Adaptive Low-Complexity Erasure-Correcting Code-Based Protocols for QoS-Driven Mobile Multicast Services Over Wireless Networks

Xi Zhang, *Senior Member, IEEE*, and Qinghe Du

***Abstract*—We propose an adaptive hybrid automatic repeat request–forward error correction (ARQ–FEC) erasure-correcting scheme for quality of service (QoS)-driven mobile multicast services over wireless networks. The main features of our proposed scheme include (i) the low complexity achieved by the graph code; (ii) dynamic adaptation to the variations of packet-loss level and QoS requirements. To increase error-control efficiency and support diverse QoS requirements, we develop a two-dimensional (2-D) adaptive error-control scheme that dynamically adjusts not only the error-control redundancy, but also the code mapping structures. By deriving and identifying the closed-form *nonlinear* analytical expression between the optimal check-node degree and the packet-loss level, we propose the *nonuniformed* adaptive coding structures to achieve high error-control efficiency. Applying the Markov chain model, we obtain closed-form expressions that derive the error-control redundancy as a function of packet-loss level and the optimal check-node degree in each adaptation step. The convergency of error-control redundancy adaptation is dynamically controlled by different QoS requirements such that a high error-control efficiency can be achieved. Using the proposed 2-D adaptive error control, we design an efficient hybrid ARQ–FEC protocol for mobile multicast services with diverse reliability QoS requirements. The proposed scheme keeps the feedback overhead low by consolidating only the numbers rather than the sequence numbers of the lost packets, which are fed back by multicast receivers. Also conducted is a set of numerical and simulation evaluations that analyze and compare our proposed adaptive scheme with those using nonadaptive graph codes, Reed–Solomon erasure codes (RSE), and the pure ARQ-based approach. The simulation results show that our proposed scheme can efficiently support QoS-driven mobile multicast services and achieve high error-control efficiency while imposing low error-control complexity and overhead for mobile multicast networks.**

***Index Terms*—Adaptive hybrid automatic repeat request–forward error correction (ARQ–FEC), error control, graph codes, low-complexity erasure codes, mobile multicast, quality of service (QoS), wireless networks.**

I. Introduction

WITH the rapid progress of cost-effective and powerful portable computer and wireless networks, there has been a significant increase in demand for multicast services over mobile networks. Mobile multicast provides a highly efficient and flexible way of simultaneously disseminating data or information from one source to multiple location-independent receivers [1]. Consequently, mobile multicast gains a wide spectrum of applications, including highway mobile traffic monitoring/updating, emergency warnings, air traffic control, remote teleconferencing, and distance learning. On the other hand, the provision of quality of service (QoS) guarantees with limited wireless resources is critically important for the development of mobile networks, and the mobile multicast is also often required to be capable of providing services flexibly according to different QoS requirements. Clearly, QoS-aware/driven characteristics have already become one of the most important parts in the design of mobile multicast schemes.

As in wired and/or unicast networks, error control not only plays an important role for reliable mobile multicast services over wireless networks, but also provides an efficient means of supporting QoS diversities for different mobile multicast services over different mobile users. However, mobile multicast imposes many new challenges in error control for supporting diverse QoS, which are not encountered in wired and/or unicast networks. First, mobile multicast itself causes feedback implosion problems in error-control protocols [2]–[4]. Second, retransmission-based error control is not scalable with multicast group size since retransmission overhead and unnecessary retransmissions grow up quickly as the number of multicast receivers increases [5], [6]. Third, packet-loss probabilities over wireless channels vary dramatically when user mobilities vary significantly and hand-offs occur frequently. Finally, wireless channels are highly asymmetric where the energy/processing power on uplink from mobile users is much less than that on downlink from the base station. Clearly, the problem on how to efficiently integrate error control with supporting QoS diversity for mobile multicast, despite its vital importance, has been neither well understood nor thoroughly studied.

There are mainly two categories of error-control techniques, namely, 1) automatic repeat request (ARQ) and 2) forward error correction (FEC) erasure coding. ARQ attempts to retransmit lost packets while FEC adds error-control redundancy into the packet flow such that the receivers recover from packet losses without sending error-control feedback to the sender for retransmission. Clearly, FEC is more suitable for error control over mobile multicast services since it can avoid feedback implosion, scales well with multicast tree size, and significantly reduces the feedback cost of precious energy/processing power at mobile users. In addition, with FEC, any *one* repairing packet can repair the loss of different data packets

Manuscript received October 2, 2005; revised January 21, 2006. This work was supported in part by the U.S. National Science Foundation CAREER Award under Grant ECS-0348694. The review of this paper was coordinated by Prof. X. Shen.

