Prioritized Network Coding for Packet Recovery in Mobile Broadcasting Systems

Wen-Hsin Yang*†, You-Chiun Wang‡, Yu-Chee Tseng*§, and Bao-Shuh P. Lin*
*Department of Computer Science, National Chiao-Tung University, Hsinchu, 300, Taiwan
†Information and Communications Research Labs, ITRI, Chutung, Hsinchu, 310, Taiwan
‡Department of Computer Science and Engineering, National Sun Yat-Sen University, Kaohsiung, 804, Taiwan
§Research Center for Information Technology Innovation, Academia Sinica, Taipei, 115, Taiwan
Email: wenhsin@cs.nctu.edu.tw; ycwang@cse.nsysu.edu.tw; yctseng@cs.nctu.edu.tw; bplin@mail.nctu.edu.tw

***Abstract*—Digital mobile broadcasting has received lots of attention recently. Due to broadcasting nature, packet loss is inevitable and the packet recovery issue needs to be addressed. The paper 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 a prioritized network coding scheme based on XOR coding, namely *RGS*. The 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. Simulation results verify that RGS can improve recovery efficiency while reduce the number of packets being dropped due to missing deadlines.**

***Index Terms*—broadcast, data recovery, digital video broadcasting (DVB), mobile broadcasting, network coding, WiMAX, wireless network.**

I. INTRODUCTION

Digital video broadcasting–handheld (DVB-H) is developed to support multimedia broadcasting services for *mobile devices (MDs)* [1]. 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 [2], which can serve as a recovery channel for MDs to request for lost packets. Fig. 1 shows the network architecture considered in the paper, 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 [3]. Since MDs may miss different pieces of a video stream, network coding is very suitable. However, video data has real-time constraints to be addressed. In addition, lost packets in DVB-H often exhibit *spatial* and *temporal* correlations [4]. 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.

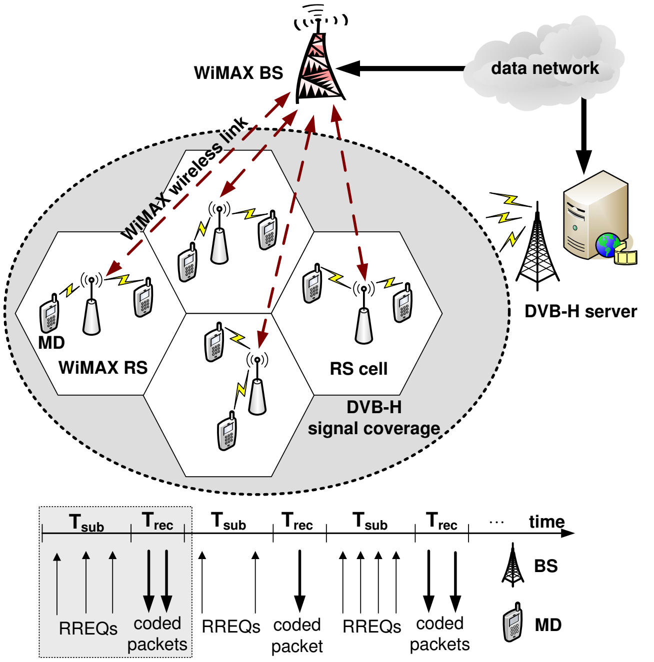


Fig. 1: A DVB-IPDC system with WiMAX networks as the recovery channel.

In the paper, 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. 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 κ slots. Given some lost packets and their deadlines, we formulate a new *prioritized network coding problem*, whose goal is to arrange at most κ 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 aim at reducing network traffic and do not address the priority issue.

To solve the prioritized network coding problem, we develop a *recovery by greedy selection (RGS)* scheme based on XOR coding. The RGS scheme 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 queried 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. Simulation results show that RGS can efficiently recover packets while reduce the number of dropped packets.

The rest of this paper is organized as follows: Section II surveys related work. Section III defines our prioritized network coding problem. Section IV presents the RGS scheme. Section V shows our simulation results. Conclusions are drawn in Section VI.

II. 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.

A. Data Recovery in DVB-IPDC Systems

Several studies suggest adopting a cellular network as the recovery channel for DVB-IPDC. Reference [5] considers transmitting parity data over the DVB-H or the cellular network. The later may use dedicated point-to-point connections or cell broadcasting. *Content delivery protocol* for regulating the cellular network's behaviors is defined in [6]. Through extensive simulations, [7] points out some 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 will serve as a *super peer* responsible for data recovery in its group [8]. WiMAX networks are considered as the recovery channel in [4], where the group packet loss property, which usually exhibits high spatial and temporal correlations, is discussed. Instead of using infrastructure networks, [9] proposes organizing MDs as a wireless ad hoc network to share lost packets through peer-to-peer links. All these studies do not consider incorporating network coding in the recovery process.

