text{var}}$ occurs in the spatially-late scenario of \mathcal{L} . Again, from Figs. 2.3 and 2.4, we find that the optimal time to send a g_{ack} is $0.8\mathcal{T}$ ³. The calculation of δ_{var} is said to *hit* the optimal value if the time when $c_{\text{var}} \geq \delta_{\text{var}}$ occurs locates between time $0.7\mathcal{T}$ and $0.9\mathcal{T}$. Fig. 2.6(b) shows the hit rate of δ_{var} , where the *exact hit* means that $c_{\text{var}} \geq \delta_{\text{var}}$ occurs exactly at time $0.8\mathcal{T}$ and the *nearby hit* means that $c_{\text{var}} \geq \delta_{\text{var}}$ occurs at time either $0.7\mathcal{T}$ or $0.9\mathcal{T}$. As can be seen, in the spatially-late scenario of \mathcal{L} , the overall hit rate is always higher than 85%. Besides, the exact hit rate

³The optimal value in fact locates at time between $0.7\mathcal{T}$ and $0.8\mathcal{T}$. For convenience, we take $0.8\mathcal{T}$ as the optimal value.

is always higher than 35%. The above result shows the correctness of our analysis on δ_{var} in the spatially-late scenario of \mathcal{L} .

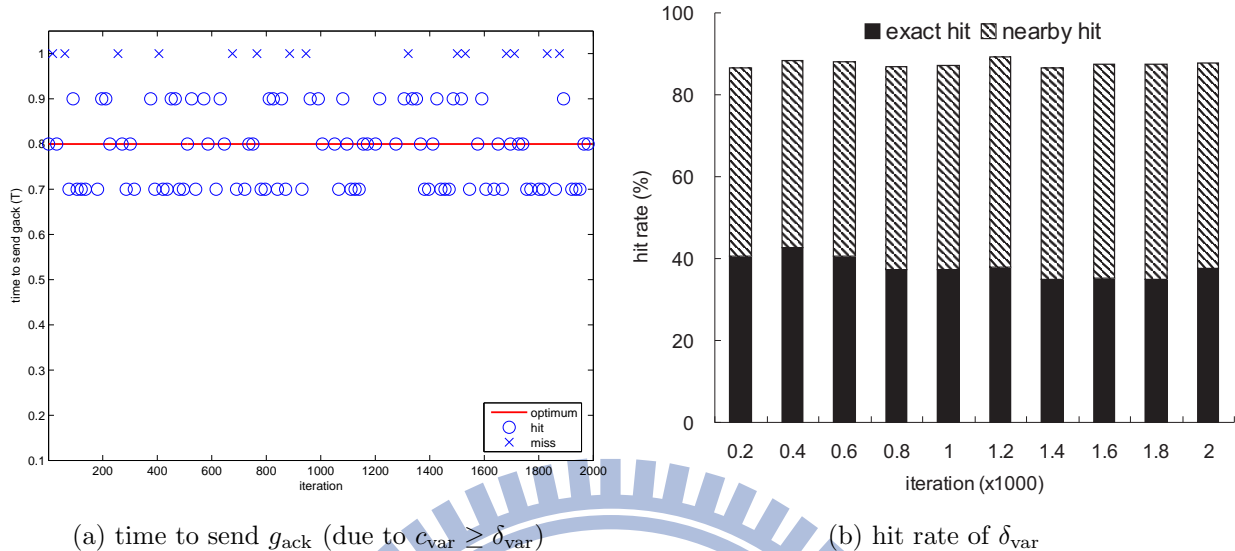


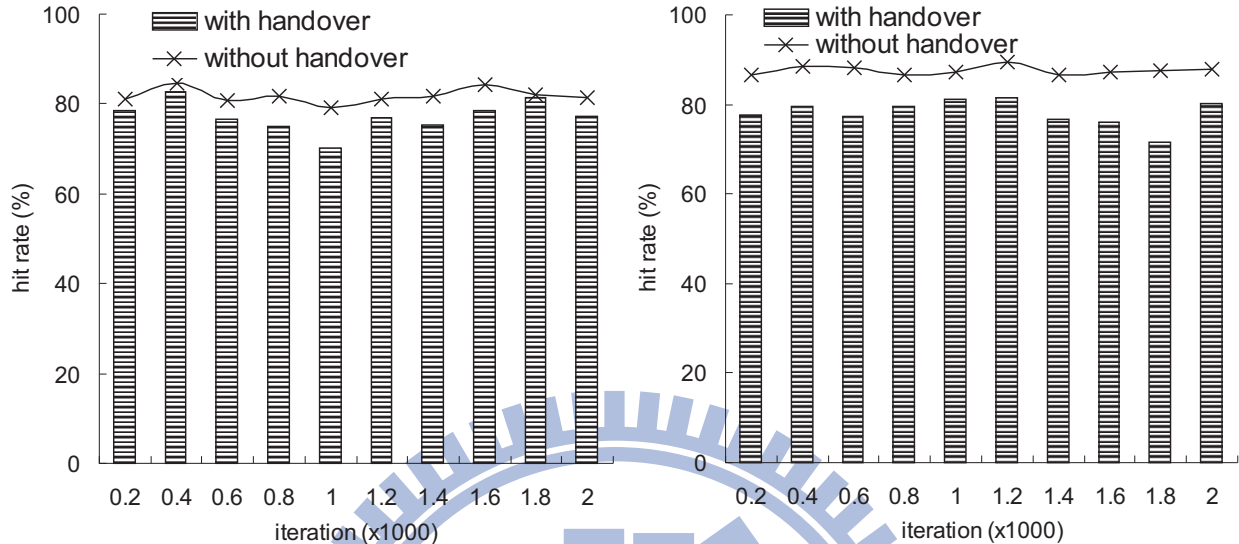
Figure 2.6: The time when $c_{\text{var}} \geq \delta_{\text{var}}$ occurs and the hit rate of δ_{var} in the spatially-late scenario of \mathcal{L} .

2.5.4 Effect of Handovers

The previous experiments consider the behavior only inside a single RS cell. In this section, we evaluate the effect of handovers. The mobility of each MD follows the random waypoint model, and thus the occurrence of handovers can be simulated by a uniform distribution. In addition, to measure the effect of handover on the timing of sending g_{ack} , we allow each MD to send out all of its “unqueried” RREQs to the new associated RS after handovering.

Fig. 2.7(a) and (b) show the hit rates of sending g_{ack} due to $c_{\text{var}} \geq \delta_{\text{var}}$ (including both exact and nearby hits) in the spatially-middle and spatially-late scenarios of \mathcal{L} , respectively. For comparison, we also observe the hit rate when there are no handovers. Because the handovering behavior of MDs extends the spatial correlation of RREQs to neighboring RS cells, our lazy-wait operation can help reduce the redundant requests sent from handovering MDs. Therefore, there is only a slight decrease of hit rate caused by handovering MDs. From Fig. 2.7, the decreases of hit rates in the spatially-middle and

spatially-late scenarios of \mathcal{L} are about 4.5% and 9.5%, respectively. On average, the hit rate always exceeds 75% even when there are handovers, which shows the effectiveness of our BR-LW scheme.



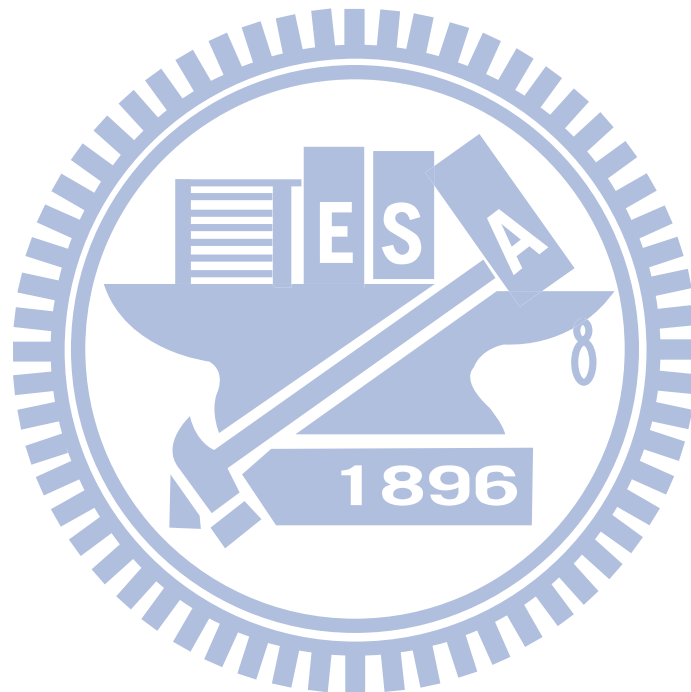
(a) hit rate of δ_{var} in the spatially-middle scenario of \mathcal{L} (b) hit rate of δ_{var} in the spatially-late scenario of \mathcal{L}

Figure 2.7: The hit rate of sending g_{ack} (due to $c_{\text{var}} \geq \delta_{\text{var}}$) when there are handovers.

2.6 Summary of Chapter 2

Recently, *DVB-H* (*digital video broadcasting-handheld*) and *DVB-IPDC* (*IP datacast over DVB-H*) have been developed to support broadcasting services. DVB-H is designed to support digital video broadcast for handheld devices, while DVB-IPDC can integrate with an IP-relay network to complement the data loss problem in DVB-H. Assuming that WiMAX networks are adopted to support DVB-IPDC, this chapter points out two critical problems: *group packet loss (GPL)* and *broadcast data handover (BDH)*. GPL occurs when there is a burst of retransmission requests for the same pieces of data with high spatial or temporal correlation. BDH happens when some devices that made the above requests handover to new serving cells. To solve these problems, we propose a *lazy wait* and a *group acknowledgement* schemes to alleviate duplicate requests by exploiting

their spatial and temporal correlation. This not only reduces the requests submitted by neighboring devices in both space and time domains, but also avoids handovering devices from sending duplicate requests in new cells. Through mathematical analysis, we show how to adaptively adjust the timers of lazy wait and group acknowledgement based on channel quality. Simulation results prove that our schemes can efficiently reduce retransmission requests and retransmission packets, thus alleviating congestion in the IP-relay network.



Chapter 3

Data Recovery by Using On-Demand Network Coding

3.1 Introduction

Digital video broadcasting–handheld (DVB-H) is developed to support multimedia broadcasting services for *mobile devices (MDs)* [26]. It adopts a time-slicing mechanism to reduce MDs' energy consumption and a forward error correction scheme to enhance data robustness. Recently, to solve the packet loss problem, *IP datacast over DVB-H (DVB-IPDC)* is proposed by incorporating another wireless network with IP datacasting [45], which can serve as a recovery channel for MDs to request for lost packets. Fig. 3.1 shows the network architecture considered in the chapter, where DVB-H and WiMAX networks coexist. The DVB-H server continually broadcasts digital videos to MDs. When an MD encounters packet loss, it requests a neighboring WiMAX *relay station (RS)* for retransmission. The WiMAX *base station (BS)* collects requests, queries the DVB-H server for the lost packets, and then retransmits them to MDs through RSs.

To improve the efficiency of the recovery job of the WiMAX networks, we consider the network coding techniques [4, 27]. Since MDs may miss different pieces of a video stream, network coding is very suitable. However, video data has real-time constraints needed

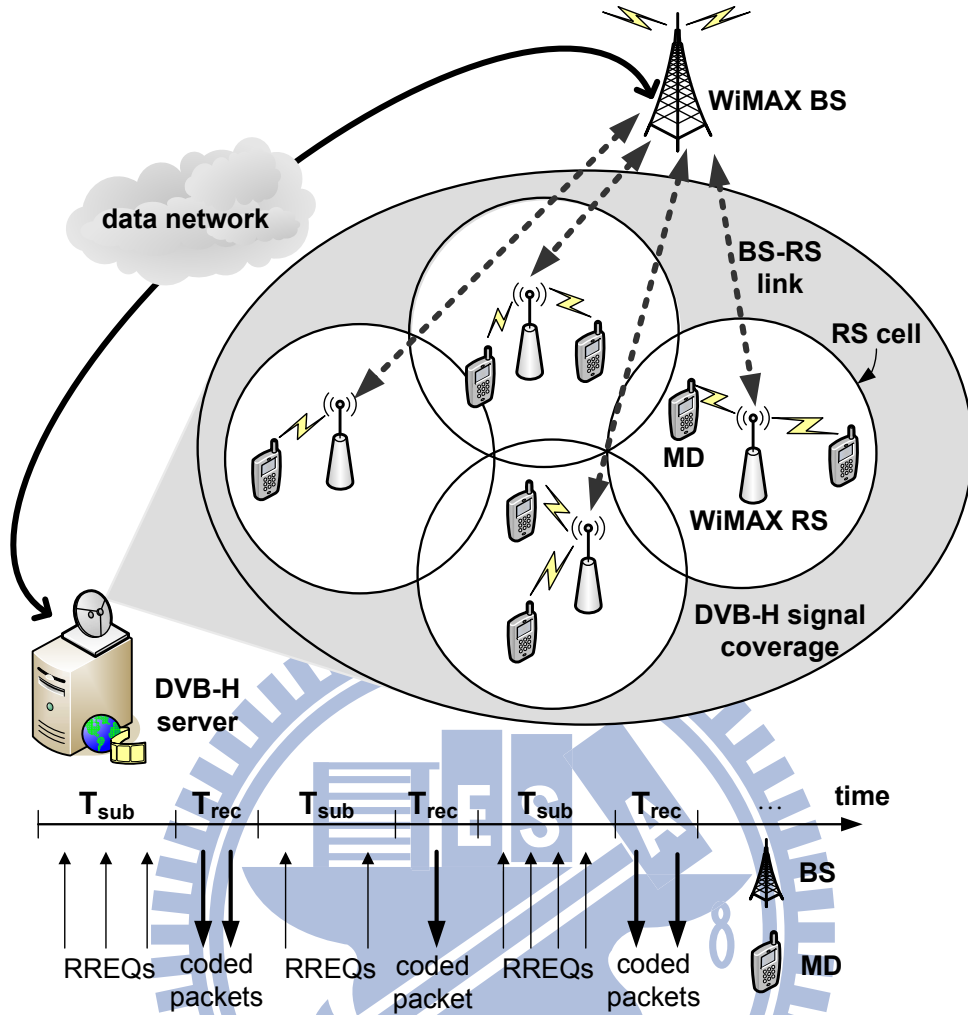


Figure 3.1: A DVB-IPDC system with WiMAX networks as the recovery channel.

to be addressed. In addition, lost packets in DVB-H often exhibit *spatial* and *temporal* correlations [92]. In the space domain, neighboring MDs may lose similar packets because they are interfered by similar noise sources. In the time domain, neighboring MDs may lose similar sequences of packets because interference sources usually exist for a while.

In the work, we model the WiMAX recovery channel as a repetitive train of frames. Each frame has a \mathcal{T}_{sub} submission period followed by a \mathcal{T}_{rec} recovery period, as shown in Fig. 3.1. MDs submit their retransmission requests during \mathcal{T}_{sub} and the WiMAX BS sends lost packets during \mathcal{T}_{rec} . To address the real-time constraint, each DVB-H packet is assigned a deadline. To best utilize each frame, we consider adopting network coding to recover MDs' packets. In particular, each \mathcal{T}_{rec} is modeled as k slots. Given some lost packets and their deadlines, we formulate a new *prioritized network coding problem*, whose

goal is to arrange at most k coded packets in each \mathcal{T}_{rec} period such that 1) the number of recovered packets per frame is maximized and 2) the number of dropped packets per frame is minimized. Existing network coding schemes for content distribution or multimedia streaming applications [100, 64, 42, 97, 17] aim at reducing network traffic and do not address the above priority issue.

To solve the prioritized network coding problem, we develop a *recovery by greedy selection (RGS)* and a *recovery by prioritized selection (RPS)* schemes based on XOR coding. RGS first constructs a weighted bipartite graph to reflect the influence of each coded packet if it is transmitted in the upcoming \mathcal{T}_{rec} period. The vertex set consists of all raw packets and their coded packets and the edge set reflects the relationships between raw packets and coded packets. Each edge's weight is calculated by the number of requests and the deadline of the corresponding lost packet. Then, RGS adopts a greedy strategy to find the maximum weight minimum dominating set from the graph to determine the coded packets to be broadcasted. On the other hand, RPS expands the above bipartite graph to adjust weights by traversing the selected coded packets and then find the maximum weight minimum dominating set from the graph by considering the selection sequence of coded packets. Simulation results show that both RGS and RPS can recover packets efficiency while reduce the number of dropped packets.

The rest of this chapter is organized as follows: Section 3.2 surveys related work and Section 3.3 defines our prioritized network coding problem. Section 3.4 presents the RGS and RPS schemes. Section 3.5 shows the simulation results. Conclusions are drawn in Section 3.6.

3.2 Related Work

In DVB-H services, MDs are vulnerable to transmission errors. To cope with this problem, DVB-IPDC employs another wireless network for recovery purpose. This recovery channel is responsible for retransmitting missing packets and maintaining smooth playback of broadcasting videos. Below, we review related recovery techniques.