The authors are with the Networking and Information Systems Laboratory, Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX 77843 USA (e-mail: xizhang@ece.tamu.edu; duqinghe@ece.tamu.edu).

Digital Object Identifier 10.1109/TVT.2006.874547

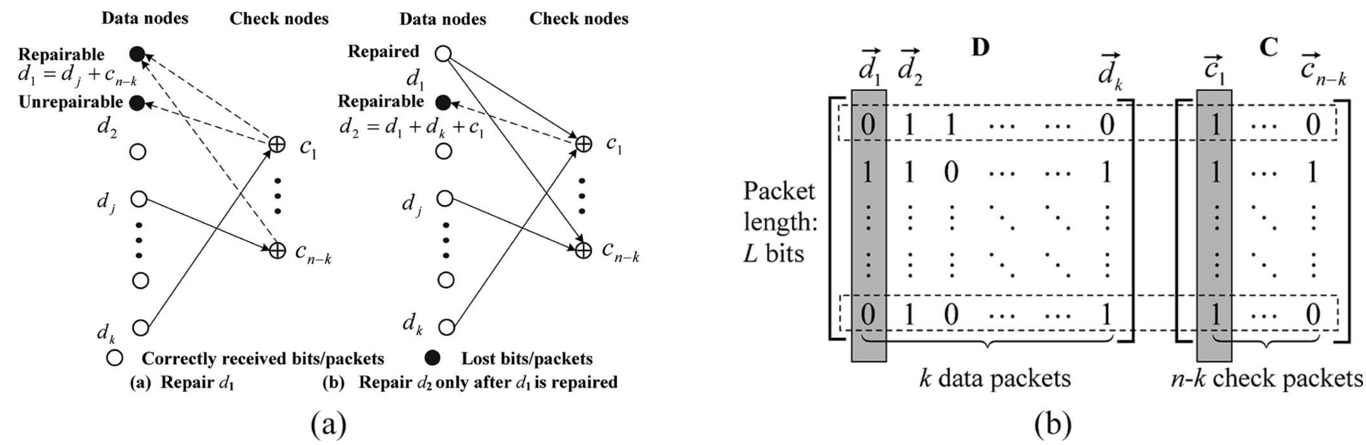


Fig. 1. (a) Iterative decoding for graph codes. (b) Employing graph codes in the packet level, where $\mathbf{D}$ forms a transmission group (TG).

at different multicast receivers [7] since FEC is a packet *sequence-number-independent* error control technique. As a result, a significant amount of research for error control in either multicast or wireless networks has mainly focused on the FEC-based schemes [7].

Most previous FEC-based multicast error-control schemes for multicasting applications mainly focused on the use of Reed–Solomon erasure (RSE) codes [7], [8]. However, there are several severe problems inherently associated with RSE-based schemes when they are applied in mobile multicast. First, the error-control redundancy level needs to be dynamically regulated according to the variation of the wireless channels' qualities. Second, the maximum error-control redundancy is upper-bounded by the RSE code symbol size, which may lead to decoding failures when wireless channel loss probabilities increase tremendously. Third, the RSE codes' fixed code structures and decoding algorithm cannot be adjusted according to the QoS variations of multicast mobile users. Finally and more importantly, the implementation complexity of RSE coding is too high, particularly when RSE block and symbol sizes are large, to be applicable to the mobile multicast networks where both energy and processing power are severely constrained at mobile users. To overcome these aforementioned problems, we propose a new adaptive low-complexity graph-code-based hybrid ARQ–FEC scheme for QoS-driven mobile multicast services. The main features of our proposed scheme are twofold: the low complexity and dynamic adaptation to the variations of packet-loss level and QoS requirements of multicast mobile users. In addition, unlike the existing RSE-code-based schemes, our proposed scheme can automatically adjust the error-control redundancy level according to different QoS requirements.