B. Network Coding for Error Recovery

Several research efforts consider adopting network coding to deal with the error recovery issue for data transmission. For example, the work of [10] extends the multiuser *automatic repeat request (ARQ)* to multicasting applications. Using the multicast ARQ, the sender can calculate the code weights for data packet linear combinations based on the feedback information from the receivers regarding their successfully received data. Then, each receiver can decode the linearly combined packets by exploiting its previously received data. The work of [11] 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 clients' states and then adopts erasure-correction codes for data transmissions. Each client then uses its cached data and the received data to derive its requested blocks. By clustering MDs into multiple cells, the work of [12] adaptively codes the data according to the data temporarily stored in each MD to minimize the bandwidth consumption. Such cell selection problem and broadcast coding problem are formulated by integer programming and shown to be NP-hard. It can be observed that the above work does not address the prioritized network coding problem. In this paper, 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.

III. PROBLEM DEFINITION

Fig. 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. 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 κ slots so that the BS is allowed to broadcast at most κ 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 κ 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 scheme in the next section.

IV. PROPOSED SOLUTION

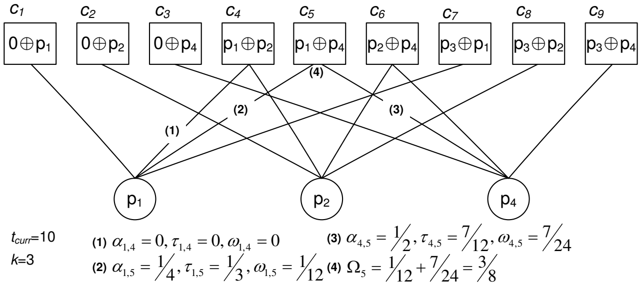
We are given n MDs, where each MD$_i$ may send one RREQ r_i in every $\mathcal{T}_{sub}$ period. For each round, t_{curr} records the time that the $\mathcal{T}_{sub}$ period ends and κ is the duration of the successive one $\mathcal{T}_{rec}$ after the $\mathcal{T}_{sub}$. Each r_i contains a set of 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 solution first constructs a weighted bipartite 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}.

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). $\mathcal{C} = \{c_1, \cdots, c_k\}$ 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. 2 gives an example on the conditions of the current time frame $t_{curr} = 10$ and the retransmission time slots $\kappa = 3$. As illustrated in Fig. 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 $\mathcal{C} = \{c_1, c_2, \cdots, c_9\}$ as shown in Fig. 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 $\mathcal{R}$.

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}　　s_2={p_1, p_2, p_3}, q_2={p_4}
s_3={p_3, p_4}, q_3={p_1, p_2}　　s_4={p_1, p_3, p_4}, q_4={p_2}

(a)



(b)

Fig. 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 $\mathcal{Q}$.

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

$$\mathcal{D}_j^i = \bigcup_{\forall r_n \in \mathcal{R}} r_n \tag{1}$$

$$\text{where } (p_i = c_j) \text{ or } (\exists p_m \in s_n \text{ s.t. } p_m \oplus p_i = c_j).$$

$\|\mathcal{D}_j^i\|$ and $\|\mathcal{R}\|$ are the numbers of the RREQs in the sets $\mathcal{D}_j^i$ and $\mathcal{R}$, respectively. $\tau_{i,j} = \sum_{\forall r_n \in \mathcal{R}, r_n \in \mathcal{D}_j^i} \frac{1}{\max((d_{r_n} - t_{curr}), \kappa)}$ 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, κ is maximum retransmission time slots which is used as the length of sliding window for retransmissions (hereof, d_{r_n}, t_{curr} and κ adopt the same time unit). Therefore, the equation of $\omega_{i,j}$ can be given as follows.

$$\omega_{i,j} = \alpha_{i,j} \times \tau_{i,j}$$

$$= \sum_{\forall r_n \in \mathcal{R}, r_n \in \mathcal{D}_j^i} \frac{\|\mathcal{D}_j^i\|}{\|\mathcal{R}\| \times \max((d_{r_n} - t_{curr}), \kappa)}. \tag{2}$$