3.2.1 Data Recovery in DVB-IPDC Systems

Several studies suggest adopting a cellular network as the recovery channel for DVB-IPDC. Reference[32] considers transmitting parity data over the DVB-H or cellular networks, where the cellular network may use either dedicated point-to-point connections or cell broadcasting. *Content delivery protocol* for regulating the cellular network's behaviors is defined in [24, 25]. Through extensive simulations, [35] points out several guidelines for choosing between point-to-point and point-to-multipoint repair mechanisms under different network situations. It is suggested to organize every three MDs into a group such that the MD with the strongest received signal strength in each group serves as a *super peer* responsible for data recovery in its group[38]. WiMAX networks are considered as the recovery channel in [90, 92], where the group packet loss property, which usually exhibits high spatial and temporal correlations, is discussed. Instead of using infrastructure networks, [79] proposes organizing MDs as a wireless ad hoc network to share lost packets through peer-to-peer links. However, all these studies do not consider incorporating network coding in the recovery process.

3.2.2 Network Coding for Content Distribution and Streaming Services

Network coding is widely used to improve communication efficiency. For content distribution, [30] addresses how to deliver large files in a distribution network, where each node in the network can generate and transmit encoded blocks of information. To make the content distribution of files more efficiently, a randomization concept is introduced in the coding process to facilitate the scheduling of block propagation. In the work of [53], a peer-to-peer content distribution system based on the sparse linear network coding scheme is proposed, where the system determines the encoding density to increase the probability of generating independent encoded block. Besides, an encoding interval is adopted to reduce the probability of transmitting linear dependent packets. The work of [49] considers using network coding for content distribution in vehicular ad hoc networks, and develops a file swarming protocol to handle the issues of topology change and inter-

mittent connectivity in the network. However, [30, 53, 49] do not address the real-time issue of the coded content.

For multimedia streaming applications, the work of [84] adopts a randomized push algorithm to take advantage of random network coding for live peer-to-peer streaming, where the objectives are to alleviate buffering delays, resist to peer dynamics, and reduce bandwidth costs on the streaming servers. The work of [99] develops a data-driven overlay network for live multimedia streaming, where each node in the network periodically exchanges data availability information with a set of partners, retrieves unavailable data from partners, and supplies available data to partners. By adopting random network coding, the work of [51] develops an on-demand video streaming system whose objective is to minimize the bandwidth cost and buffering delay of each streaming server in a large-scale network. Explicitly, the above research efforts aim at using network coding to help reduce the bandwidth cost. In contrast, our work considers how to efficiently select coded packets with priority concerns to recover the lost packets of MDs in limited time.

3.2.3 Network Coding for Error Recovery

Several studies consider adopting network coding to deal with the error recovery issue for data transmission. The work of [48] extends the multiuser *automatic repeat request (ARQ)* to multicasting applications. Based on the feedback information of the receivers with their successfully received data, the sender calculates the code weights for the linear combination of data packets. Then, each receiver can decode the linearly combined packets by exploiting its previously received data. The work of [9] considers a network that consists of a central server and several caching clients, where data are transmitted over a broadcast channel. The server creates ad-hoc error-correction sets according to the states of clients and then adopts erasure-correction codes for data transmissions. Each client uses its cached data and the received data to derive its requested blocks. By clustering MDs into multiple cells, the work of [87] adaptively codes the data according to the data temporarily stored in each MD to minimize the bandwidth consumption. Such a cell selection problem

and a broadcast coding problem are formulated by integer programming and shown to be NP-hard. However, none of the above work considers our prioritized network coding problem. Therefore, in this chapter we investigate how to use a limited number of coded packets to satisfy the most number of MDs' recovery requests while reduce the number of dropped packets due to passing their deadlines.

3.3 Problem Definition

Fig. 3.1 illustrates our DVB-H system with a WiMAX network as the recovery channel. Each MD is equipped with a DVB-H receiver and a WiMAX interface. The WiMAX network consists of multiple BSs. Each BS can support multiple RSs. Whenever an MD finds itself losing some packets, it sends a *recovery request (RREQ)* to its neighboring RS, through unicasting, for retransmissions. Each RREQ contains the indices of the lost packets and the corresponding deadlines. The BS then aggregates these RREQs, queries the lost packets from the DVB-H server, and forwards them (if feasible) to the MDs.

To save the downlink bandwidth from the BS to MDs, we consider using network coding. We model the the WiMAX channel as in Fig. 3.1, where the time axis is modeled by a repetitive frame of a \mathcal{T}_{sub} submission period followed by a \mathcal{T}_{rec} recovery period. MDs are allowed to submit their RREQs during each \mathcal{T}_{sub} period and the BS retransmits the lost packets during each \mathcal{T}_{rec} period. We assume that \mathcal{T}_{rec} contains k slots so that the BS is allowed to broadcast at most k packets during each \mathcal{T}_{rec} period. In addition, after collecting RREQs at the end of each \mathcal{T}_{sub} period, the BS has the most updated information of the lost and successfully received packets of each MD.

Given the lost packets to be recovered and their deadlines, the *prioritized network coding problem* asks how the BS uses at most k coded packets to recover the lost packets of MDs in every \mathcal{T}_{rec} period such that 1) the maximum number of RREQs is satisfied and 2) the number of dropped packets by MDs due to out of deadlines is minimized. To solve this problem, we propose RGS and RPS schemes in the next section.

3.4 The Proposed Solutions

We are given n MDs, where each MD $_i$ may send one RREQ r_i in every \mathcal{T}_{sub} period. For each round, t_{curr} records the time that the \mathcal{T}_{sub} period ends and k is the duration of the successive one \mathcal{T}_{rec} after the \mathcal{T}_{sub} . Each r_i contains a set of packets s_i *successfully* received by MD $_i$ and a set of packets q_i *queried* by MD $_i$ (due to loss) in the current frame. Assume that the BS knows the deadline d_j of each packet p_j , which can be achieved by querying the DVB-H server. The packet p_j in each s_i and q_i shall be discarded whenever $d_j \leq t_{curr}$. The recovery deadline d_{r_i} to the RREQ r_i is culled the earliest d_j corresponded to the queried packet p_j from the set q_i . For convenience, let $\mathcal{R} = \bigcup_{i=1..n} r_i$, $\mathcal{S} = \bigcup_{i=1..n} s_i$, and $\mathcal{Q} = \bigcup_{i=1..n} q_i$. In other words, \mathcal{R} is the set of all RREQs whose length is n , \mathcal{S} is the set of packets successfully received by MDs, and \mathcal{Q} is the set of packets queried by MDs in the current frame. Then, our solutions first construct a weighted graph to reflect the relationship among all coded packets and the queried packets in \mathcal{Q} , and find a maximum-weight minimum dominating set from the graph to select the coded packets to be broadcasted by the BS under the limitation of recovery deadlines from each d_{r_i} .

3.4.1 Recovery by Greedy Selection (RGS)

RGS considers a two-operand XOR coding approach, that is, each coded packet is generated by conducting the XOR operation (denoted by ' \oplus ') on two packets. Then, RGS constructs a weighted bipartite graph $\mathcal{G} = (\mathcal{Q} \cup \mathcal{C}, \mathcal{Q} \times \mathcal{C})$ such that the vertex set contains \mathcal{Q} (all queried packets) and \mathcal{C} (all coded packets). $\mathcal{C} = \{c_1, \dots, c_k\}$ and each coded packet $c_k = p_x \oplus p_y$ where $p_x \in \{0\} \cup \mathcal{S}$, $p_y \in \mathcal{Q}$, and $p_x \neq p_y$. In particular, each coded packet in \mathcal{C} must be either a queried packet in \mathcal{Q} (that is, XORing with a zero-string '0') or the result by XORing a successfully-received packet in \mathcal{S} and a queried packet in \mathcal{Q} . For the edge set $\mathcal{Q} \times \mathcal{C}$, we allow each edge (p_i, c_j) accompanying with its corresponded weight $\omega_{i,j}$ where $p_i \in \mathcal{Q}$ and $c_j \in \mathcal{C}$. The weight $\omega_{i,j}$ represents the effect on decoding the coded packet c_j to recover back the queried packet p_i . To each edge (p_i, c_j) , if c_j is an operand

that can decode to generate p_i , then the weight $\omega_{i,j} > 0$ (how to assign the weight will be discussed later); otherwise, the weight $\omega_{i,j} = 0$ as long as c_j is helpless for getting back p_i . Fig. 3.2 gives an example on the conditions of the current time frame $t_{curr} = 10$ and the retransmission time slots $k = 3$. As illustrated in Fig. 3.2(a), four RREQs r_1, r_2, r_3 and r_4 are collected on a BS in a \mathcal{T}_{sub} frame, where $\mathcal{S} = \{p_1, p_2, p_3, p_4\}$, $\mathcal{Q} = \{p_1, p_2, p_4\}$, and their recovery deadlines $d_{r_1}, d_{r_2}, d_{r_3}$ and d_{r_4} , respectively. Based on these RREQs, the coded packets are $\mathcal{C} = \{c_1, c_2, \dots, c_9\}$ as shown in Fig. 3.2(b). At the edge (p_1, c_4) , labeled as (1), p_1 in \mathcal{Q} cannot be generated by decoding c_4 according to the contents of \mathcal{R} , therefore $\omega_{1,4} = 0$; while at edge (p_1, c_5) , labeled as (2), $\omega_{1,5} = 1/12$ because decoding c_5 can generate p_1 for the r_3 in \mathcal{R} .

The value of each weight $\omega_{i,j}$ consists of two parts: one is $\alpha_{i,j}$ which reflects how many RREQs are influenced (*i.e.*, need the queried packet p_i) by decoding the coded packet c_j ; the other is $\tau_{i,j}$ that measures how much time is left before recovery deadlines corresponding to RREQs. We let $\omega_{i,j} = \alpha_{i,j} \times \tau_{i,j}$ because the larger amount of the needed RREQs and the shorter remaining time for recovery imply the higher priority of the coded packet c_j to be used, that is the value of $\omega_{i,j}$ is larger. $\alpha_{i,j} = \frac{\|\mathcal{D}_j^i\|}{\|\mathcal{R}\|}$ which gives the ratio of the number of the influenced RREQs by decoding the coded packet c_j for recovering the queried packet p_i to the whole \mathcal{R} . \mathcal{D}_j^i is the set of the influenced RREQ r_n , such that

$$\mathcal{D}_j^i = \bigcup_{\forall r_n \in \mathcal{R}} r_n \text{ where } (p_i = c_j) \text{ or } (\exists p_m \in \mathcal{S} \text{ s.t. } p_m \oplus p_i = c_j). \quad (3.1)$$

$\|\mathcal{D}_j^i\|$ and $\|\mathcal{R}\|$ are the numbers of the RREQs in the sets \mathcal{D}_j^i and \mathcal{R} , respectively. $\tau_{i,j} = \sum_{\forall r_n \in \mathcal{R}, r_n \in \mathcal{D}_j^i} \frac{1}{\max((d_{r_n} - t_{curr}), k)}$ that is defined by the inverse ratio before the expiry recovery time with respect to each influenced RREQ in \mathcal{D}_j^i . d_{r_n} is the valid expiration of recovery time for r_n , t_{curr} is the current timestamp, k is maximum retransmission time slots which is used as the length of sliding window for retransmissions (hereof, d_{r_n} , t_{curr} and k adopt the same time unit). Therefore, the equation of $\omega_{i,j}$ can be given as follows.

$$\omega_{i,j} = \alpha_{i,j} \times \tau_{i,j} = \sum_{\forall r_n \in \mathcal{R}, r_n \in \mathcal{D}_j^i} \frac{\|\mathcal{D}_j^i\|}{\|\mathcal{R}\| \times \max((d_{r_n} - t_{curr}), k)}. \quad (3.2)$$

After constructing the graph \mathcal{G} , we then find the maximum weight minimum domi-

RREQ	$p_1,$ $d_1=11$	$p_2,$ $d_2=12$	$p_3,$ $d_3=13$	$p_4,$ $d_4=14$	deadline
r_1	O	X	O	X	$d_{r_1}=12$
r_2	O	O	O	X	$d_{r_2}=14$
r_3	X	X	O	O	$d_{r_3}=11$
r_4	O	X	O	O	$d_{r_4}=12$

$$s_1=\{p_1, p_3\}, q_1=\{p_2, p_4\} \quad s_2=\{p_1, p_2, p_3\}, q_2=\{p_4\}$$

$$s_3=\{p_3, p_4\}, q_3=\{p_1, p_2\} \quad s_4=\{p_1, p_3, p_4\}, q_4=\{p_2\}$$

(a)

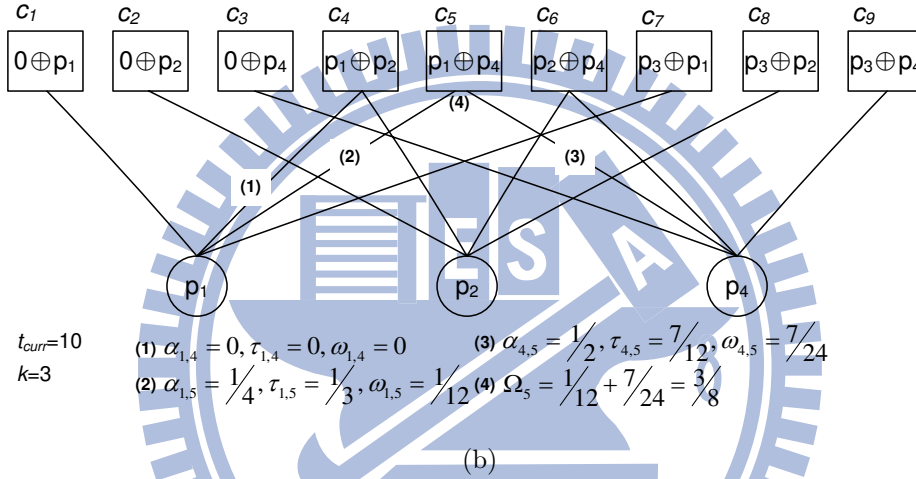


Figure 3.2: An example of RGS: (a) the received RREQs, where ‘O’ means a successful reception while ‘X’ means an erroneous reception and (b) the weighted bipartite graph, where a rectangular vertex belongs to \mathcal{C} while a circular vertex belongs to \mathcal{Q} .

nating set on \mathcal{G} . Our goal is to find the minimum number of vertices in \mathcal{C} (that is, coded packets) such that they can dominate the maximum number of vertices in \mathcal{Q} (that is, queried packets) and the total weights of the corresponding edges is maximized. That is finding out the most retransmitted coded packets whose total weights are maximal and can serve for recovering maximal queried packets in a \mathcal{T}_{rec} frame. Specifically, Ω_j represents the selection priority of a coded packet c_j . Based on the means of $\omega_{i,j}$, Ω_j is the sum of all $\omega_{i,j}$ s where each corresponded queried packet p_i belongs to \mathcal{Q} . Thus, from Eq. (3.2),

for any coded packet $c_j \in \mathcal{C}$ whose Ω_j is calculated by

$$\Omega_j = \sum_{\forall p_i \in \mathcal{Q}} \omega_{i,j} = \sum_{\forall p_i \in \mathcal{Q}} \sum_{\forall r_n \in \mathcal{R}, r_n \in \mathcal{D}_j^i} \frac{\|\mathcal{D}_j^i\|}{\|\mathcal{R}\| \times \max((d_{r_n} - t_{curr}), k)}. \quad (3.3)$$

For instance, in Fig. 3.2(b), Ω_5 , labeled as (4), sums the related $\omega_{1,5}$ and $\omega_{4,5}$ which are labeled as (2) and (3), respectively. Besides, the dominator f_j for any coded packet c_j is the set that contains any queried packet p_i in \mathcal{Q} whose corresponded edge (p_i, c_j) with $\omega_{i,j} > 0$. That is

$$f_j = \bigcup_{\forall p_i \in \mathcal{Q}} p_i \text{ where } \omega_{i,j} > 0, \quad (3.4)$$

and the dominator set for all coded packets \mathcal{F} is the union of all f_j s, hence,

$$\mathcal{F} = \bigcup_{\forall c_j \in \mathcal{C}} f_j. \quad (3.5)$$

Based on these information, RGS conducts the following iterative loop process to find the maximum weight minimum dominating set on \mathcal{G} .

We let the solution set \mathcal{A} be the set of the selected f_j s and the set of queried packets that have not been covered by any f_j in \mathcal{A} be \mathcal{U} amid the iterative loop procedure. The default value of \mathcal{A} is empty while \mathcal{U} contains all queried packets in \mathcal{Q} (*i.e.*, $\mathcal{U} = \mathcal{Q}$). Below two steps are conducted in each iteration of the process loop till \mathcal{U} becomes empty ($\mathcal{U} = \phi$).