This paper is organized as follows. Section II introduces the low-complexity graph codes used for error corrections. Section III describes the system model for the QoS-driven mobile multicast and defines the performance evaluation metrics. Section IV proposes the two-dimensional (2-D) adaptive mobile multicast error-control scheme and presents its analytical and numerical analyses. Section V evaluates the performance of our proposed schemes through simulations. The paper concludes with Section VI.

II. Low-Complexity Erasure Graph Codes

The principle and structure of the graph code [9] can be described by a bipartite graph shown in Fig. 1(a). A bipartite graph consists of two disjoint classes of nodes. Two nodes in different classes can be connected by an edge, but there are no edges connecting any two nodes within the same class. The number of edges connected to a node is called *degree* of that node. In a bipartite graph, each node on the left-hand side, representing a data bit, is called a data node. Each node on the right-hand side, representing a parity check bit, is called a check node. Consider a graph code of length n with k data nodes and $(n-k)$ check nodes in a bipartite graph. Let d_i denote the ith data bit and c_j denote the jth check bit. We call the edge connection pattern between data nodes and check nodes in the bipartite graph the *mapping structure* of the graph code, each of which determines a specific graph code structure.

As shown in Fig. 1(a), each check bit is calculated as the sum in Galois Fields [GF(2)] of all the data bits connected to it. Graph codes can *iteratively* correct/repair erasure errors by decoding through simple modulo-2 additions [9] (we use "+" to represent modulo-2 additions in all encoding/decoding operations throughout this paper) as follows.

- Step 1: Search for the check bits that are connected with *only one* lost data bit.
- Step 2: Recover corresponding lost data bits according to this code mapping structure.
- Step 3: Go back to Step 1 until all the lost data bits are repaired or no more can be repaired.

Fig. 1(a) shows an example of this procedure. First, assume that d_1 and d_2 are the only lost bits as shown in Fig. 1(a)-(i). Thus, only d_1 can be repaired by $d_1 = d_j + c_{n-k}$. Following this, d_2 can be iteratively repaired by $d_2 = d_1 + d_k + c_1$, as shown in Fig. 1(a)-(ii). Clearly, it is possible that some lost data bits still cannot be repaired even after the iterative decoding procedure ends, depending on the code's mapping structure used and which/how many data bits are lost.

The graph code mapping structures can be algebraically expressed by the code structure matrix $\mathbf{P} = (p_{ij})_{k\times(n-k)}$ with $p_{ij} \in \{0,1\}$, where p_{ij} equals 1 (0) if the ith data bit is (not) connected to the jth check bit in the bipartite graph. Then, we can obtain the $(n-k)$-bit-long check-bit vector $\mathbf{c}$ by the simple encoding procedure as follows:

$$\mathbf{c} \stackrel{\triangle}{=} [c_1 c_2 \cdots c_{n-k}] = \mathbf{d}\mathbf{P}_{k\times(n-k)} \tag{1}$$

in GF(2) from the k-bit-long data-bit vector $\mathbf{d} \stackrel{\triangle}{=} [d_1 d_2 \cdots d_k]$. Considering systematic graph codes, the generating matrix of graph codes can be expressed as $\mathbf{G}_{k\times n} = [\mathbf{I}_{k\times k}\mathbf{P}_{k\times(n-k)}]$, and an n-bit-long code word can be generated by $\mathbf{w} = \mathbf{d}\mathbf{G}_{k\times n}$. Then, the degrees of the ith data node, denoted by α_i, and jth check node, denoted by γ_j, are equal to the number of 1's in the ith row and jth column of $\mathbf{P}$, respectively. We also call α_i and γ_j the weights of the ith row and jth column, respectively. Generally, in order to increase the probability of successful decoding/repairing and reduce the computational complexity, α_i and γ_j usually need to be much smaller than k. This implies that a sparse $\mathbf{P}$ is generally required.

The most important advantage of graph-code-based error-control schemes [9], [10] is that the encoding/decoding time complexity is much lower as compared to RSE-code-based schemes. Consequently, the graph-code-based error-control scheme has been applied into the asynchronous reliable multicast transmission [11] to achieve high efficiency while keeping the error-control complexity low. In addition, the decoding procedures for graph codes can be iteratively performed with any number of check packets correctly received instead of having to wait until at least k distinct packets (including both data and check packets) are correctly received, like in the decoding of RSE codes. This can help save a significant amount of bandwidth for QoS-driven mobile multicast services. Moreover, graph-code-based schemes enable code structures to be adaptive for improving the error-control efficiency.