After constructing the graph $\mathcal{G}$, we then find the maximum weight minimum dominating 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. (2), for any coded packet $c_j \in \mathcal{C}$ whose Ω_j is calculated by

$$\begin{aligned}\Omega_j &= \sum_{\forall p_i \in \mathcal{Q}} \omega_{i,j} \\ &= \sum_{\forall p_i \in \mathcal{Q}} \sum_{\forall r_n \in \mathcal{R}, r_n \in \mathcal{D}_j^i} \frac{\|\mathcal{D}_j^i\|}{\|\mathcal{R}\| \times \max((d_{r_n} - t_{curr}), \kappa)}.\end{aligned} \tag{3}$$

For instance, in Fig. 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

$$f_j = \bigcup_{\forall p_i \in \mathcal{Q}} p_i \text{ where } \omega_{i,j} > 0, \tag{4}$$

and the dominator set for all coded packets $\mathcal{F}$ is the union of all f_js, hence, $\mathcal{F} = \bigcup_{\forall c_j \in \mathcal{C}} f_j$. 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 $\mathcal{C}$ be the set of the selected f_js and the set of queried packets that have not been covered by any f_j be $\mathcal{U}$ amid the iterative loop procedure. The default value of $\mathcal{C}$ 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 ($\mathcal{U} = \phi$).

- Fist, we choose one f_j from $\mathcal{F}$ respecting to any coded packet c_j if the condition $\max(f_j \bigcap \mathcal{U})$ 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 $\mathcal{C}$ ($\mathcal{C} = \mathcal{C} \bigcup \{c_j\}$).

Finally, we shape the solution set $\mathcal{C}$ whose size fits to κ. Thus, if the number of the selected coded packets in $\mathcal{C}$ is less than or equal to κ, then the whole set of $\mathcal{C}$ is the solution for RGS directly; otherwise, outputs the first-κ coded packets in $\mathcal{C}$ for the solution of RGS. The example in Fig. 2(b), the solution set $\mathcal{C} = \{c_8, c_6, c_5\}$ or $\{c_2, c_6, c_5\}$ under the condition of $\kappa = 3$.

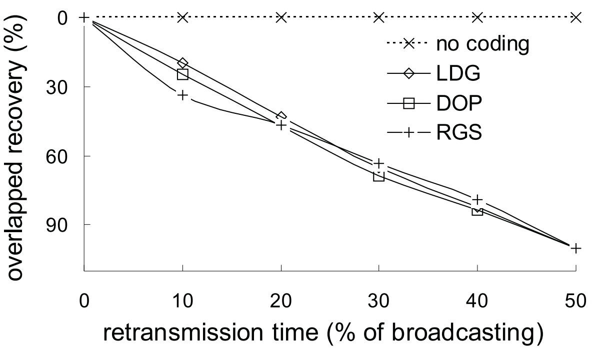
V. EXPERIMENT RESULTS

We compare our RGS scheme with a no coding scheme and the ISCOD approach methods in which Least Difference Greedy (LDG) algorithm[11] and Demand-Oriented Pairing (DOP) algorithm[12] are simulated. Assume the DVB-H broadcasting packets rate is constant in $\mathcal{T}_{sub}$, and $\mathcal{T}_{rec}$ is proportional to $\mathcal{T}_{sub}$, which ranges $10\%, 20\%, \cdots, 50\%$ of $\mathcal{T}_{sub}$. We model the packet loss happened in $\mathcal{T}_{sub}$ to be a uniform distribution and 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 lost packet to be retransmitted.

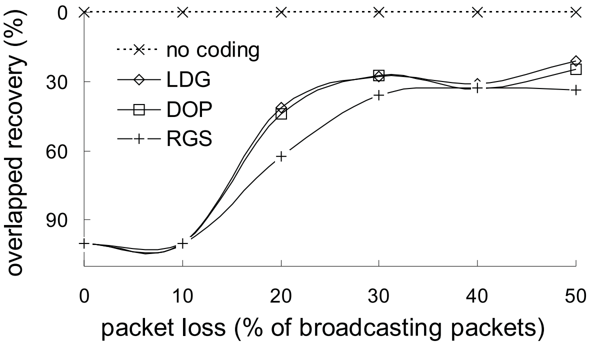
Depending on the sequence order of the left time before due of recovery for the lost packets, RGS scheme selects the corresponding coded packets that can decode for the queried packets by considering the factor of Ω (consists of deadline and the number of the influenced RREQs). Moreover, we assume the pattern of the RREQs that recover their corresponding lost 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 a simulated scheme chooses a coded packet that can recover in time to the queried packet p$_i$, and $\theta_i = (\frac{1}{i})^\gamma / \sum_{j=1}^{\|\mathcal{Q}\|} (\frac{1}{j})^\gamma$ where γ is the skewness parameter, $1 \leq i \leq \|\mathcal{Q}\|$. By comparing to the no coding scheme, we set *Overlapped Recovery Rate* (Θ) which comes from θ_i with $\gamma = 0.6$ 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 scheme being more efficient in terms of much more RREQs and their requesting queried packets are recovered in time. We observe the value of Θ depending on the varying retransmission time κ and packet loss distribution as follows.