- First, we choose one f_j from \mathcal{F} respecting to any coded packet c_j if the condition $\max(f_j \cap \mathcal{U})$ exists. That means the most of the same queried packets are contained in f_j and \mathcal{U} whereas the corresponding weight Ω_j is also maximal. Otherwise, if no such f_j can be found, we turn to select another one f_x from \mathcal{F} whose Ω_x is the largest.
- Second, we shall choose any one coded packet c_j in the prior step. So, we remove the corresponding f_j from \mathcal{U} (*i.e.*, $\mathcal{U} = \mathcal{U} - \{f_j\}$) which represents the queried packets dominated by f_j have been recovered, and add c_j into \mathcal{A} ($\mathcal{A} = \mathcal{A} \cup \{c_j\}$).

Finally, we shape the solution set \mathcal{A} whose size fits to k . Thus, if the number of the selected coded packets in \mathcal{A} is less than or equal to k , then the whole set of \mathcal{A} is the solution for RGS directly; otherwise, outputs the first- k coded packets in \mathcal{A} for the solution of RGS. The example in Fig. 3.2(b), the solution set $\mathcal{A} = \{c_8, c_6, c_5\}$ or $\{c_2, c_6, c_5\}$ under the condition of $k = 3$.

3.4.2 Recovery by Prioritized Selection (RPS)

Through finding out the maximum weight minimum dominating set of coded packets, RGS provides a way to generate the coded packets for retransmissions which fits the time constraint of k . Nevertheless, RGS does not consider the relation between the queried packets and the collected RREQs as well as the influence of the selection sequence amid the coded packets. RPS scheme takes these two concerns into consideration for the selection process of coded packets.

RPS adopts a two-operand XOR coding approach to generate coded packets and constructs a weighted bipartite graph $\mathcal{G} = (\mathcal{Q} \cup \mathcal{C}, \mathcal{Q} \times \mathcal{C})$, which is same to that of RGS. We say $\mathcal{C} = \{c_1, \dots, c_k\}$ in which each coded packet $c_k = p_x \oplus p_y$ where $p_x \in \{0\} \cup \mathcal{S}$, $p_y \in \mathcal{Q}$, and $p_x \neq p_y$. That is, each coded packet c_k consists of either a queried packet in \mathcal{Q} or the result by XORing a successfully-received packet in \mathcal{S} and a queried packet in \mathcal{Q} . Each edge (p_i, c_j) in the edge set $\mathcal{Q} \times \mathcal{C}$ associates with its corresponded weight $\omega_{i,j}$ where $p_i \in \mathcal{Q}$ and $c_j \in \mathcal{C}$. For each edge (p_i, c_j) , if c_j is an operand that is able to be decoded to generate p_i , then the weight $\omega_{i,j} > 0$; otherwise, the weight $\omega_{i,j} = 0$ which represents c_j is helpless for the recovery to p_i . The value of each weight $\omega_{i,j}$ is given by Eq. (3.2). The dominator f_j as Eq. (3.4) for any coded packet c_j is the set that contains any queried packet p_i in \mathcal{Q} and the dominator set for all coded packets \mathcal{F} is the union of all f_j s that is defined as Eq. (3.5).

After constructing the graph \mathcal{G} , RPS checks the relation between the queried packets and the collected RREQs to select coded packets. The situation of the much more queried packets requested by only some RREQs shall appear when the packets losses with spatial-

temporal locality feature occurred [92]. Thus, we shall be aware of the sequence of the sequence of the coded packets in a row while doing the selection process. Specifically, depending on the numbers of the queried packets and the collected RREQs, two cases are dealt with in RPS. The first case, when the number of the coded packets is not larger than the number of the collected RREQs, RPS performs the selection operations like that in RGS to find the maximum weight minimum dominating set on \mathcal{G} and comes out the solution set \mathcal{A} whose size fits to k . The second case is to cope with finding the maximum weight minimum dominating set for \mathcal{A} by considering the sequence of the selected coded packets as if the number of the queried packets is larger than the number of the collected RREQs. We still use the definitions of Eq. (3.4) and Eq. (3.5) for the dominator f_j and dominators set \mathcal{F} , then the solution set \mathcal{A} consists of the selected f_j s while the set of queried packets that have not been covered by any f_j in \mathcal{A} be \mathcal{U} amid the iterative selection process. The default value of \mathcal{A} is empty while \mathcal{U} contains all queried packets in \mathcal{Q} (*i.e.*, $\mathcal{U} = \mathcal{Q}$). Below three steps are conducted in each iteration of the selection process till \mathcal{U} becomes empty (*i.e.*, $\mathcal{U} = \phi$).

- First, we choose one f_j from \mathcal{F} respecting to any coded packet c_j if the condition $\max(f_j \cap \mathcal{U})$ exists. That means the most of the same queried packets are contained in f_j and \mathcal{U} whereas the corresponding weight Ω_j is also maximal. Otherwise, if no such f_j can be found, we turn to select another one f_x from \mathcal{F} whose Ω_x is the largest.
- Second, after we choose any one coded packet c_j in the prior step, we remove the queried packets contained in f_j from \mathcal{U} (*i.e.*, $\mathcal{U} = \mathcal{U} - \{f_j\}$) which represents the queried packets dominated by f_j have been recovered, and add c_j into \mathcal{A} (*i.e.*, $\mathcal{A} = \mathcal{A} \cup \{c_j\}$).
- Third, to reflect the influence of recovery once decoding the coded packet c_j in the prior step, we need to re-calculate the weight of each coded packet c_z which is related to c_j . That is, we say $c_j = p_x \oplus p_y$ and for each coded packet $c_z = p_a \oplus p_b$, if p_a (or p_b) = p_x (or p_y) then we calculate the corresponding weight based on Eq. (3.2).

So, $\omega_{a,z} = \sum_{\forall r_n \in \mathcal{R}, r_n \in \mathcal{D}_z^a} \frac{\|\mathcal{D}_z^a\|}{\|\mathcal{R}\| \times \max((d_{r_n} - t_{curr}), k)}$ or $\omega_{b,z} = \sum_{\forall r_n \in \mathcal{R}, r_n \in \mathcal{D}_z^b} \frac{\|\mathcal{D}_z^b\|}{\|\mathcal{R}\| \times \max((d_{r_n} - t_{curr}), k)}$.

After that, by Eq. (3.3), we aggregate the weights associated with the coded packet

$$c_z \text{ as } \Omega_z = \sum_{\forall p_i \in \mathcal{Q}} \omega_{i,z}.$$

Finally, we shape the solution set \mathcal{A} whose size fits to k . Thus, if the number of the selected coded packets in \mathcal{A} is less than or equal to k , then the whole set of \mathcal{A} is the solution for RPS directly; otherwise, outputs the first- k coded packets in \mathcal{A} for the solution of RPS.

3.5 Experiment Results

In this section, the performance of the recovery schemes are evaluated by simulations. We compare our RGS and RPS schemes with a no coding scheme and the ISCOD approach methods in which Least Difference Greedy (LDG) algorithm[9] and Demand-Oriented Pairing (DOP) algorithm[16] are simulated. In the following, we first present the simulation environment for the on-demand recovery to mobile broadcasting errors, then investigate the results with different parameters.

3.5.1 Simulation Model

The simulation environment is set up below to evaluate the recovery efficiency of each method. Assume the DVB-H broadcasting packets rate per time slot is fixed, so we give a specific number to \mathcal{T}_{sub} and the simulation traffic flows during a period of time \mathcal{T}_{sub} are also constant. Besides, \mathcal{T}_{rec} is proportional to \mathcal{T}_{sub} , which ranges 10% or 20%, \dots , or 50% of \mathcal{T}_{sub} . We model the packet loss happening in a period of time \mathcal{T}_{sub} to be a uniform distribution, thus any MD encounters the packet loss is fair which ranges 10% or 20%, \dots , or 50% of the amount of DVB-H broadcasting packets in a period of time \mathcal{T}_{sub} . Based on the packet loss distribution, we assume the MDs generate their RREQs with a specific exponential inter-arrival time and a mean arrival rate λ , in which any one RREQ requests only one queried packet to be retransmitted.

While conducting a recovery method, since each RREQ declares its recovery deadline, we sort the whole queried packets in advance by ascending order depending on their left time before due of recovery. Then, along with the sorted sequence of the queried packets, each method selects the coded packets that can decode for the queried packets by considering the factor of Ω (consists of deadlines and the number of the influenced RREQs). Moreover, We assume the pattern of the RREQs that recover their corresponding queried packets in time (*i.e.*, before due of their deadlines) is generated by following the Zipf distribution. Such that, let θ_i represent the proportion of the whole RREQs which will be affected when each method chooses a coded packet that can recover in time to the queried packet p_i , and $\theta_i = (\frac{1}{i})^\nu / \sum_{j=1}^{\|\mathcal{Q}\|} (\frac{1}{j})^\nu$ where ν is the skewness parameter, $1 \leq i \leq \|\mathcal{Q}\|$. In this simulation, compared to the no coding scheme, we come out the *Overlapped Recovery Rate* (Θ) from θ_i with $\nu = 0.6$ respecting to the whole RREQs in \mathcal{T}_{rec} . So, $\Theta = \sum_i \theta_i$ where i is any chosen coded packet in \mathcal{T}_{rec} . The larger percentage of Θ represents the simulated recovery method being more efficient in terms of much more RREQs and their requesting queried packets are recovered in time. We observe the changes of Θ depending on the varying retransmission time k and packet loss distribution as follows.

3.5.2 Simulation Result

The scheme of the ‘no coding’ always needs retransmitting all queried packets no matter what the value of retransmission time slot or packet loss distribution is, so it cannot contribute any overlapped recovery rate (*i.e.*, $\Theta = 0\%$). Based on the knowledge about the received DVB-H packets in any MD, LDG algorithm adopts a greedy approach to choose the coded packets while DOP algorithm selects the linear combination of coded packets with a weighting function for a on-demand broadcasting program. Both of the LDG and DOP algorithms consider the weights to satisfy the needs for maximum RREQs, but not includes the sequence order of the recovery deadlines of RREQs. RGS scheme relies on the knowledge of generating the weight Ω of each coded packet and its corresponded dominator relation to the queried packets. On the other hand, RPS scheme selects the

coded packets aside from checking the values of Ω_s , it takes the sequence of the recovery deadlines for queried packets into consideration.

We first conduct a simulation that a half of the DVB-H broadcasting packets are lost and their corresponding RREQs are submitted from MDs uniformly distributed in a period of time \mathcal{T}_{sub} . The retransmission time duration k for coded packets is varied from 10% to 50% of \mathcal{T}_{sub} and then investigating the changes of the overlapped recovery rate Θ . The larger number of k means there are much more coded packets can be retransmitted. Fig. 3.3 shows the changes of Θ respecting to the retransmission time slot k whereas the results are summarized in Table 3.1. By comparing the number of Θ in Fig. 3.3, RPS gets the highest number of Θ which is followed by RGS, DOP and LDG, sequentially. Although LDG, DOP and RGS adopt similar greedy search method for coded packets, RGS takes advantage of getting higher number of Θ because the due time factor is considered. In Table 3.1, once the retransmission time is increased about 10% of \mathcal{T}_{sub} each time, the standard deviation of Θ for LDG, DOP, RGS and RPS methods are 28.3%, 26.6%, 23.4%, and 19.6%, respectively.

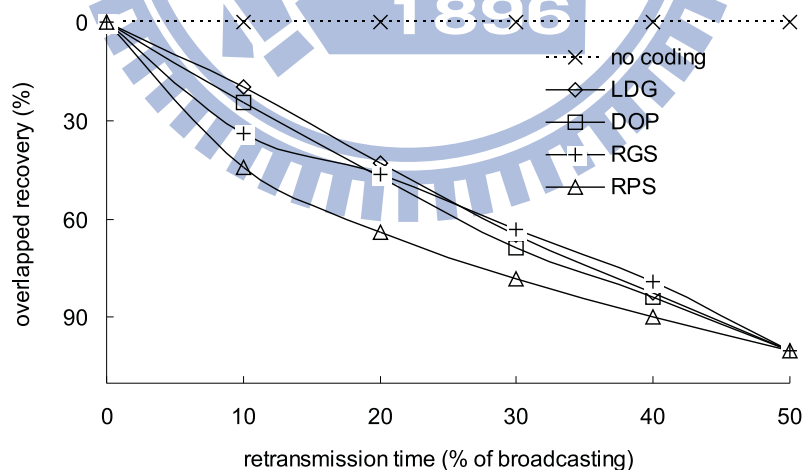


Figure 3.3: The changes of overlapped recovery rates respecting to different retransmission time durations while the packet loss rate is 50%.

Next, we simulate the recovery methods with regard to different packets losses whose loss probabilities are varied from 10% to 50% of \mathcal{T}_{sub} , and then investigate their corre-

Table 3.1: Overlapped recovery on varying retransmissions time.

reTX time (% \mathcal{T}_{sub})	10%	20%	30%	40%	50%	std. dev. (per 10%)	comparison (normalized)
no coding	0%	0%	0%	0%	0%	0%	0
LDG	19.8%	42.9%	65.1%	82.3%	100%	28.3%	1
DOP	24.5%	47.1%	68.8%	83.7%	100%	26.6%	1.1
RGS	33.9%	46.5%	63.1%	79.1%	100%	23.4%	1.2
RPS	44.3%	63.8%	78.2%	89.9%	100%	19.6%	1.4

sponding results of Θ . The retransmission time duration $k = 10\%$ of \mathcal{T}_{sub} . The larger probability of packet loss represents there should be much more coded packets occurred and needed to retransmit. Fig. 3.4 shows the simulation results of the changes of Θ and summarizes these results in Table 3.2. As illustrated in Fig. 3.4, RPS is better than RGS in terms of the comparison of their Θ and both RPS and RGS are also superior to the LDG and DOP. When the packet loss rate continues increasing 10% each time but the retransmission time is fixed at 10% of \mathcal{T}_{sub} , the standard deviation of Θ for LDG, DOP, RGS, and RPS methods are 28.6%, 27.8%, 25.9%, and 19.9%, respectively.

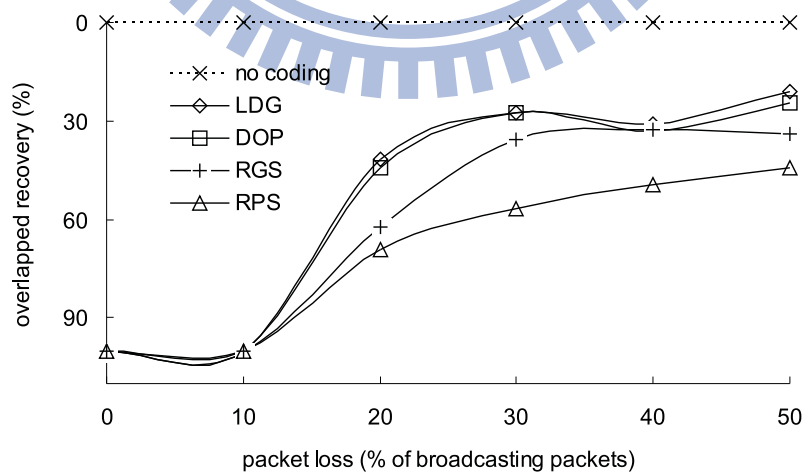


Figure 3.4: The changes of overlapped recovery rates respecting to different percentage of packet loss while the retransmission time is fixed at 10% of \mathcal{T}_{sub} .

Table 3.2: Overlapped recovery under different packet loss rates.

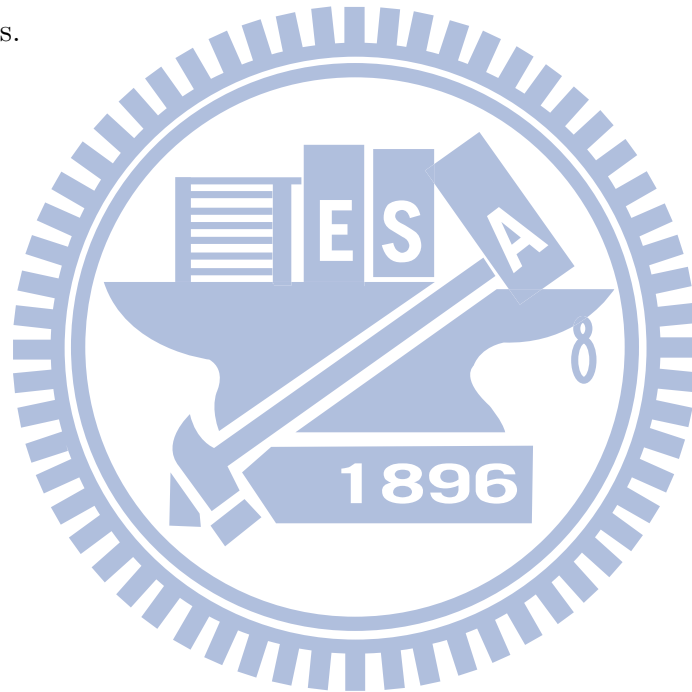
pkt loss	10%	20%	30%	40%	50%	std. dev.	comparison
						(per 10%)	(normalized)
no coding	0%	0%	0%	0%	0%	0%	0
LDG	100%	41.5%	27.6%	30.9%	21.2%	28.6%	1
DOP	100%	44.1%	27.6%	33.1%	24.5%	27.8%	1.1
RGS	100%	62.3%	35.8%	32.6%	33.9%	25.9%	1.3
RPS	100%	69.4%	56.7%	49.3%	44.3%	19.9%	1.7

Finally, if there is not enough retransmission time for the recovery methods to deal with much more broadcasting packets losses, we compare the numbers of Θ s of DOP, RGS, and RPS with that of LDG. Based on Table 3.1, when the retransmission time k is not enough (but not too short) for recovering all of the queried packets, the Θ s for DOP, RGS, and RPS are on average 1.1, 1.2, and 1.4 times of that of adopting LDG method, respectively. On the other hand, when the retransmission time k is too short to recovering all of the queried packets, averagely, from Table 3.2 the Θ s for DOP, RGS, and RPS are 1.1, 1.3, and 1.7 times of that of adopting LDG method, respectively. The simulation results appeal that RPS and RGS can keep the efficiency by decoding the coded packets for recovering much more queried packets while the due time factor of recovery is considered.