To extend graph codes to the packet level in implementing hybrid ARQ–FEC-based multicast services over wireless networks, we divide the source data packet stream into blocks each consisting of k consecutive data packets, which form *transmission groups* (TG) [see Fig. 1(b)]. Assuming the packet length is L bits, we denote a data packet by an $L \times 1$ column vector $\vec{d_i}$, where $i = 1, 2, \ldots, k$, as shown in the solid-lined box on the left-hand side in Fig. 1(b). Let k data packets form a data matrix $\mathbf{D}_{L\times k}$, as shown in Fig. 1(b), where the jth column comes from the jth data packet and the ith row consists of ith bit of all k data packets. Then, the encoding procedure given in (1) can be used to generate a $1 \times (n-k)$ check-bit vector in the ith $(i = 1, 2, \ldots, L)$ row of the check matrix $\mathbf{C}$. The data bits in a row and corresponding check bits form a code word as shown in a dash-lined box in Fig. 1(b). All the jth check bits in each row of $\mathbf{C}$ form the jth check packet with L bits long, denoted as $\vec{c}_j$, where $j = 1, 2, \ldots, (n-k)$, as shown in the solid-lined box on the right-hand side in Fig. 1(b). The above encoding procedure at packet level can be algebraically expressed in GF(2) by

$$\mathbf{C}_{L\times(n-k)} = \mathbf{D}_{L\times k}\mathbf{P}_{k\times(n-k)} \tag{2}$$

which is virtually the same as the encoding procedure given in (1) at bit level.

III. System Model of Hybrid ARQ–FEC-Based Mobile Multicast

A. Hybrid ARQ–FEC-Based Mobile Multicast Transmission Model

We model the mobile multicast transmission system by a multicast tree, which consists of one sender and a number of mobile multicast receivers. The sender multicasts a stream of data packets to each receiver with the required packet-loss-rate QoS, denoted by ξ [see (3)]. We assume that the packet losses are independent and identically distributed (i.i.d.) in terms of time (for different packets) and space (for different receivers). The assumption of i.i.d. loss for different packets is particularly suitable for wireless networks, where the random loss often happens, unlike the wired networks, where the data loss usually occurs in the bursty fashion due to the congestion in bottlenecks. It should be also noted that FEC codes usually have much higher erasure-correcting capability for random loss than for bursty loss. The integrated ARQ-FEC error-control schemes are implemented through closed-loop information exchanges by using forward and feedback control packets between the sender and the receivers in the mobile multicast tree. Error-control information is exchanged in each *transmission round* (TR), which is defined as follows. To implement the adaptive error control, a TG of data packets is usually transmitted through a number of TR's. Each TR begins with the sender multicasting k data packets (i.e., data-packet TR or retransmission round) or a certain number of check packets (i.e., check-packet TR), and ends with the sender having received consolidated feedbacks from all multicast receivers. So, TR is also the basic control period of adaptation, where TR is indexed by $t = 1, 2, \ldots$.

The packet stream from the data source is divided into a number of TGs each with k data packets. For each TG, the sender multicasts the k data packets in the first TR. Then, the sender waits until all feedback packets arrive, which carry the error-control information from the mobile receivers. Based on the feedback error-control information (e.g., the packet-loss level, to be detailed later), the sender determines to transmit either a new next TG or a number of parity-check packets to repair losses for the current TG. Specifically, unless the reliability–QoS [to be detailed later in (3) and Section III-B] is satisfied by all receivers, the sender must generate a number of check packets from the k data packets of the current TG and then multicast them to all mobile receivers for loss repairing. This loss-repairing procedure repeats until the reliability–QoS requirement is satisfied by all mobile receivers. However, if the reliability–QoS fails to be satisfied after all the available check packets have been generated and transmitted, the retransmission of the current TG must be executed by the sender. In addition, we assume that the control information such as the packet sequence number and the packet-loss level in each TR can be reliably transmitted between the sender and receivers. To achieve excellent performance, several parameters need to be selected carefully. A set of parameter selection algorithms are presented in Section IV.