We first conduct a simulation that a half of the DVB-H broadcasting packets are lost and uniformly distributed to each MD in a period of time $\mathcal{T}_{sub}$. Each recovery scheme performs the retransmissions for coded packets in the given time duration κ which is varied from 10% to 50% of $\mathcal{T}_{sub}$, and then investigates the corresponding number of the Θ. The larger number of κ means there should be much more coded packets can be retransmitted. Fig. 3(a) shows the changes of Θ respecting to the retransmission time slot κ. By comparing the value of Θ, RGS is superior to the LDG and DOP. As similar as LDG and DOP, RGS adopts a greedy search for coded packets, however, RGS takes advantage of getting higher number of Θ because it takes the due time factor into account that may affect much more queried packets to be recovered. Once the retransmission time increases about 10% of $\mathcal{T}_{sub}$ each time, the standard deviation of Θ of performing LDG, DOP and RGS are 28.3%, 26.6% and 23.4%, respectively.

Next, we simulate the recovery schemes with regard to different packet losses whose loss probabilities are varied



(a) packet loss rate = 50%



(b) retransmission time = 10% $\mathcal{T}_{sub}$

Fig. 3: The changes of overlapped recovery rate:(a)different retransmissions time duration; (b)different packet losses.

from 10% to 50% of $\mathcal{T}_{sub}$, and then investigate their corresponding results of Θ. Each recovery scheme performs the retransmissions for coded packets only in the time duration $\kappa = 10\%$ of $\mathcal{T}_{sub}$. The larger probability of packet loss represents there should be much more coded packets need to retransmit. Fig. 3(b) shows the simulation results of the changes of Θ corresponding to each scheme in which RGS is 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 performing LDG, DOP, and RGS scheme are 28.6%, 27.8% and 25.9%, respectively.

Finally, we observe the Θ of each scheme when the retransmission time is not enough for it to deal with much more broadcasting packet losses, which compares to the Θ of LDG scheme. When the retransmission time κ is not enough (but not too short) for recovering all of the requesting packet losses, the values of Θ performing DOP and RGS are on average 1.1 and 1.2 times of adopting LDG scheme, respectively. On the other hand, when the retransmission time κ is too short for recovering all of the requesting packet losses, averagely, the values of Θ performing DOP and RGS are 1.1 and 1.3 times of adopting LDG scheme, respectively. The simulation results support that RGS can keep the efficiency by decoding the coded packets for recovering most of the requesting queried packets while the due time factor of recovery is considered.

VI. CONCLUSIONS

Mobile broadcasting services are available and believed to become more popular in broadcast and wireless networks. However, how to efficiently recover packet loss occurring at MDs is a big challenge because of the nature of wireless broadcast. In this paper, We applied a network coding method to improve the efficiency of the retransmissions for DVB-H packet losses through the WiMAX network, in which the prioritized network coding problem was addressed. We have proposed the RGS scheme to determine the retransmissions of the coded packets in the constrained κ time slots, which can support the maximal number of mobile devices to get back their queried packets and serve for the maximum amount of recovery requests whose corresponding deadlines cannot be violated. Simulation results show that our scheme can averagely increase $1.2 \sim 1.3$ times more overlapped recovery rate compared to the ISCOD approach schemes. 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.

REFERENCES

[1] 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.

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

[3] 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.

[4] 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*, 2011.

[5] 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.

[6] European Telecommunications Standards Institute, "Digital video broadcasting (DVB); IP datacast over DVB-H: content delivery protocols," ETSI TS 102 472, 2006.

[7] B. Hechenleitner, "Repair cost of the IPDC/DVB-H file repair mechanism," in *Proc. IEEE Wireless Telecommunications Symposium*, 2008, pp. 137–144.

[8] P. Hummelbrunner, S. Buchinger, W. Robitza, D. Selig, M. Nezveda, and H. Hlavacs, "Peer to peer mobile TV recovery system," in *Proc. the 8th International Interactive Conference on Interactive TV&Video*, 2010, pp. 263–272.

[9] 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*, 2008, pp. 395–403.

[10] P. Larsson, "Multicast multiuser ARQ," in *Proc. IEEE International Conference on Wireless Communications and Networking Conference*, 2008, pp. 1985 –1990.

[11] 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.

[12] 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.