3.6 Summary of Chapter 3

Mobile broadcasting services are available and believed to become more popular in broadcast and wireless networks. However, because of the nature of wireless broadcast, how to efficiently recover packet loss occurring at MDs is a big challenge. In this work, we aim at DVB-H system which supports the broadcasting service for MDs to discuss the data recovery scheme through the IP-realy WiMAX network, in which the broadcasting lost packets are retransmitted by the on-demand requests. We applied a network coding method to improve the retransmission efficiency and addressed the PNC problem. Such

that, we have proposed the RGS and RPS algorithms to determine the retransmissions sequence of the encoded lost packets in the constrained k time slots, which can support the maximal number of mobile devices to get back their lost packets and serve for the maximum amount of recovery requests whose corresponding deadlines can be fit. Simulation results show that our methods can substantially increase much more overlapped recovery rate compared to no coding scheme and the ISCOD approach broadcasting scheduling method. Our RGS and RPS algorithms can be 1.2~1.5 times of the ISCOD approach algorithms on average. Therefore, considering the timing priority of on-demand recovery requests in network coding scheme can improve the efficiency of data recovery for mobile broadcasting errors.



Chapter 4

Recovery Network Selection in WiMAX and WiFi Networks

4.1 Introduction

Broadband network applications, such as video streaming and VoIP, have recently become popular and have been embraced by many handheld devices. Meanwhile, in many buildings and campuses, WiFi networks have been densely deployed. However, service regions of WiFi networks are still limited due to WiFi's narrow coverage range. To support seamless wireless access, we propose integrating WiFi networks with the emerging WiMAX networks, which are known to have a theoretical data rate of up to 75 Mbps and a transmission range of up to 50 km [29, 52]. WiMAX is considered as a potential candidate to support global broadband services. In this work, we consider an integrated WiMAX and WiFi network, as shown in Fig. 4.1. The WiMAX network serves as a wireless backbone with relay capability and each WiFi *access point (AP)* is connected to a WiMAX *relay station (RS)*. Each *mobile device (MD)* is equipped with dual WiMAX and WiFi interfaces to access both networks, but it will only keep one interface active at a time to save energy.

Under this integrated WiMAX and WiFi network, how to conduct energy-efficient

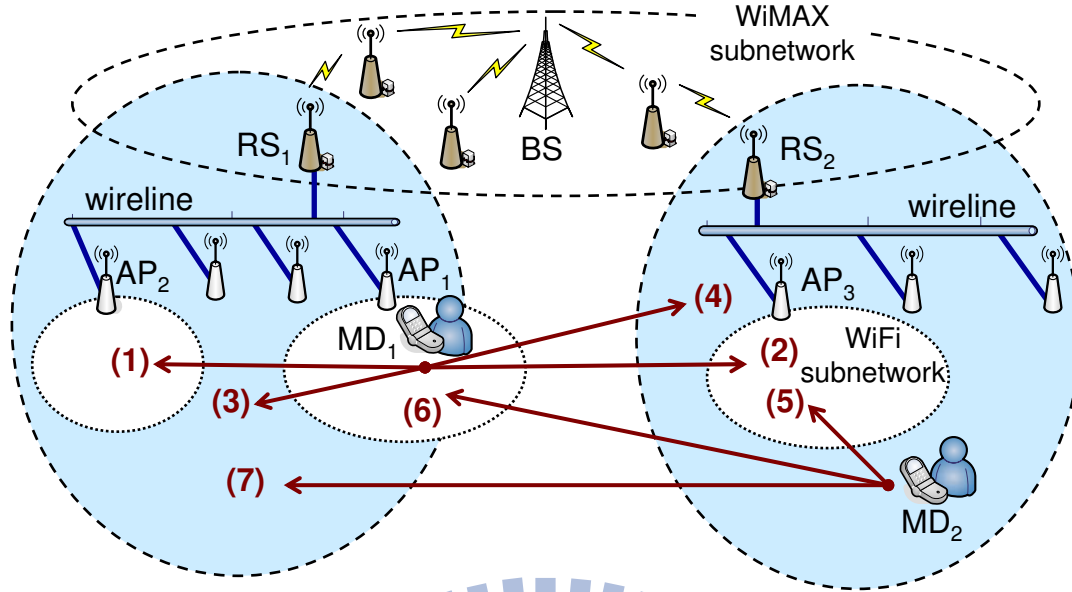


Figure 4.1: An integrated WiMAX and WiFi network.

handovers for battery-powered MDs is a critical issue. Since WiFi networks are often deployed to cover certain indoor and outdoor regions (e.g., rooms and campuses), users' mobility is likely to follow some geographic patterns. Therefore, MDs would also handover in some predictable ways, which we call it as the *geographic mobility (GM) feature* of MDs. References [80, 43, 3] indicate that the GM feature has a significant effect on designing handover schemes. Actually, properly exploiting the GM feature can help conserve MDs' energy due to handover. In this work, we propose a *handover scheme with geographic mobility awareness (HGMA)*, which can conserve MDs' energy from three aspects. Firstly, it prevents MDs from conducting too frequent handovers in a short duration (called *ping-pong effect*). HGMA will measure the moving speeds and *received signal strength (RSS)* of MDs to avoid unnecessary handovers. Secondly, we develop a *handover candidate selection (HCS) method* for MDs to intelligently select a subset of WiFi APs or WiMAX RSs to be scanned, according to MDs' GM features. Thirdly, HGMA prefers MDs staying in their original WiMAX or WiFi networks. This can save MDs' energy by preventing them from switching network interfaces too frequently.

In the literature, research efforts [94, 57, 66, 65, 88] have focused on how to conduct handovers between a low-bandwidth cellular network and high-bandwidth WLANs. They

focus on making handover decisions to obtain a higher bandwidth for MDs, so the results may not be directly applied in our integrated WiMAX and WiFi network because both WiMAX and WiFi are high-bandwidth networks. The work [34] proposes a proactive handover scheme between WLANs and a 3G network. It maintains a handover trigger table that records the locations where APs' signals will drop significantly. This work requires the knowledge of MDs' positions to decide whether it should trigger pre-handoff operations. The handover issue in a heterogeneous wireless network has been widely discussed in [71, 96, 98, 44, 85, 5, 93, 41, 77, 11, 95, 12] by favoring networks with wide coverage. Nevertheless, our handover scheme prefers keeping MDs in their original networks to avoid too frequent interface switching.

Several studies consider the handover issue in an integrated WiMAX and WiFi network. The work [59] avoids unnecessary handovers in an integrated IEEE 802.16a and IEEE 802.11n network by detecting the changes of RSS. It prefers keeping MDs in the IEEE 802.11n network to get a higher bandwidth. Reference [60] proposes a bandwidth-based fuzzy handover scheme in an integrated IEEE 802.16 and IEEE 802.11 network. Reference [28] proposes a general architecture to integrate IEEE 802.11 and IEEE 802.16 networks under the IEEE 802.21 *media independent handover (MIH)* framework to manage the handover procedure. The work [40] combines SIP (session initial protocol) and MIH to manage the handovers between IEEE 802.11e and IEEE 802.16 networks, where SIP is to handle the end-to-end QoS requirements of multimedia sessions. Reference [69] considers using the MIH framework to manage handovers in WiMAX and WiFi networks. The work in [86] adopts MIH to support always-on connectivity services with QoS provision of MDs in integrated WiMAX and WiFi networks. Reference [39] proposes an analytical model based on the economical facet to unravel the handover design tradeoffs between maximizing the service provider profit and satisfying the MDs' performance requirements in hierarchical WiMAX-WiFi networks. As can be seen, these work do not consider the energy issue of MDs. The work [14] considers that WiMAX *base stations (BSs)* will periodically broadcast the density of WiFi APs within their coverage to help MDs to adjust their active scan interval to search for APs. This can help reduce MDs'

energy consumption, but it differs from our work because we focus on reducing scans in both directions and BSs do not need to conduct such broadcasts. The work [13] proposes a *vertical handoff translation center (VHTC)* architecture to improve transmission QoS of MDs. While [13] considers a centralized solution, our HGMA scheme is a distributed one because each AP and RS will help MDs decide their handover targets. Besides, VHTC does not consider the energy issue of MDs. In [50], an architecture of *WiMAX/WiFi access point (W²-AP)* devices is proposed to combine the WiMAX and WiFi technologies. The protocol operation of the WiFi hotspots is the same as that of the WiMAX network, so W²-AP devices can support connection-oriented transmissions and QoS in a similar fashion to the WiMAX network. However, it is not compatible to existing WiFi networks.

The major contributions of this work are three-fold. Firstly, we identify the GM feature of MDs in an integrated WiMAX and WiFi network. Secondly, we propose an energy-efficient HGMA handover scheme by adopting the GM feature. Our HGMA scheme can eliminate unnecessary handovers, reduce the number of network scanning, and avoid switching network interfaces too frequently. Extensive simulation results show that HGMA can conserve about 59% to 80% energy consumption to conduct a handover operation, save about 70% of network scanning, and reduce about 31% of interface switching. Thirdly, HGMA prefers the low-tier WiFi network over the WiMAX network, and guarantees the bandwidth requirements of handovering MDs. It is verified by simulations that HGMA can make MDs to associate with WiFi networks with 16% to 62% more probabilities, and increase 20% to 61% of possibility that WiFi APs can afford the demanded bandwidths of MDs after they handover to.

The rest of this chapter is organized as follows. Section 4.2 presents our system architecture. Section 4.3 proposes our handover schemes. Simulation results are given in Section 4.4. Section 4.5 summarizes this work.

4.2 The Integrated WiMAX and WiFi Network Architecture

The system architecture of our integrated WiMAX and WiFi network has two tiers, as shown in Fig. 4.1. The upper tier is a WiMAX network where each BS is connected to multiple RSs. The lower tier is WiFi networks with lots of APs connected to RSs and then to BSs. We say that an RS and an AP have a *parent-child relationship* if they are connected through a wireline. The wireless coverage of adjacent APs are not necessarily always continuous. We assume that the whole service area is completely covered by the WiMAX network, but may not be completely covered by the WiFi networks, in which both WiMAX and WiFi networks belong to the same service provider. Each MD is equipped with dual 802.11b/g and 802.16e interfaces. However, it tends to keep one interface active at once.

There are seven handover cases in the integrated WiMAX and WiFi network, as numbered by 1–7 in Fig. 4.1:

1. MD₁ handovers from AP₁ to AP₂. AP₁ and AP₂ have the same parent RS₁.
2. MD₁ handovers from AP₁ to AP₃. AP₁ and AP₃ have different parent RSs.
3. MD₁ handovers from AP₁ to its parent RS₁.
4. MD₁ handovers from AP₁ to another RS₂.
5. MD₂ handovers from RS₂ to its child AP₃.
6. MD₂ handovers from RS₂ to another AP₁.
7. MD₂ handovers from RS₂ to RS₁.

In cases 3 and 4, the MD has to switch to the WiMAX mode. In cases 5 and 6, the MD can switch to the WiFi mode to enjoy a higher bandwidth. The above cases can be categorized into two classes. The *HO_{AP}* class contains handover cases 1–4, where an MD moves out of its current AP. The *HO_{RS}* class contains handover cases 5–7, where an MD moves out of its current RS.

Our goal is to design a handover strategy to minimize the energy consumption of MDs such that the ping-pong effect can be alleviated, the number of network scanning can be minimized, the frequency of interface switching can be reduced, and the possibility that MDs associate with APs can be increased. Table 4.1 summarizes our notations used in Chapter 4.

Table 4.1: Notations used in Chapter 4.

notations	definition
e_{MD}	energy level of an MD (0 to γ , empty to fully-charged)
ε_{RS}	energy threshold to switch to the WiFi interface in the HO_{RS} class
V	current speed of an MD
B_{MD}	bandwidth demand of an MD
Δ_{MD}	dwel-time of an MD
$V_0(AP)/V_0(RS)$	average speed of an MD when it is within a WiFi/WiMAX network
$T_{AP}(V)/T_{RS}(V)$	average observation period for an MD to measure RSS in a WiFi/WiMAX network
$S_{AP}^{min}/S_{RS}^{min}$	minimum RSS threshold to trigger a handover in the HO_{AP}/HO_{RS} class
S_{AP}/S_{RS}	RSS threshold to check if trigger a handover in the HO_{AP}/HO_{RS} class
n_{exp}	expected number of candidate APs/RSs
$\mathcal{L}_{AP}(RS_i)$	table in an MD that records the neighboring APs of the associating RS_i
$\mathcal{C}_{AP}/\mathcal{C}_{RS}$	candidate APs/RSs returned by HCS
$\mathcal{T}_{AP}(A_i)/\mathcal{T}_{RS}(A_i)$	table that records neighboring APs/RSs of an AP A_i
$\mathcal{T}_{AP}(R_i)/\mathcal{T}_{RS}(R_i)$	table that records neighboring APs/RSs of an RS R_i

4.3 The HGMA Scheme

Fig. 4.2 shows the flowchart of our HGMA scheme. It has two parts, namely HO_{AP} and HO_{RS} classes.

HO_{AP} class: When an MD is currently associated with an AP_i , it periodically checks

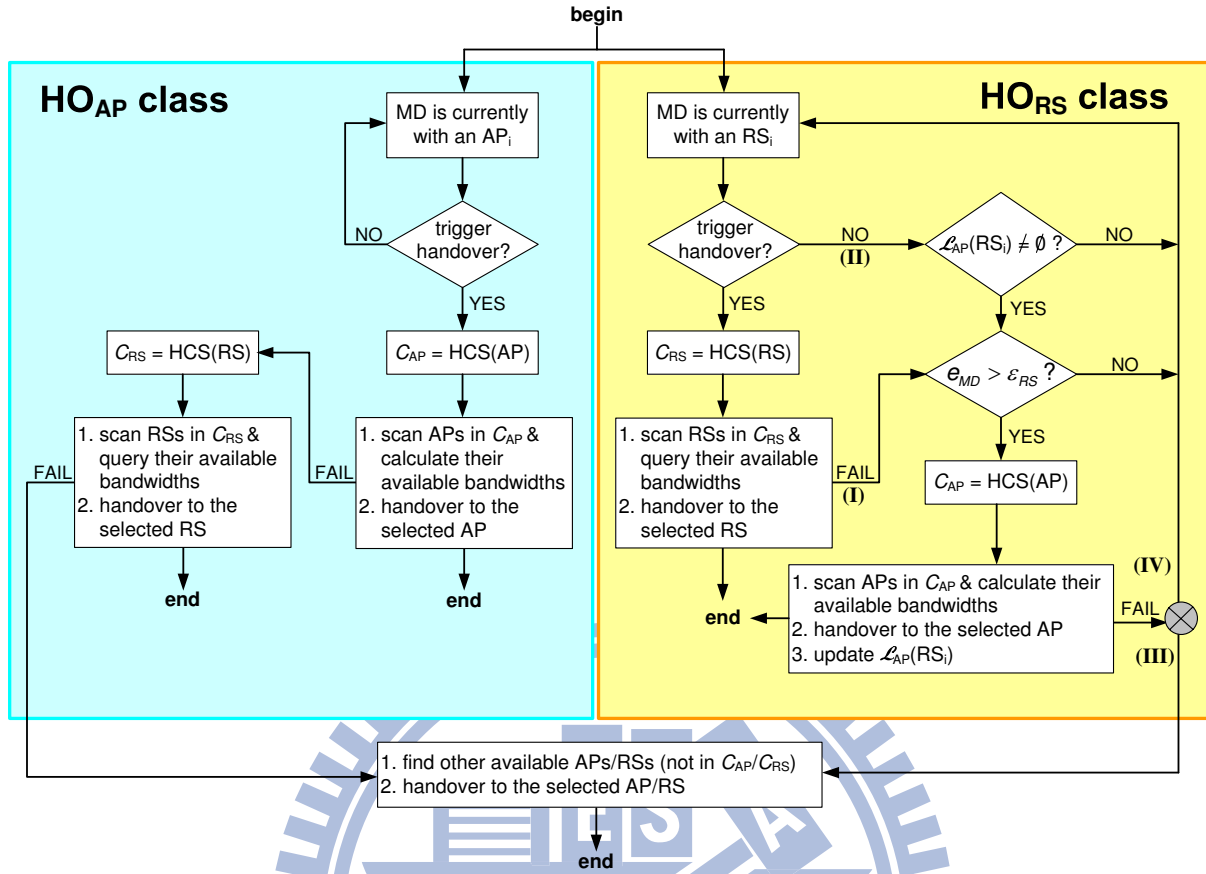


Figure 4.2: The HGMA scheme.

whether it should trigger a handover. The checking procedure will be discussed in Section 4.3.1. If triggering a handover is needed, it will invoke the HCS method in Section 4.3.2 to select a set of candidate APs, called C_{AP} . The MD then scans those APs in C_{AP} and calculates each AP's available bandwidth (for example, the results in [58, 89] point out some ways to calculate an AP's remaining bandwidth). Then, among those APs whose available bandwidths can afford the MD's traffic demand B_{MD} , the MD will select the AP with the strongest RSS to handover to. However, if the MD fails to find an available AP in C_{AP} , it will invoke the HCS method, which will return a set of candidate RSs, call C_{RS} . Then, it scans these RSs and queries their available bandwidths by a request-grant protocol. Among those RSs whose available bandwidths can afford the bandwidth demand B_{MD} , the MD will select the RS with the strongest RSS to handover to. However, if no feasible RS can be found, the MD will search for other APs/RSs (not in C_{AP}/C_{RS}).