B. Different QoS Requirements for Mobile Multicast Services

While there are a wide range of QoS metrics, we mainly focus on the QoS metrics closely associated with error control for mobile multicast, which include reliability and transmission delays. To efficiently use the limited resources in mobile wireless networks while supporting QoS requirements, the error-control parameters need to be adjusted dynamically according to different QoS requirements for different mobile multicast services. In particular, real-time (e.g., video/audio) mobile multicast services must upper-bound the transmission delay, but can tolerate certain packet losses, implying that a relatively higher packet-loss rate is allowed than that for reliable services. Furthermore, this required loss-rate QoS threshold can be increased (or decreased) as the required quality of received audio/video streams decreases (or increases). On the other hand, data mobile multicast services must have zero loss while tolerating a certain transmission delay. As a result, the various QoS requirements of interest in this paper can be characterized by the reliability–QoS. Thus, we define the required reliability–QoS by packet-loss rate, denoted by ξ. To complete the transmission of a TG with the required packet-loss rate QoS ξ, the following condition must be satisfied by all receivers:

$$\frac{f_r(t)}{k} \leq \xi, \qquad \forall\, 1 \leq r \leq R \tag{3}$$

where $f_r(t)$ is the number of lost/unrepaired data packets (the packet-loss level) of a TG for the rth receiver after the t-th $(t = 1, 2, \ldots)$ TR. Note that reliability–QoS is not the only QoS measure in this paper. On the condition that reliability–QoS requirements must be satisfied, we also consider other QoS metrics such as average delay and so on, which are defined in Section III-D. For our proposed error-control scheme, once the above condition is satisfied after a certain number t of TR's, the sender completes sending check/repairing packets for the current TG and then immediately starts transmitting the next new TG. As a result, a significant amount of bandwidth can be saved for graph-code-based error-control schemes, where the decoding procedure can proceed iteratively and cumulatively for *any* given number of check packets correctly received. By contrast, RSE-code-based schemes do not have this advantage because the decoding procedure cannot start until at least k distinct data/check packets have been correctly received at any mobile multicast receiver. Note that throughout this paper, we use two similar terms that have different meanings, namely, 1) *packet-loss rate*, denoted by ξ, represents the required reliability–QoS and 2) *packet-loss probability*, denoted by p, represents the channel quality.

C. Cost-Effective Feedback Signaling Algorithms

To solve the feedback explosion and synchronous problems, we propose to use our previously developed soft synchronous protocol (SSP) [2]–[4] in this adaptive protocol for mobile multicast services, which consolidates the numbers $f_r(t)$, $r \in \{1, 2, \ldots, R\}$, of lost data packet for the rth receiver in the t-th TR by selecting/feeding back the maximum number $\theta_{\max}(t)$ of lost packets among all receivers as

$$\theta_{\max}(t) \triangleq \max_{r \in \{1,2,\ldots,R\}} \{f_r(t)\} \tag{4}$$

in the t-th TR with $t = 1, 2, \ldots$. Note that the feedback consolidation procedure given in (4) is just the general procedure, which in fact is iteratively implemented at each branch node within that multicast subtree. Thus, (3) can be equivalently rewritten as

$$\frac{\theta_{\max}(t)}{k} = \frac{\max\limits_{r \in \{1,2,\ldots,R\}} \{f_r(t)\}}{k} \leq \xi, \qquad \forall\, 1 \leq r \leq R. \tag{5}$$

By using SSP, the packet-sequence-independent error-control schemes can be efficiently applied. The feedbacks only contain information on the *number* of lost packets rather than a series of the *sequence numbers* of lost packets during each TR. Consequently, the feedback bandwidth overhead is significantly reduced. Note that by using SSP, the sender adjusts error-control parameters for each next TR only based on the worst packet-loss level among all receivers. For the detailed SSP, see [2]–[4].

D. Performance Metrics

For the FEC-based error-control protocols/schemes used in mobile multicast, we use following metrics to evaluate their performance.

D.1. Bandwidth Efficiency η

To complete the transmission for a TG with k data packets, the sender usually needs to transmit a random number $M (M \geq k)$ of packets until (5) is satisfied. We define the bandwidth efficiency η by

$$\eta \triangleq \frac{k}{E\{M\}} \tag{6}$$

where $E\{M\}$ is the expectation of M. Clearly, we have $0 \leq \eta \leq 1$. Note that bandwidth efficiency is an important metric to evaluate the performance of multicast protocols. Since the RSE code has almost the highest loss-repairing efficiency for erasure channels (the RSE code is a type of maximum-distance separable (MDS) code [9]), the performance of a new FEC-based protocol (not RSE-code based) can be evaluated by comparing its η with that of the RSE code in terms of following criterions: 1) For reliable services, η should be close to η_{RS}, which is the bandwidth efficiency for RSE-code-based error-control schemes; 2) η does not decrease quickly when the packet-loss probability increases; 3) η does not decrease quickly when the number of receivers increases and thus the protocol has good scalability.