HO_{RS} class: When an MD is currently associated with an RS_i , it also periodically checks whether it should trigger a handover. If needed, the MD will invoke the HCS method and then scan the candidate RSs in \mathcal{C}_{RS} . If this step fails (case I in Fig. 4.2), it will check whether its current energy level e_{MD} is larger than a predefined threshold ε_{RS} . If so, the MD will invoke the HCS method, turn on its WiFi interface, scan the candidate APs in \mathcal{C}_{AP} , find one target AP, and add the selected AP into its $\mathcal{L}_{AP}(RS_i)$, where $\mathcal{L}_{AP}(RS_i)$ is a table that records all APs to which the MD has ever handed over from its current RS_i . In the above discussion, even if the MD decides not to trigger handover (see case II in Fig. 4.2), it will still try to search for APs if both $\mathcal{L}_{AP}(RS_i) \neq \emptyset$ (i.e., there are potential APs in this RS_i 's neighborhood) and $e_{MD} > \varepsilon_{RS}$ (i.e., the MD has sufficient energy to conduct extra network scanning). In this case, the MD will invoke the HCS method, scan the candidate APs in \mathcal{C}_{AP} , find one target AP, and add the selected AP into its $\mathcal{L}_{AP}(RS_i)$. However, when the MD cannot find any feasible RS/AP, two cases will happen (refer to the switch point ‘ \otimes ’ in Fig. 4.2). If the MD arrives at \otimes from case I, it will follow case III and search for other APs/RSs not in $\mathcal{C}_{AP}/\mathcal{C}_{RS}$. However, if the MD arrives at \otimes from case II, it will follow case IV and keep associating with its current RS. We remark that one possible way to define ε_{RS} is to set it as a fixed value such as $\lfloor \frac{\gamma}{2} \rfloor$, where $\gamma \in \mathbb{N}$ is the maximum energy level of an MD when it is fully-charged. Another way is to adaptively change ε_{RS} according to the MD's current speed V . For example, let $V_0(RS)$ be the average speed of the MD when it is within WiMAX networks. We can update $\varepsilon_{RS} = \min\{1, \frac{V}{V_0(RS)}\} \times \varepsilon_{RS}^D$, where ε_{RS}^D is the default value of ε_{RS} . Specifically, when $V < V_0(RS)$, a smaller ε_{RS} is set since the MD has more opportunity to stay in its originally serving RS. In this case, we can encourage the MD to search for neighboring APs to handover to.

To summarize, our HGMA scheme has several special designs to conserve MDs' energy. Firstly, to prevent MDs from conducting unnecessary handovers due to the ping-pong effect, we propose a handover-triggering process in Section 4.3.1, which is based on the RSSs and moving speeds of MDs. Secondly, to reduce the number of network scanning, we design the HCS method in Section 4.3.2, which is based on the GM feature of MDs

to select only a subset of neighboring APs/RSs to be scanned. Thirdly, to reduce the frequency of interface switching, we make an MD in the HO_{AP} class to scan available APs first, and then scan RSs when no AP can be found. The similar design is also applied to the HO_{RS} class. Last, to increase the probability that MDs stay in WiFi networks, we search for APs first in the HO_{AP} class when a handover event is triggered. Also, in the HO_{RS} class, an MD can still try to search for APs if its energy level is sufficient.

4.3.1 When to Trigger a Handover

The process to trigger a handover event is shown in Fig. 4.3. Here, we assume that an MD can measure its current speed V . In the HO_{AP} class (refer to Fig. 4.3(a)), an MD will periodically measure the average RSS S from its associating AP_i . Typically, a lower RSS will trigger an MD to start a handover process. However, to alleviate the ping-pong effect when an MD is around cell boundaries due to temporal RSS dropping, we propose adaptively adjusting an *observation interval* as follows:

$$T_{AP}(V) = \frac{V_0(AP)}{V} \times T_0(AP),$$

where $V_0(AP)$ is the average speed of the MD when it is within WiFi networks and $T_0(AP)$ is a constant representing the MD's average observation interval. If its average RSS is continuously below the handover threshold $S_{AP}^{\min 1}$ over the interval $T_{AP}(V)$, it means that the MD is very close to the coverage boundary of that AP and thus the MD should trigger a handover event. Otherwise, we check whether S is continuously below a second threshold $S_{AP}^{\min 2}$, where $S_{AP}^{\min 2} > S_{AP}^{\min 1}$. If so, it means that the MD is *near* the coverage boundary of AP_i and we will further check whether the MD's current speed $V > V_0(AP)$. If the speed criterion is met, the MD will also trigger a handover event because it is likely to move out of its current AP_i .

Similarly, in the HO_{RS} class (refer to Fig. 4.3(b)), an MD will continuously measure the average RSS from its associating RS_j for an observation interval:

$$T_{RS}(V) = \frac{V_0(RS)}{V} \times T_0(RS),$$

where $T_0(RS)$ is the average observation interval to measure RSS in the WiMAX network. When the RSS S is continuously below the minimum threshold $S_{RS}^{\min 1}$, a handover event should be triggered. Otherwise, if both $S < S_{RS}^{\min 2}$ and $V > V_0(RS)$, the MD should also trigger a handover event, where $S_{RS}^{\min 2} > S_{RS}^{\min 1}$ is a secondary threshold.

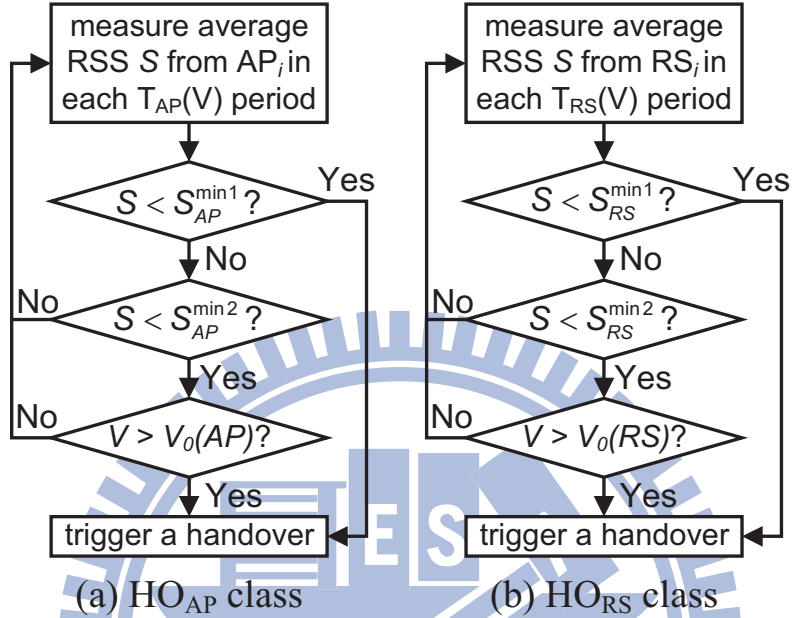


Figure 4.3: Checking procedures to trigger a handover event.

4.3.2 The HCS Method

HCS exploits the GM feature based on past handover patterns of MDs to select candidate APs/RSs to be scanned. Specifically, for each AP A_i , we maintain two tables $\mathcal{T}_{AP}(A_i)$ and $\mathcal{T}_{RS}(A_i)$ to record the information of A_i 's neighboring APs and RSs, respectively. An AP A_j is in A_i 's $\mathcal{T}_{AP}(A_i)$ if A_i and A_j share the same parent RS or an MD has ever handovered from A_i to A_j . An RS is in A_i 's $\mathcal{T}_{RS}(A_i)$ if it is A_i 's parent RS or an MD has ever handovered from A_i to this RS. Similarly, for each RS R_i , we maintain its neighboring RSs and APs in tables $\mathcal{T}_{RS}(R_i)$ and $\mathcal{T}_{AP}(R_i)$, respectively. An RS R_j is in R_i 's $\mathcal{T}_{RS}(R_i)$ if an MD has ever handovered from R_i to R_j . An AP is in R_i 's $\mathcal{T}_{AP}(R_i)$ if it is a child AP of R_i or an MD has ever handovered from R_i to this AP.

In the above four tables, three extra pieces of information are maintained. Without

loss of generality, we consider table $\mathcal{T}_{\text{AP}}(A_i)$ of an AP A_i . For each A_j in $\mathcal{T}_{\text{AP}}(A_i)$, we record three statistics over a time interval: $v(A_i, A_j)$ counts the total number of MDs that have ever handovered from A_i to A_j , $t(A_i, A_j)$ records the average dwell-time in A_i for these MDs, and $b(A_i, A_j)$ is the average available bandwidth provided by A_j when these MDs handovered from A_i . The other three tables $\mathcal{T}_{\text{RS}}(A_i)$, $\mathcal{T}_{\text{AP}}(R_i)$, and $\mathcal{T}_{\text{RS}}(R_i)$ also maintain the similar information.

HCS works as follows. Without loss of generality, we consider an MD that is currently associated with AP A_i and intends to scan other APs. HCS will select a number $n = \min\{n_{\text{exp}}, |\mathcal{T}_{\text{AP}}(A_i)|\}$ of candidate APs from its $\mathcal{T}_{\text{AP}}(A_i)$ to be scanned, where n_{exp} is the expected number of candidate APs and $|\mathcal{T}_{\text{AP}}(A_i)|$ is the total number of APs in $\mathcal{T}_{\text{AP}}(A_i)$. One way to define n_{exp} is to consider the MD's remaining energy. When the MD has more energy, a larger n_{exp} can be used. For example, supposing that the MD remains l percent of energy, we can set its $n_{\text{exp}} = \lfloor l \times \gamma \rfloor$. To select these n APs, we calculate the *similarity* $s(A_i, A_j)$ of A_i and A_j for each $A_j \in \mathcal{T}_{\text{AP}}(A_i)$ as follows:

$$s(A_i, A_j) = \sqrt{W_v \times f_v(A_i, A_j)^2 + W_t \times f_t(A_i, A_j)^2 + W_b \times f_b(A_i, A_j)^2}, \quad (4.1)$$

where W_v , W_t , and W_b are weights such that $W_v + W_t + W_b = 1$ and

$$f_v(A_i, A_j) = \frac{v(A_i, A_j)}{\sum_{A_k \in \mathcal{T}_{\text{AP}}(A_i)} v(A_i, A_k)}, \quad (4.2)$$

$$f_t(A_i, A_j) = \frac{\min\{\Delta_{\text{MD}}, t(A_i, A_j)\}}{\max\{\Delta_{\text{MD}}, t(A_i, A_j)\}}, \quad (4.3)$$

$$f_b(A_i, A_j) = \min\left\{1, \frac{b(A_i, A_j)}{B_{\text{MD}}}\right\}, \quad (4.4)$$

where Δ_{MD} is the current dwell-time that the MD has stayed in A_i and B_{MD} is the MD's bandwidth demand. Then, HCS selects the first n most similar APs in $\mathcal{T}_{\text{AP}}(A_i)$ and stores them in \mathcal{C}_{AP} .

Eq. (4.1) is to measure the similarity between A_i and A_j . Eqs. (4.2)–(4.4) are three factors to be considered and their values should be bounded by $0 \leq f_v(A_i, A_j), f_t(A_i, A_j), f_b(A_i, A_j) \leq 1$. Eq. (4.2) represents the *visiting frequency* $f_v(A_i, A_j)$ from A_i to A_j in the past handover

statistics. Eq. (4.3) represents the *dwelt-time relationship* $f_t(A_i, A_j)$. Recall that $t(A_i, A_j)$ is the average dwelt-time before handovering. When the MD's dwelt-time Δ_{MD} is closer to $t(A_i, A_j)$, there is a higher probability that the MD will take a handover. Eq. (4.4) represents the *expected available bandwidth* $f_b(A_i, A_j)$ of A_j . Since $b(A_i, A_j)$ records the past available bandwidth of A_j , we thus prefer those with a larger available bandwidth. However, this value is upper bounded by 1. These factors are given weights W_v , W_t , and W_b . To determine their values, we can consider the moving speed of an MD. For example, we can set

$$W_v = \begin{cases} \frac{v}{V+V_0(AP)} & \text{in HO}_{\text{AP}} \text{ class} \\ \frac{v}{V+V_0(RS)} & \text{in HO}_{\text{RS}} \text{ class,} \end{cases}$$

$$W_t = W_b = \begin{cases} \frac{V_0(AP)}{2 \times [V+V_0(AP)]} & \text{in HO}_{\text{AP}} \text{ class} \\ \frac{V_0(RS)}{2 \times [V+V_0(RS)]} & \text{in HO}_{\text{RS}} \text{ class.} \end{cases}$$

4.4 Experimental Results

In this section, we present extensive simulation results to verify the effectiveness of the proposed HGMA scheme. Our simulations are conducted by the IEEE 802.16 modules based on ns-2 [2]. The physical layer adopts an OFDM (orthogonal frequency division multiplexing) module and the radio propagation model is set to two-ray ground. We consider two types of network topologies. In the *dense* topology, each RS has 10 child APs. In the *random* topology, each RS has arbitrarily 0 to 10 child APs. Each MD will consume 5% of its energy every five minutes. We rank an MD's remaining energy by levels. When fully charged, its energy level is γ , where $\gamma \in \mathbb{N}$. The calculation of energy level is as follows:

$$\left\lfloor \frac{\text{current remaining energy}}{\text{fully-charged energy}} \times \gamma \right\rfloor.$$

We set $\varepsilon_{\text{RS}} = 1$ and $W_v = W_t = W_b = \frac{1}{3}$.

The BonnMotion tool [1] is adopted to generate three types of mobility models for MDs. In the *random waypoint (WAYPOINT) model*, as shown in Fig. 4.4(a), an MD

randomly chooses one destination (e.g., d_2) to move to, with an average speed of $[0, 1]$ m/s. After reaching its destination, the MD pauses about 120 seconds and then selects another destination (e.g., d_3) to move to. In the *reference point group mobility (GROUP) model*, each MD belongs to one group and the same group of MDs will move in the same direction and have the same speed. The speed is generated randomly from $[0.5, 1.5]$ m/s and the pause time is upper bounded by 60 seconds. An MD may leave its group and join another group with a probability of 0.01. Fig. 4.4(b) gives an example, where MD₁, MD₂ and MD₃ belong to the same group j . Although MD₁ wants to move to another destination based on its own moving path (the dotted circle), it adapts its direction by following the indication of group j , which depends on the mobility behavior of the reference point MD₃. Thus, with the same speed and direction, MD₁, MD₂ and MD₃ will move simultaneously. In the *Manhattan grid (GRID) model*, MDs move on a number of horizontal and vertical streets in an urban area. The speeds of MDs range from 0.5 to 1 m/s, with a maximum pause time of 120 seconds. Each MD may change its direction when it reaches an intersection, with a turning probability of 0.5. Fig. 4.4(c) gives an example, where the MD moves along the path of $d_1 \rightarrow d_2 \rightarrow d_3 \rightarrow d_4 \rightarrow d_5$. In our simulations, we set $V_0(AP) = 0.5$ m/s, $V_0(RS) = 1$ m/s, and $T_0(AP) = T_0(RS) = 100 \mu\text{s}$. We mainly compare our HGMA scheme with the traditional handover scheme, where a handovering MD will scan all APs/RSs around it.