D.2. Average Number $E\{Q\}$ of TR's to Reach the Reliability QoS Requirement ξ

We denote the number of TR's to complete the transmission of a TG and its expectation by Q and $E\{Q\}$, respectively. Clearly, to obtain the feedbacks in each TR, the sender needs to wait at least a round-trip-time (RTT), which is the major contributor to the delay. Thus, the multicast protocol needs to keep a low $E\{Q\}$ to achieve the low delay. Also, a low $E\{Q\}$ represents a low overhead introduced to multicast services.

D.3. Average Delay QoS to Reach the Reliability–QoS Requirement ξ

The average delay, denoted by τ, to complete the transmission of a TG between the sender and the receivers is expressed by using (6) as

$$\tau = \frac{LE\{M\}}{B} + (\text{RTT})E\{Q\} = \frac{kL}{\eta B} + (\text{RTT})E\{Q\} \tag{7}$$

where L is the packet length (we assume fixed packet length throughout this paper), B is the bottleneck bandwidth among all receivers, and RTT is the maximum end-to-end RTT among all the sender–receiver pairs. From (7), our error-control scheme has two factors affecting the delay QoS. One is bandwidth efficiency η and the other is the total average number of TR's $E\{Q\}$. Either increasing η or decreasing $E\{Q\}$ will improve the delay QoS. However, increasing η may lead to a higher $E\{Q\}$. Thus, this introduces a tradeoff between η and $E\{Q\}$.

IV. 2-D Adaptive Error-Control Design Based on Graph Codes

Unlike RSE-based FEC multicast error control, where the sender only dynamically adjusts the code redundancy according to packet-loss levels while the coding scheme (RSE codes) stays the same, to further improve error-control efficiency and support the QoS diversity, we propose the 2-D graph-code-based multicast error-control schemes that regulate not only the code redundancy, but also the code structures, dynamically, based on different packet-loss levels fed back from multicast mobile receivers. This is motivated by our analyses of the graph-code-based schemes, which indicate that besides adapting error-control redundancy in each TR, the loss–repairing efficiency can also be significantly improved by using *nonuniformed* code mapping structures corresponding to different packet-loss levels. The key components and principles of our proposed 2-D adaptive graph-code-based scheme for providing QoS-driven mobile multicast services are detailed below in terms of *code mapping structure adaptation* and *error-control redundancy adaptation*, respectively.

In particular, for the transmission of each TG, the matrix $\mathbf{P}$ characterizing the graph code (see Section II) is composed of $(Q-1)$ submatrices denoted by $\mathbf{P}_1, \mathbf{P}_2, \ldots, \mathbf{P}_{Q-1}$, where $\mathbf{P} = [\mathbf{P}_1 \mathbf{P}_2 \cdots \mathbf{P}_{Q-1}]$. The submatrix $\mathbf{P}_{t-1}$ represents the mapping structure for the check packets generated in the t-th TR (the first TR is the data TR). In the t-th TR, $t \geq 2$, the sender dynamically generates $\mathbf{P}_{t-1}$ for loss repairing according to the packet-loss level $\theta_{\max}(t)$. Also, the error-control redundancy in the t-th TR (the number of check packets, or equivalently the number of columns of $\mathbf{P}_{t-1}$) is dynamically determined according to $\theta_{\max}(t)$. How to determine the mapping structure and the error-control redundancy in each TR will be elaborated on in Sections IV-A and IV-B, respectively.

A. Code Mapping Structure Adaptation

The construction of the mapping structure for one check packet (one column of $\mathbf{P}_{t-1}$) includes two parts. One is the selection of check-node degree (the numbers of 1's in each column of $\mathbf{P}_{t-1}$), denoted by γ. The other is the selection of which γ data packets are connected to the check packet (edge connection pattern). Consider one single receiver. We denote the packet-loss level by θ. Because losses are i.i.d. for different packets, then given the packet-loss level θ, the probabilities of occurrences for each loss pattern (loss pattern refers to which θ data packets are lost) are equal. Consequently, the probability of repairing one lost data packet by one single check packet does not depend on the edge connection pattern, but only on the check-node degree γ. Thus, we select the check-node degree and the edge connection pattern separately.