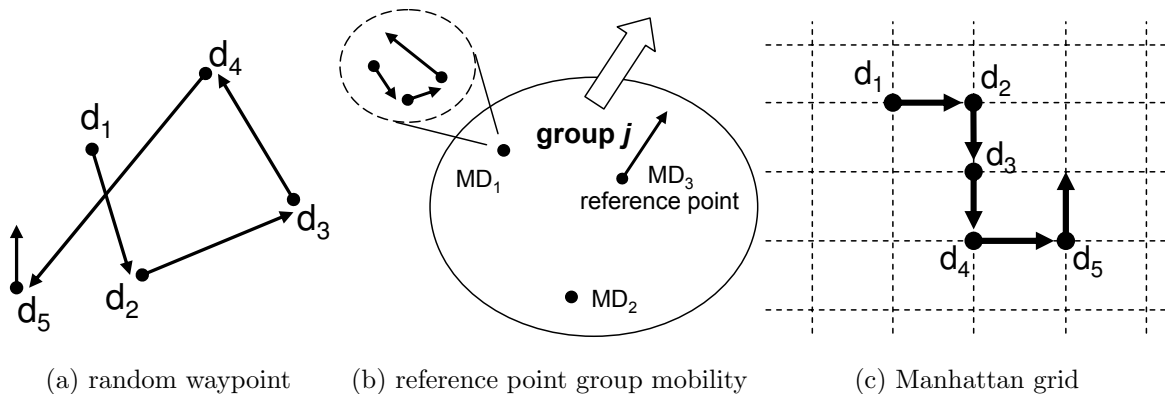


Figure 4.4: Scenarios of three mobility models.

4.4.1 Number of Network Scans

Fig. 4.5 shows the average number of APs scanned by MDs. Clearly, the average number of APs scanned by MDs in the dense topology is larger than that in the random topology because the former has more APs. The traditional handover scheme will ask MDs to scan all possible APs around them, even though they have lower remaining energy. On the contrary, HGMA allows MDs to scan fewer APs when they have lower energy. In this way, the energy of MDs can be conserved. When comparing these three mobility models, we can observe that HGMA will cause more scans in the WAYPOINT model than those in the GROUP and GRID models when there are more than 45% of remaining energy. This is because the WAYPOINT model has a less regular mobility pattern, causing lower predictability, and a higher n_{exp} value is used. With the GROUP and GRID models, the candidate APs/RSs are more predictable. This result supports that our HGMA scheme is capable of decreasing the number of network scans for MDs if they move in some regular patterns. Note that in the WAYPOINT model, the number of APs scanned by MDs plunges when the remaining energy of MDs is within [40%, 35%]. This is because less handovers are incurred by MDs in the simulation.

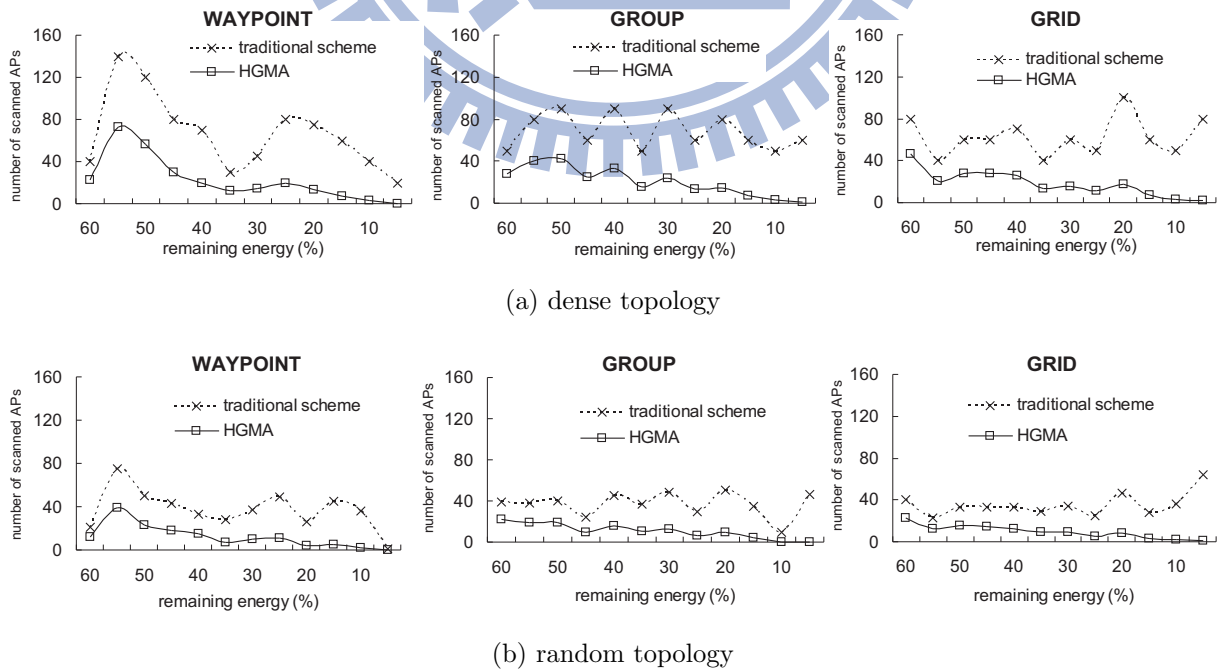


Figure 4.5: Average number of scanned APs in different models.

Table 4.2 summarizes the average reduction of the number of scanned APs by our HGMA scheme. HGMA can reduce about 70% of AP scanning, thereby significantly reducing the energy consumption of MDs. Additionally, HGMA works the best when MDs move in the GRID model, especially in the random topology. This is because the GRID model has the most regular mobility pattern for MDs, making the prediction more accurate. On the other hand, even in the non-regular WAYPOINT model, HGMA can still save about 66% to 67% of the scanning events. This clearly shows the advantages of HGMA.

Table 4.2: Reduction of the number of scanned APs by HGMA.

topology	WAYPOINT	GROUP	GRID	average
dense	66.13%	70.12%	70.80%	68.99%
random	67.12%	70.81%	73.11%	70.31%

4.4.2 Cumulative Number of Interface Switching

Fig. 4.6 shows the cumulative number of interface switching of MDs. When MDs have more than 40% of remaining energy, HGMA has almost no effect on interface switching. This is because it encourages those MDs associating with RSs to handover to WiFi networks when they have more energy. However, when MDs remain less energy, HGMA will prevent them from frequently switching network interfaces to save their energy.

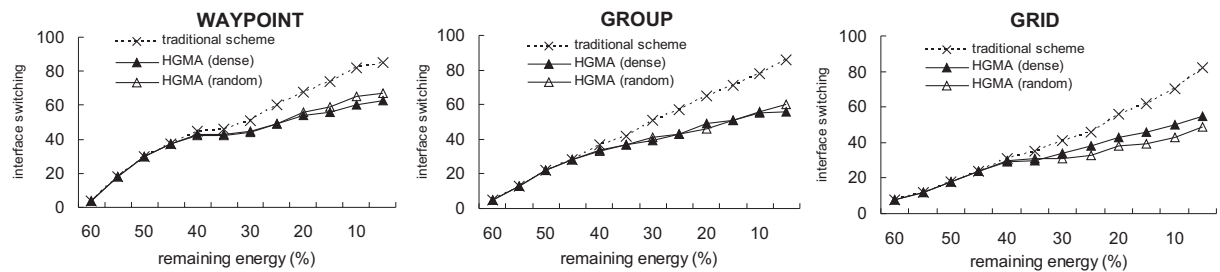


Figure 4.6: Cumulative number of interface switching in different models.

Table 4.3 summarizes the average reduction of the number of interface switching by HGMA. HGMA can reduce about 31% of interface switching, thus significantly conserving

the energy of MDs. In addition, HGMA works better when MDs move in the more regular GROUP and GRID models. On the other hand, even in the non-regular WAYPOINT model, HGMA can still reduce about 21% to 26% of the switching events. This verifies the effectiveness of HGMA.

Table 4.3: Reduction of the number of interface switching by HGMA.

topology	WAYPOINT	GROUP	GRID	average
dense	25.88%	34.88%	32.93%	31.23%
random	21.18%	30.23%	40.24%	30.55%

4.4.3 Cumulative Number of Handovers to APs

Fig. 4.7 shows the cumulative number of handovers to APs. When trying to associate with a WiFi AP, we simulate an *association failure probability* $p = 0.2 \sim 0.4$ (this is to take factors such as contention into account). When p increases, the number of handovers to APs will decrease. From Fig. 4.7(a), we can observe that HGMA can still perform well even with a larger p . This is because the AP density is large enough and thus MDs can easily find an AP to handover to. On the other hand, in Fig. 4.7(b), the improvement of HGMA is less significant when p is larger. This is because MDs may not always find suitable APs in their neighborhood. However, when MDs move in the more regular GROUP and GRID models, HGMA can increase the possibility that MDs associate with WiFi networks.

Table 4.4 gives the improvement of the number of handovers to APs by HGMA. On average, HGMA can increase 62% and 16% of probability for MDs to associate with WiFi networks when p is set to 0.2 and 0.4, respectively.

4.4.4 Dwell-time in WiFi Networks

Fig. 4.8 illustrates the average dwell-time that MDs stay in WiFi networks during one hour. When the failure probability p increases, the dwell-time will reduce because MDs

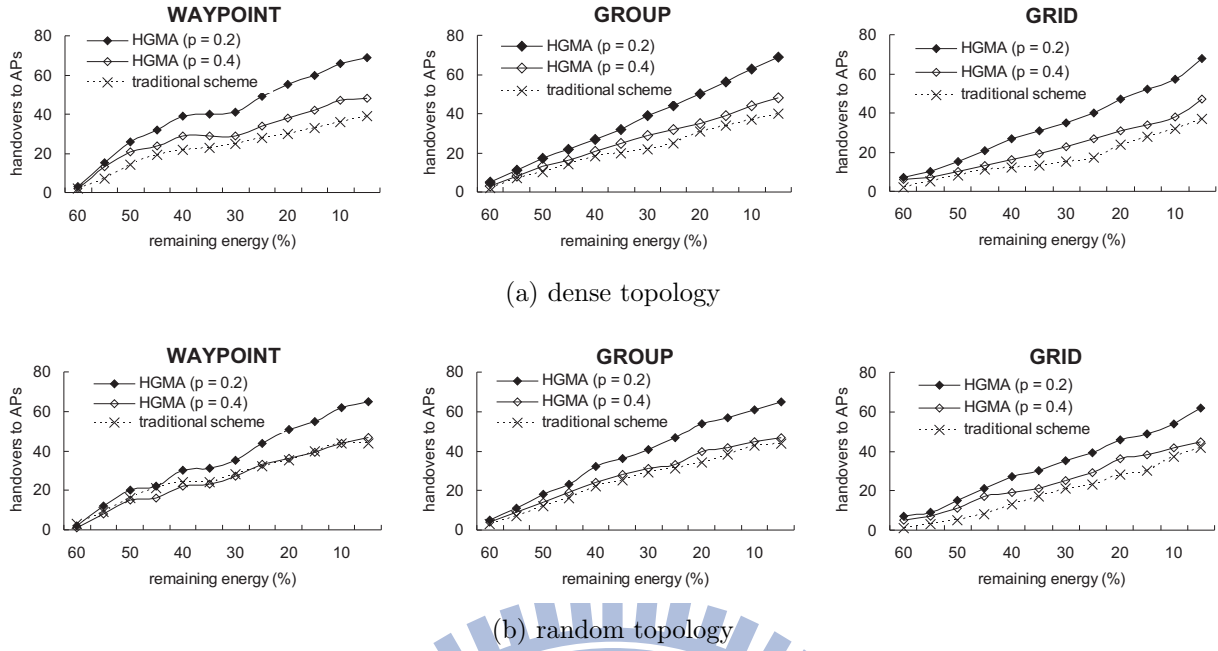


Figure 4.7: Cumulative number of handovers to APs in different models.

Table 4.4: Improvement of the number of handovers to APs by HGMA.

p	topology	WAYPOINT	GROUP	GRID	average
0.2	dense	1.769	1.673	1.838	1.760
	random	1.477	1.480	1.476	1.478
0.4	dense	1.231	1.203	1.270	1.235
	random	1.068	1.105	1.071	1.081

may fail in the association more often. When MDs move in a more regular pattern, they can stay in WiFi networks for a longer time when our HGMA scheme is adopted. From Fig. 4.8, we can observe that HGMA can keep MDs in WiFi networks more than half an hour in the GROUP and GRID models when $p = 0.2$.

Table 4.5 gives the improvement on average dwell-time in WiFi networks by HGMA. On average, HGMA can increase 132% and 76% of dwell-time for MDs when the p is set to 0.2 and 0.4, respectively.

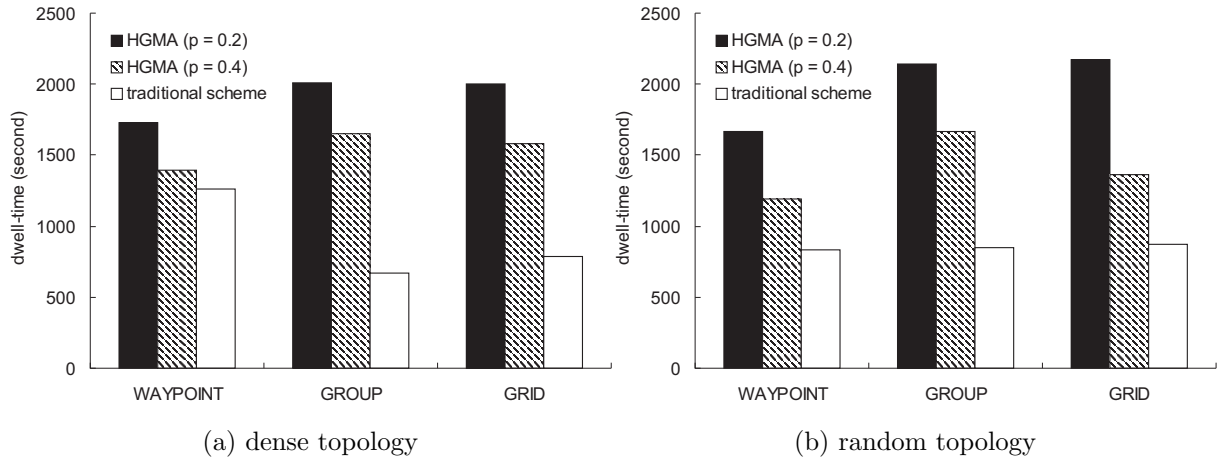


Figure 4.8: Average dwell-time of MDs in WiFi networks during one hour in different models.

Table 4.5: Improvement of average dwell-time of MDs in WiFi networks by HGMA.

p	topology	WAYPOINT	GROUP	GRID	average
0.2	dense	1.372	2.990	2.548	2.303
	random	2.002	2.526	2.497	2.342
0.4	dense	1.106	2.458	2.020	1.861
	random	1.430	1.967	1.568	1.655

4.4.5 Energy Consumption

Next, we compare the energy consumption of MDs by HGMA and by the traditional handover scheme. We set $p = 0 \sim 0.4$ and the energy cost for an MD to scan an AP/RS to 1/1.1 unit of energy. Also, the cost to switch network interface is 1 unit of energy. Fig. 4.9 shows the average energy consumption of MDs to conduct a handover operation. We can observe that when the traditional handover scheme is adopted, MDs will consume much energy even if they remain less energy. This is due to two reasons. Firstly, the traditional handover scheme will make an MD scan all neighboring APs/RSs to determine its handover target. From Fig. 4.5, we can observe that the number of scanned APs by the traditional handover scheme is still large even though MDs remain less energy. Secondly,

the traditional handover scheme will make MDs frequently switch their network interfaces, no matter how much energy they remain (such effect can be found in Fig. 4.6). Thus, the energy consumption of MDs to conduct a handover operation by the traditional handover scheme may increase when MDs' energy decreases. On the contrary, our HGMA scheme can make MDs spend less energy when conducting handovers. Again, from Figs. 4.5 and 4.6, HGMA will reduce the numbers of AP scanning and interface switching when MDs remain less energy. Thus, the energy consumption of MDs to conduct a handover operation by HGMA decreases when MDs' energy decreases. Such phenomenon is more significant when MDs have no more than 30% of remaining energy. When $p = 0$, the energy consumption of MDs is quite less (especially when the remaining energy is lower than 30%), because MDs can easily find an AP to associate with.

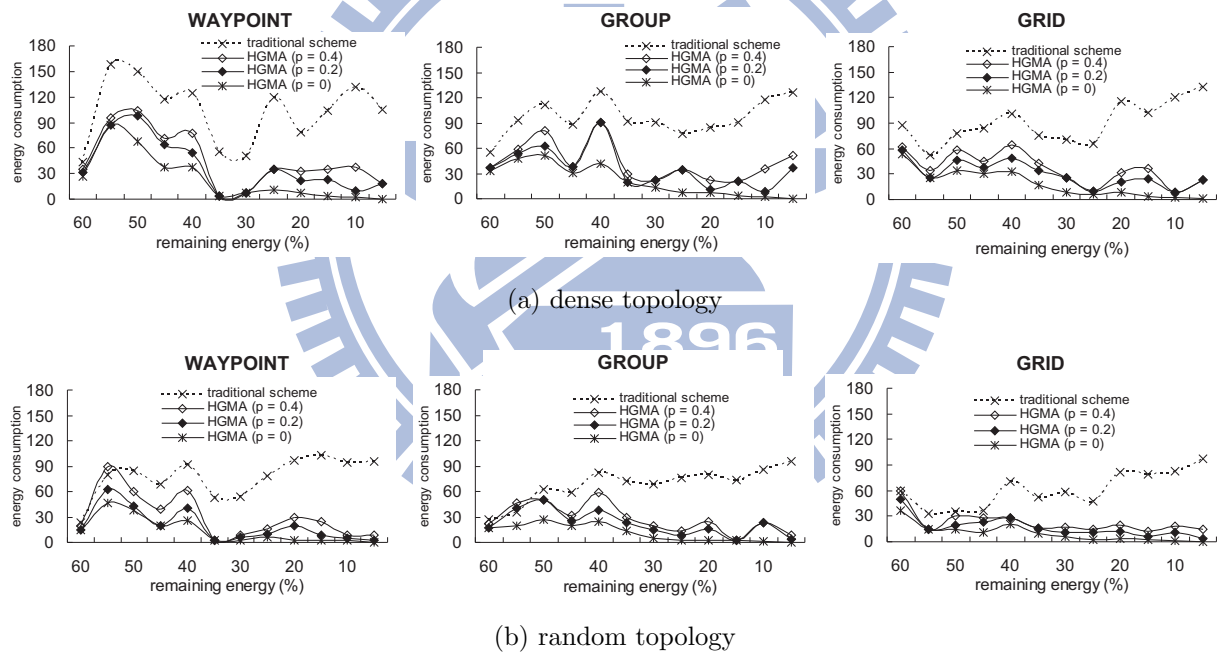


Figure 4.9: Average energy consumption of MDs to conduct a handover operation in different models.

Table 4.6 summarizes the average reduction of energy consumption of MDs to conduct a handover operation by HGMA. Clearly, HGMA can reduce the most energy consumption in the GRID model due to MDs' predictable mobility. On average, HGMA can reduce about 80%, 68%, and 59% of energy consumption of handovering MDs when p is set to

0, 0.2, and 0.4, respectively.

Table 4.6: Reduction of energy consumption of MDs to conduct a handover operation by HGMA.

p	topology	WAYPOINT	GROUP	GRID	average
0	dense	76.37%	77.66%	79.17%	77.73%
	random	82.10%	83.50%	83.06%	82.89%
0.2	dense	63.47%	62.34%	66.54%	64.13%
	random	74.40%	67.73%	71.86%	71.33%
0.4	dense	55.48%	54.98%	59.26%	56.57%
	random	60.30%	59.66%	62.57%	60.84%

4.4.6 QoS Measurement

Last, we measure the *QoS-satisfied handover ratio* by HGMA and by the traditional handover scheme during one hour, which is defined by the ratio of the number of handovers to WiFi networks that can satisfy the demanded bandwidths of handovering MDs to the total number of handovers to WiFi networks. Fig. 4.10 illustrates the QoS-satisfied handover ratio in different mobility models. We can observe that HGMA outperforms the traditional handover scheme because HCS considers the bandwidth factor (i.e., Eq. (4.4)) when selecting candidate APs. When p increases, the QoS-satisfied handover ratio of HGMA decreases because MDs may not easily find suitable APs in their neighborhood.

Table 4.7 shows the improvement of QoS-satisfied handover ratios by HGMA. We can observe that HGMA works better in more regular GROUP and GRID models. On average, HGMA can improve about 61% and 20% of ratio when p is set to 0.2 and 0.4, respectively. This indicates that HGMA can guarantee QoS requirements of handovering MDs.

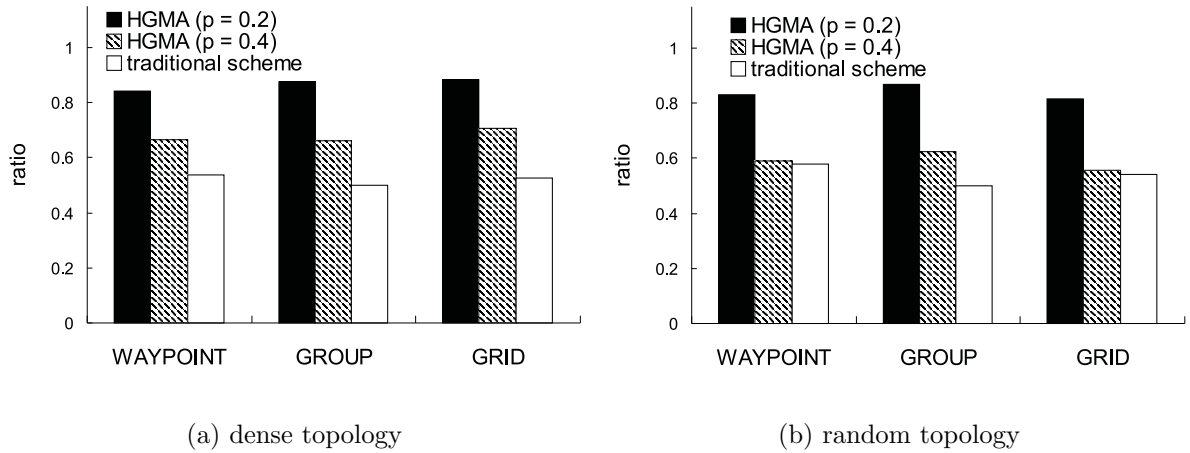


Figure 4.10: QoS-satisfied handover ratios in different models.

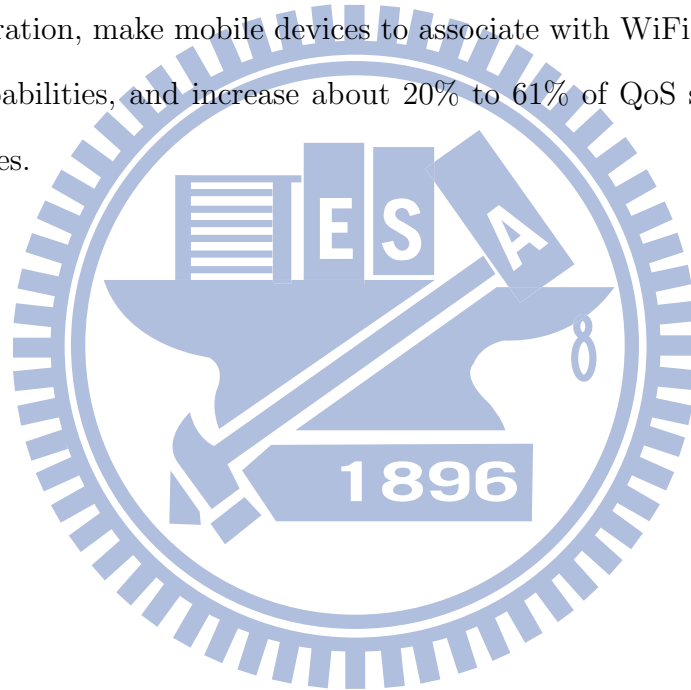
Table 4.7: Improvement of QoS-satisfied handover ratios by HGMA.

p	topology	WAYPOINT	GROUP	GRID	average
0.2	dense	1.569	1.75	1.681	1.667
	random	1.435	1.736	1.501	1.557
0.4	dense	1.242	1.321	1.345	1.303
	random	1.017	1.245	1.023	1.095

4.5 Summary of Chapter 4

To provide wireless Internet access, WiFi networks have been deployed in many regions such as buildings and campuses. However, WiFi networks are still insufficient to support ubiquitous wireless service due to their narrow coverage. Specifically, when a mobile device want to recover some data losses on another broadcast network through accessing the wireless Internet channel. One possibility to resolve this deficiency is to integrate WiFi networks with the wide-range WiMAX networks. Under such an integrated WiMAX and WiFi network, how to conduct energy-efficient handovers is a critical issue. In this work, we propose a *handover scheme with geographic mobility awareness (HGMA)*, which considers the historical handover patterns of mobile devices. HGMA can conserve the energy of handovering devices from three aspects. Firstly, it prevents mobile devices from

triggering unnecessary handovers according to their received signal strength and moving speeds. Secondly, it contains a *handover candidate selection (HCS) method* for mobile devices to intelligently select a subset of WiFi access points or WiMAX relay stations to be scanned. Therefore, mobile devices can reduce their network scanning and thus save their energy. Thirdly, HGMA prefers mobile devices staying in their original WiMAX or WiFi networks. This can prevent mobile devices from consuming too much energy on interface switching. In addition, HGMA prefers the low-tier WiFi network over the WiMAX network and guarantees the bandwidth requirements of handovering devices. Simulation results show that HGMA can save about 59% to 80% of energy consumption of a handover operation, make mobile devices to associate with WiFi networks with 16% to 62% more probabilities, and increase about 20% to 61% of QoS satisfaction ratio to handovering devices.



Chapter 5

Conclusion and Future Work

This dissertation contains three works for the data recovery in mobile broadcast networks. The first work is the message-efficient data recovery for DVB-H data losses through an IP-relay network. The second work is the on-demand data recovery by network coding for mobile broadcasting systems. The third work is the energy-efficient network selection in an integrated WiMAX and WiFi network.

In Chapter 2, we follow the DVB-IPDC architecture by integrating a DVB-H system with a broadband WiMAX network to support the packet recovery mechanism. We have addressed two critical GPL and BDH problems and developed a novel BR-LW scheme to solve these problems. BR-LW exploits the spatial and temporal correlation of recovery requests, and efficiently reduces duplicate request submissions while merging retransmissions of lost packets. Simulation results have verified that BR-LW significantly reduces the amount of message transmissions inside a WiMAX RS cell, thereby alleviating network congestion while improving communication efficiency. Furthermore, with mathematical analysis, we have discussed how to adaptively determine the timing for each MD to send its requests based on the channel condition, as well as the timing for each RS to broadcast a group acknowledgement according to the spatial and temporal correlation of received requests.

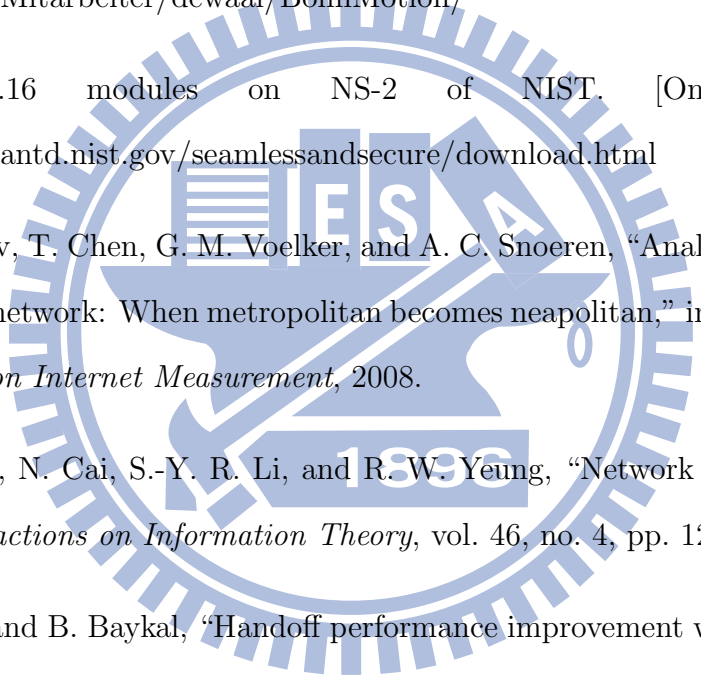
In Chapter 3, we aim at DVB-H system which supports the broadcasting service for

MDs to discuss the data recovery scheme through the IP-realy WiMAX network, in which the broadcasting lost packets are retransmitted by the on-demand requests. We applied a network coding method to improve the retransmission efficiency and addressed the PNC problem. Such that, we have proposed the RGS and RPS algorithms to determine the retransmissions sequence of the encoded lost packets in the constrained k time slots, which can support the maximal number of mobile devices to get back their lost packets and serve for the maximum amount of recovery requests whose corresponding deadlines can be fit. Simulation results show that our methods can substantially increase much more overlapped recovery rate compared to no coding scheme and the ISCOD approach broadcasting scheduling method. Our RGS and RPS algorithms can be 1.2~1.5 times of the ISCOD approach algorithms on average. Therefore, considering the timing priority of on-demand recovery requests in network coding scheme can improve the efficiency of data recovery for mobile broadcasting errors.

In Chapter 4, we have defined an integrated WiMAX and WiFi network architecture and discussed the corresponding handover scenarios. We have proposed an energy-efficient HGMA scheme that considers the GM feature of MDs. By eliminating unnecessary handovers, reducing the number of network scanning, and avoiding too frequent interface switching, the proposed HGMA scheme can significantly conserve the energy of MDs due to handover. Simulation results have justified the efficiency and improvement of the proposed scheme in various mobility models.

For the future work, we will further investigate how to improve video quality of MDs by considering some QoS requirements such as delay, jitter, and PSNR (peak signal to noise ratio) in the proposed BR-LW scheme. Besides, we will analyze the time complexity of the proposed RGS and RPS algorithms.

Bibliography

- 
- [1] BonnMotion: A mobility scenario generation and analysis tool by University of Bonn. [Online]. Available: <http://web.informatik.uni-bonn.de/IV/Mitarbeiter/dewaal/BonnMotion/>
- [2] IEEE 802.16 modules on NS-2 of NIST. [Online]. Available: <http://www.antd.nist.gov/seamlessandsecure/download.html>
- [3] M. Afanasyev, T. Chen, G. M. Voelker, and A. C. Snoeren, “Analysis of a mixed-use urban WiFi network: When metropolitan becomes neapolitan,” in *ACM SIGCOMM Conference on Internet Measurement*, 2008.
- [4] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, “Network information flow,” *IEEE Transactions on Information Theory*, vol. 46, no. 4, pp. 1204–1216, 2000.
- [5] O. B. Akan and B. Baykal, “Handoff performance improvement with latency reduction in next generation wireless networks,” *Wireless Networks*, vol. 11, no. 3, pp. 319–332, 2005.
- [6] R. Akester, “Reducing multicast collision loss for digital TV over 802.11 wireless networks,” in *Proc. WSEAS International Conference on Multimedia, Internet and Video Technologies*, 2004.
- [7] ATSC: advanced television system committee. [Online]. Available: <http://www.atsc.org/>

- [8] S. A. Azad and M. Murshed, "An efficient transmission scheme for minimizing user waiting time in Video-on-Demand systems," *IEEE Communications Letters*, vol. 11, no. 3, pp. 285–287, 2007.
- [9] Y. Birk and T. Kol, "Coding on demand by an informed source (ISCOD) for efficient broadcast of different supplemental data to caching clients," *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2825–2830, 2006.
- [10] J. W. Byers, M. Luby, and M. Mitzenmacher, "A digital fountain approach to asynchronous reliable multicast," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 8, pp. 1528–1540, 2002.
- [11] R. C. Chalmers, G. Krishnamurthi, and K. C. Almeroth, "Enabling intelligent handovers in heterogeneous wireless networks," *Mobile Networks and Applications*, vol. 11, no. 2, pp. 215–227, 2006.
- [12] B. J. Chang, S. Y. Lin, and Y. H. Liang, "Minimizing roaming overheads for vertical handoff in heterogeneous wireless mobile networks," in *ACM International Conference on Wireless Communications and Mobile Computing*, 2006, pp. 957–962.
- [13] Y. C. Chen, J. H. Hsia, and Y. J. Liao, "Advanced seamless vertical handoff architecture for WiMAX and WiFi heterogeneous networks with QoS guarantees," *Computer Communications*, vol. 32, no. 2, pp. 281–293, 2009.
- [14] Y. Choi and S. Choi, "Service charge and energy-aware vertical handoff in integrated IEEE 802.16e/802.11 networks," in *IEEE INFOCOM*, 2007, pp. 589–597.
- [15] E. K. P. Chong and S. H. Zak, *An Introduction to Optimization*. John Wiley & Sons Inc., 2008.
- [16] C.-H. Chu, D.-N. Yang, and M.-S. Chen, "Using network coding for dependent data broadcasting in a mobile environment," in *Proc. IEEE Global Telecommunications Conference*, 2007, pp. 5346–5350.

- [17] X. Chu and Y. Jiang, "Random linear network coding for peer-to-peer applications," *IEEE Network*, vol. 24, no. 4, pp. 35–39, 2010.
- [18] I. Djama and T. Ahmed, "A cross-layer interworking of DVB-T and WLAN for mobile IPTV service delivery," *IEEE Transactions on Broadcasting*, vol. 53, no. 1, pp. 382–390, 2007.
- [19] DMB: digital multimedia broadcasting in South Korea. [Online]. Available: <http://www.t-dmb.org/>
- [20] DVB mobile TV: DVB-H, DVB-SH, and DVB-IPDC. [Online]. Available: <http://www.dvb-h.org/>
- [21] O. Eerenberg, A. Koppelaar, A. M. Stuivenwold, and P. H. N. de With, "IP-recovery in the DVB-H link layer for TV on mobile," in *Proc. IEEE International Conference on Consumer Electronics*, 2006, pp. 411–412.
- [22] ETSI TR 102 469, "Digital video broadcasting (DVB); IP datacast over DVB-H: architecture," *European Telecommunications Standards Institute*, 2006.
- [23] ETSI TS 102 471, "Digital video broadcasting (DVB); IP datacast over DVB-H: electronic service guide," *European Telecommunications Standards Institute*, 2006.
- [24] ETSI TS 102 472, "Digital video broadcasting (DVB); IP datacast over DVB-H: content delivery protocols," *European Telecommunications Standards Institute*, 2006.
- [25] ETSI TS 102 591, "Digital video broadcasting (DVB); IP datacast over DVB-H: content delivery protocols (CDP) implementation guidelines," *European Telecommunications Standards Institute*, 2007.
- [26] G. Faria, J. A. Henriksson, E. Stare, and P. Talmola, "DVB-H: Digital broadcast services to handheld devices," *Proceedings of the IEEE*, vol. 94, no. 1, pp. 194–209, 2006.