In this paper, we propose to use the *random mapping structure* for the construction of each check packet. In particular, for each check packet, we randomly choose γ distinct data packets and then connect them with this check packet in the bipartite graph. Note that each data packet is equally likely to be chosen. In addition, because the TR is the adaptation cycle, we let all check packets in a TR have the same error-correcting capability. That is, all check packets generated in the same TR have the same check-node degree. Also, we assume that the selections of edge connection patterns for different check packets are independent. The random mapping structure described above has the following characteristics. First, it is easy for implementation. Second, all check packets generated in the same TR have the same error-correcting capability. Third, the maximum error-control redundancy is virtually not upper-bounded. Moreover, by using the same random number generating algorithm and setting the same initial random number seed, both the sender and all receivers can construct exactly the same mapping structure in each TR based on the same control information, e.g., the packet-loss level. Thus, the sender needs to transmit only a small amount of control information instead of the entire mapping structure to all receivers.

Next, we discuss how to select the check-node degree in each TR to achieve high error-control efficiency. Note that in this section, the derived parameter selection algorithms are based on the single receiver case. However, these algorithms are also efficient for multiple receiver cases. Because the consolidated $\theta_{\max}(t)$ represents the highest packet-loss level among all receivers, thus the derived algorithms actually aim at efficiently improving the error-control efficiency for the receiver with the worst-case losses.

For the given check-node degree γ and packet-loss level θ, and m correctly received check packets, we derive the *average*

TABLE I
PARAMETERS AND METRICS TO EVALUATE REPAIRING EFFICIENCY

k	Size of a Transmission Group (TG).
θ	The number of lost/unrepaired data packets out of k data packets, also called packet-loss level.
γ	Check node degree.
ℓ	The number of data packets which are successfully repaired by m received check packets. $0 \leq \ell \leq \min\{m, \theta\}$.
$\psi_m(k,\theta,\gamma,\ell)$	Given k, θ and γ, the probability that total ℓ data packets are successfully repaired by m ($m \geq 1$) received check packets.
$N_m(k,\theta,\gamma)$	Given k, θ and γ, the average number of successfully repaired data packets by m ($m \geq 1$) received check packets, which is defined as loss-repairing efficiency.
$\gamma_m^*(k,\theta)$	Given the number m of received check packets and packet-loss level θ, the optimal check-node degree maximizing the loss-repairing efficiency.

number $N_m(k,\theta,\gamma)$ of successfully repaired data packets to represent the loss-repairing efficiency, which is expressed as

$$N_m(k,\theta,\gamma) = \sum_{\ell=1}^{\min\{\theta,m\}} \ell\psi_m(k,\theta,\gamma,\ell) \tag{8}$$

where $\psi_m(k,\theta,\gamma,\ell)$ is the probability that the total ℓ data packets are successfully repaired by m ($m \geq 1$) received check packets with given k, θ, and γ. All the related parameters are defined in Table I. Also, we define the *optimal check-node degree* by

$$\gamma_m^*(k,\theta) \stackrel{\triangle}{=} \arg\max_{1\leq\gamma\leq k} N_m(k,\theta,\gamma) \tag{9}$$

which maximize the average number of successfully repaired data packets.

A.1. Single Check Packet Case ($m = 1$)

Note that with a single check packet ($m = 1$), at most one lost data packet can be repaired. Thus, the loss-repairing efficiency becomes $N_1(k,\theta,\gamma)$, which actually equals the loss-repairing probability $\psi_1(k,\theta,\gamma,1)$. Theorem 1 introduced below derives the equations and criteria to determine the optimal check-node degree γ^* for any given code size k and the number θ of lost packets with the single ($m = 1$) check packet.

Theorem 1: If a graph code has k data packets in which θ data packets are lost randomly with i.i.d. distributions, then the following claims hold for $k \geq 1$ and $\theta = 1, 2, \cdots, k$.