- [27] C. Fragouli, J.-Y. L. Boudec, and J. Widmer, “Network coding: an instant primer,” *ACM SIGCOMM Computer Communication Review*, vol. 36, no. 1, pp. 63–68, 2006.
- [28] A. Garg and K. C. Yow, “Determining the best network to handover among various IEEE 802.11 and IEEE 802.16 networks by a mobile device,” in *IEEE International Conference on Mobile Technology, Applications and Systems*, 2005.
- [29] A. Ghosh, D. R. Wolter, J. G. Andrews, and R. Chen, “Broadband wireless access with WiMAX/802.16: Current performance benchmarks and future potential,” *IEEE Communications Magazine*, vol. 43, no. 2, pp. 129–136, 2005.
- [30] C. Gkantsidis and P. R. Rodriguez, “Network coding for large scale content distribution,” in *Proc. IEEE INFOCOM*, 2005, pp. 2235–2245.
- [31] D. Gomez-Barquero and A. Bria, “Repair mechanisms for broadcast transmissions in hybrid cellular & DVB-H systems,” in *Proc. IEEE International Symposium on Wireless Communication Systems*, 2006, pp. 398–402.
- [32] D. Gomez-Barquero, N. Cardona, A. Bria, and J. Zander, “Affordable mobile TV services in hybrid cellular and DVB-H systems,” *IEEE Network*, vol. 21, no. 2, pp. 34–40, 2007.
- [33] D. Gomez-Barquero, D. Gozalvez, and N. Cardona, “Application layer FEC for mobile TV delivery in IP datacast over DVB-H systems,” *IEEE Transactions on Broadcasting*, vol. 55, no. 2, pp. 396–406, 2009.
- [34] W. Guan, X. Ling, X. Shen, and D. Zhao, “Handoff trigger table for integrated 3G/WLAN networks,” in *ACM International Conference on Wireless Communications and Mobile Computing*, 2006, pp. 575–580.
- [35] B. Hechenleitner, “Repair cost of the IPDC/DVB-H file repair mechanism,” in *Proc. IEEE Wireless Telecommunications Symposium*, Pomona, CA, USA, 2008, pp. 137–144.

- [36] A. Helmy, M. Jaseemuddin, and G. Bhaskara, "Efficient micro-mobility using intra-domain multicast-based mechanisms (M&M)," *ACM SIGCOMM Computer Communication Review*, vol. 32, no. 5, pp. 61–72, 2002.
- [37] C. Heuck, "An analytical approach for performance evaluation of hybrid (broadcast/mobile) networks," *IEEE Transactions on Broadcasting*, vol. 56, no. 1, pp. 9–18, 2010.
- [38] P. Hummelbrunner, S. Buchinger, W. Robitza, D. Selig, M. Nezveda, and H. Hlavacs, "Peer to peer mobile TV recovery system," in *EuroITV '10: Proc. of the 8th International Interactive Conference on Interactive TV & Video*, Tampere, Finland, 2010, pp. 263–272.
- [39] M. Ibrahim, K. Khawam, A. E. Samhat, and S. Tohme, "Analytical framework for dimensioning hierarchical WiMAX-WiFi networks," *Computer Networks*, vol. 53, no. 3, pp. 299–309, 2009.
- [40] Y. C. Jung, B. K. Kim, and Y. T. Kim, "SIP based end-to-end QoS negotiation scheme for MIH," in *IEEE/IFIP International Workshop on Broadband Convergence Networks*, 2007.
- [41] V. P. Kafle, E. Kamioka, and S. Yamada, "A scheme for graceful vertical handover in heterogeneous overlay networks," in *IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*, 2006, pp. 343–348.
- [42] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Médard, and J. Crowcroft, "XORs in the air: Practical wireless network coding," *IEEE/ACM Transactions on Networking*, vol. 16, no. 3, pp. 497–510, 2008.
- [43] M. Kim and D. Kotz, "Periodic properties of user mobility and access-point popularity," *Personal and Ubiquitous Computing*, vol. 11, no. 6, pp. 465–479, 2007.

- [44] S. K. Kim, C. G. Kang, and K. S. Kim, "An adaptive handover decision algorithm based on the estimating mobility from signal strength measurements," in *IEEE Vehicular Technology Conference*, 2004, pp. 1004–1008.
- [45] 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, 2007.
- [46] D. Kouis, P. Demestichas, G. Koundourakis, and M. E. Theologou, "Resource management of IP-enabled DVB-T networks in the context of wireless B3G systems," *Wireless Networks*, vol. 13, no. 2, pp. 165–175, 2007.
- [47] D. Kouis, D. Loukatos, K. Kontovasilis, G. Kormentzas, and C. Skianis, "On the effectiveness of DVB-T for the support of IP-based services in heterogeneous wireless networks," *Computer Networks*, vol. 48, no. 1, pp. 57–73, 2005.
- [48] P. Larsson, "Multicast multiuser ARQ," in *Proc. IEEE International Conference on Wireless Communications and Networking Conference*, 2008, pp. 1985–1990.
- [49] U. Lee, J.-S. Park, J. Yeh, G. Pau, and M. Gerla, "CodeTorrent: Content distribution using network coding in VANET," in *Proc. the 1st international workshop on Decentralized resource sharing in mobile computing and networking*, 2006, pp. 1–5.
- [50] H. T. Lin, Y. Y. Lin, W. R. Chang, and R. S. Cheng, "An integrated WiMAX/WiFi architecture with QoS consistency over broadband wireless networks," in *IEEE Consumer Communications and Networking Conference*, 2009, pp. 1–7.
- [51] Z. Liu, C. Wu, B. Li, and S. Zhao, "UUSee: Large-scale operational on-demand streaming with random network coding," in *Proc. IEEE INFOCOM*, 2010, pp. 1–9.
- [52] K. Lu, Y. Qian, H.-H. Chen, and S. Fu, "WiMAX networks: from access to service platform," *IEEE Network*, vol. 22, no. 3, pp. 38–45, 2008.
- [53] G. Ma, Y. Xu, M. Lin, and Y. Xuan, "A content distribution system based on sparse network coding," in *Proc. NetCod*, 2007.

- [54] D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *SIAM Journal on Applied Mathematics*, vol. 11, no. 2, pp. 431–441, 1963.
- [55] G. Mastorakis, G. Kormentzas, and E. Pallis, "A fusion IP/DVB networking environment for providing always-on connectivity and triple-play services to urban and rural areas," *IEEE Network*, vol. 21, no. 2, pp. 21–27, 2007.
- [56] MediaFLO technologies: a division of Qualcomm incorporated. [Online]. Available: <http://www.mediaflo.com/>
- [57] S. Mohanty, "A new architecture for 3G and WLAN integration and inter-system handover management," *Wireless Network*, vol. 12, no. 6, pp. 733–745, 2006.
- [58] J. Nie, X. He, Z. Zhou, and C. Zhao, "Communication with bandwidth optimization in IEEE 802.16 and IEEE 802.11 hybrid networks," in *IEEE International Symposium on Communications and Information Technology*, 2005, pp. 26–29.
- [59] J. Nie, J. Wen, Q. Dong, and Z. Zhou, "A seamless handoff in IEEE 802.16a and IEEE 802.11n hybrid networks," in *IEEE International Conference on Communications, Circuits and Systems*, 2005, pp. 383–387.
- [60] J. Nie, L. Zeng, and J. Wen, "A bandwidth based adaptive fuzzy logic handoff in IEEE 802.16 and IEEE 802.11 hybrid networks," in *IEEE International Conference on Convergence Information Technology*, 2007, pp. 24–29.
- [61] J. Paavola, H. Himmanen, T. Jokela, J. Poikonen, and V. Ipatov, "The performance analysis of MPE-FEC decoding methods at the DVB-H link layer for efficient IP packet retrieval," *IEEE Transactions on Broadcasting*, vol. 53, no. 1, pp. 263–275, 2007.
- [62] T. Paila, M. Luby, R. Lehtonen, and V. Roca, "FLUTE – file delivery over unidirectional transport," *IETF RFC 3926*, 2004.

- [63] P. Pangalos, J. M. D. L. T. Verver, M. Dashti, A. Dashti, and H. Aghvami, "Confirming connectivity in interworked broadcast and mobile networks," *IEEE Network*, vol. 21, no. 2, pp. 13–20, 2007.
- [64] J.-S. Park, M. Gerla, D. S. Lun, Y. Yi, and M. Medard, "CodeCast: A network-coding-based ad hoc multicast protocol," *IEEE Wireless Communications*, vol. 13, no. 5, pp. 76–81, 2006.
- [65] F. A. Phiri and M. B. Murthy, "WLAN-GPRS tight coupling based interworking architecture with vertical handoff support," *Wireless Personal Communications*, vol. 40, no. 2, pp. 137–144, 2007.
- [66] M. Pischella, F. Lebeugle, and S. B. Jamaa, "UMTS to WLAN handover based on a priori knowledge of the networks," in *IEEE International Conference on Communications*, vol. 5, 2006, pp. 2009–2013.
- [67] D. Plets, W. Joseph, L. Verloock, E. Tanghe, L. Martens, E. Deventer, and H. Gauderis, "Influence of reception condition, MPE-FEC rate and modulation scheme on performance of DVB-H," *IEEE Transactions on Broadcasting*, vol. 54, no. 3, pp. 590–598, 2008.
- [68] J. Poikonen, J. Paavola, and V. Ipatov, "Aggregated renewal Markov process with applications in simulating mobile broadcast systems," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 1, pp. 21–31, 2009.
- [69] A. B. Pontes, D. D. P. Silva, J. Jailton, O. Rodrigues, and K. L. Dias, "Handover management in integrated WLAN and mobile WiMAX networks," *IEEE Wireless Communications*, vol. 15, no. 5, pp. 86–95, 2008.
- [70] J. Postel, "UDP: user datagram protocol," *IETF RFC 768*, 1980.
- [71] R. Prakash and V. V. Veeravalli, "Locally optimal soft handoff algorithm," in *IEEE Vehicular Technology Conference*, 2000, pp. 1450–1454.

- [72] C. Rauch, W. Kellerer, and P. Sties, “Hybrid mobile interactive services combining DVB-T and GPRS,” in *Proc. European Personal Mobile Communications Conference*, 2001.
- [73] R. M. Roth and A. Lempel, “Composition of Reed-Solomon codes and geometric designs,” *IEEE Transactions on Information Theory*, vol. 34, no. 4, pp. 810–816, 1988.
- [74] R. Schatz, N. Jordan, and S. Wagner, “Beyond broadcast—a hybrid testbed for mobile TV 2.0 services,” in *International Conference on Networking*, 2007.
- [75] R. Schatz, S. Wagner, and N. Jordan, “Mobile social TV: Extending DVB-H services with P2P-interaction,” in *Proc. IEEE International Conference on Digital Telecommunications*, 2007.
- [76] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, “RTP: a transport protocol for real-time applications,” *IETF RFC 3550*, 2003.
- [77] K. Sethom, H. Affi, and G. Pujolle, “A distributed and secured architecture to enhance smooth handoffs in wide area wireless IP infrastructures,” *Mobile Computing and Communications Review*, vol. 10, no. 3, pp. 46–57, 2006.
- [78] A. Shokrollahi, “Raptor codes,” *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2251–2567, 2006.
- [79] K. Sinkar, A. Jagirdar, T. Korakis, H. Liu, S. Mathur, and S. Panwar, “Cooperative recovery in heterogeneous mobile networks,” in *Proc. IEEE International Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, San Francisco, CA, USA, 2008, pp. 395–403.
- [80] L. Song, D. Kotz, R. Jain, and X. He, “Evaluating next-cell predictors with extensive Wi-Fi mobility data,” *IEEE Transactions on Mobile Computing*, vol. 5, no. 12, pp. 1633–1649, 2006.

- [81] A. Stephane, A. Mihailovic, and A. H. Aghvami, "Mechanisms and hierarchical topology for fast handover in wireless IP networks," *IEEE Communications Magazine*, vol. 38, no. 11, pp. 112–115, 2000.
- [82] P. Sties and W. Kellerer, "Radio broadcast networks enable broadband Internet access for mobile users," in *Proc. EUNICE*, 1999.
- [83] N. H. Vaidya and S. Hameed, "Scheduling data broadcast in asymmetric communication environments," *Wireless Networks*, vol. 5, no. 3, pp. 171–182, 1999.
- [84] M. Wang and B. Li, "R2: Random push with random network coding in live peer-to-peer streaming," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 9, pp. 1655–1666, 2007.
- [85] Y. Wang, P. H. Ho, S. Shen, S. Li, and S. Naik, "Modularized two-step vertical handoff scheme in integrated WWAN and WLAN," in *IEEE Workshop on High Performance Switching and Routing*, 2005, pp. 520–524.
- [86] J. S. Wu, S. F. Yang, and B. J. Hwang, "A terminal-controlled vertical handover decision scheme in IEEE 802.21-enabled heterogeneous wireless networks," *International Journal of Communication Systems*, vol. 22, no. 7, pp. 819–834, 2009.
- [87] D.-N. Yang and M.-S. Chen, "Data broadcast with adaptive network coding in heterogeneous wireless networks," *IEEE Transactions on Mobile Computing*, vol. 8, no. 1, pp. 109–125, 2009.
- [88] P. Yang, H. Deng, and Y. Ma, "Seamless integration of 3G and 802.11 wireless network," in *ACM International Workshop on Mobility Management and Wireless Access*, 2007, pp. 60–65.
- [89] S. F. Yang and J. S. Wu, "Handoff management schemes across hybrid WiMAX and WiFi networks," in *IEEE TENCON*, 2007.

- [90] W.-H. Yang, Y.-C. Wang, Y.-C. Tseng, and B.-S. P. Lin, "Spatial and temporal packet recovery schemes for DVB-H systems through IP-relay wireless networks," in *Proc. IEEE International Conference on Communications*, 2009.
- [91] W.-H. Yang, Y.-C. Wang, Y.-C. Tseng, and B.-S. P. Lin, "Energy-efficient network selection with mobility pattern awareness in an integrated WiMAX and WiFi network," *International Journal on Communication Systems*, vol. 23, no. 2, pp. 213–230, 2010.
- [92] W.-H. Yang, Y.-C. Wang, Y.-C. Tseng, and B.-S. P. Lin, "A request control scheme for data recovery in DVB-IPDC systems with spatial and temporal packet loss," *Wireless Communications and Mobile Computing*, to appear.
- [93] M. Ylianttila, J. Makela, and K. Pahlavan, "Analysis of handoff in a location-aware vertical multi-access network," *Computer Networks*, vol. 47, no. 2, pp. 185–201, 2005.
- [94] M. Ylianttila, M. Pande, J. Makela, and P. Mahonen, "Optimization scheme for mobile users performing vertical handoffs between IEEE 802.11 and GPRS/EDGE networks," in *IEEE Global Telecommunications Conference*, 2001, pp. 3439–3443.
- [95] A. H. Zahran, B. Liang, and A. Saleh, "Signal threshold adaptation for vertical handoff in heterogeneous wireless networks," *Mobile Network Application*, vol. 11, no. 4, pp. 625–640, 2006.
- [96] Q. A. Zeng and D. P. Agrawal, "Modeling and efficient handling of handoffs in integrated wireless mobile networks," *IEEE Transactions on Vehicular Technology*, vol. 51, no. 6, pp. 1469–1478, 2002.
- [97] J. Zhang, Y. P. Chen, and I. Marsic, "Network coding via opportunistic forwarding in wireless mesh networks," in *Proc. IEEE International Conference on Wireless Communications and Networking Conference*, 2008, pp. 1775–1780.

- [98] Q. Zhang, C. Guo, Z. Guo, and W. Zhu, “Efficient mobility management for vertical handoff between WWAN and WLAN,” *IEEE Communications Magazine*, vol. 41, no. 11, pp. 102–108, 2003.
- [99] X. Zhang, J. Liu, B. Li, and T.-S. P. Yum, “CoolStreaming/DONet: A data-driven overlay network for peer-to-peer live media streaming,” in *Proc. IEEE INFOCOM*, 2005, pp. 2102–2111.
- [100] Y. Zhu, B. Li, and J. Guo, “Multicast with network coding in application-layer overlay networks,” *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 1, pp. 107–120, 2004.
- [101] G. K. Zipf, *Human Behaviour and the Principle of Least Effort: An Introduction to Human Ecology*. Addison-Wesley, 1949.