Claim 1: The probability, denoted by $\psi_1(k,\theta,\gamma,1)$, that one ($\ell = 1$) lost packet can be repaired by one ($m = 1$) received parity-check packet with check-node degree γ is determined by

$$\psi_1(k,\theta,\gamma,1) = N_1(k,\theta,\gamma) = \begin{cases} \frac{\theta\gamma(k-\theta)!(k-\gamma)!}{(k-\gamma-\theta+1)!k!}, & \text{if } \gamma \leq k-\theta+1 \\ 0, & \text{if } \gamma > k-\theta+1. \end{cases} \tag{10}$$

Claim 2: For any given (k,θ) satisfying $k \geq 1$ and $1 \leq \theta \leq k$, there exists the maximum for $N_1(k,\theta,\gamma)$ as a function of γ, and the maximizer $\gamma_1^*(k,\theta)$ is determined by

$$\begin{aligned}\gamma_1^*(k,\theta) &\stackrel{\triangle}{=} \arg\max_{1\leq\gamma\leq k} N_1(k,\theta,\gamma) \\ &= \arg\max_{1\leq\gamma\leq k} \psi_1(k,\theta,\gamma,\ell)\Big|_{\ell=1} \\ &= \left\lceil \frac{(k+1)-\theta}{\theta} \right\rceil\end{aligned} \tag{11}$$

where $\lceil w \rceil$ denotes the least integer number that is larger than or equal to w.

Claim 3: The dynamics of $\psi_1(k,\theta,\gamma,1)$ is symmetric with respect to θ and γ such that $\psi_1(k,\theta,\gamma,1) = \psi_1(k,\gamma,\theta,1)$, and if and only if ($\theta = 1$, $\gamma_1^*(k,1) = k$) or ($\theta = k$, $\gamma_1^*(k,k) = 1$), $\psi_1(k,\theta,\gamma,1)$ attains its least upper bound $\psi_1^*(k,\theta,\gamma_1^*(k,\theta),1)$ determined by

$$\begin{aligned}&\psi_1^*\left(k,\theta,\gamma_1^*(k,\theta),1\right) \\ &\quad= \sup_{\substack{1\leq\theta\leq k\\1\leq\gamma\leq k}} \{\psi_1(k,\theta,\gamma,1)\} \\ &\quad= \psi_1\left(k,\theta,\gamma_1^*(k,\theta),1\right)\big|_{\left(\theta=1,\gamma_1^*(k,1)=k\right) \text{ or } \left(\theta=k,\gamma_1^*(k,k)=1\right)} \\ &\quad= 1.\end{aligned} \tag{12}$$

Proof: The detailed proof is provided in Appendix I. ■

Remarks on Theorem 1: Claim 1 derives general expressions for the loss-repairing probability/efficiency with a single check packet. Claim 2 states the existence and gives the closed-form expression of $\gamma_1^*(k,\theta)$. For any given (k,θ), a γ either much larger or much smaller than $\gamma_1^*(k,\theta)$ is undesired. This is expected since a γ much larger than $\gamma_1^*(k,\theta)$ can increase the cases of having *two or more than two* edges of the same check packet to be connected to the lost data packets, while a γ much smaller than $\gamma_1^*(k,\theta)$ can yield more cases where all edges of the check packet are only connected to the correctly received data packets. Equation (11) makes the critical observation that $\gamma_1^*(k,\theta)$ is generally a *nonlinear* decreasing function of the number θ of lost data packets. More importantly, (11) provides network designers with a closed-form analytical expression to calculate the optimal value $\gamma_1^*(k,\theta)$ of check-node degree according to the feedback of packet-loss level θ for any given graph code block size k. Claim 3 implies that variables θ and γ are functionally equivalent or exchangeable. This firmly supports the rationality of our random mapping structure. In addition, this claim derives the conditions when $\psi_1(k,\theta,\gamma,1)$ attains its globally absolute maximum. When $\theta = 1$, i.e., at most one data packet is lost for any multicast receivers, the optimal check-node degree satisfies $\gamma_1^*(k,\theta) = k$ based on Claim 2. Thus, the check packet actually is the modulo-2 addition of all the data packets (in this case, the code reduces to the well-known single parity check code [13, Ch. 3.8.1] and its loss-repairing probability attains its upper bound 1 according to Claim 3. It is clear that this mapping structure can repair the lost packet for any loss pattern with $\theta = 1$. Since this case corresponds to the possible last mapping structure to be selected for $\theta = 1$ immediately before all lost packets are repaired, we call this mapping structure the *final protocol*, which has the

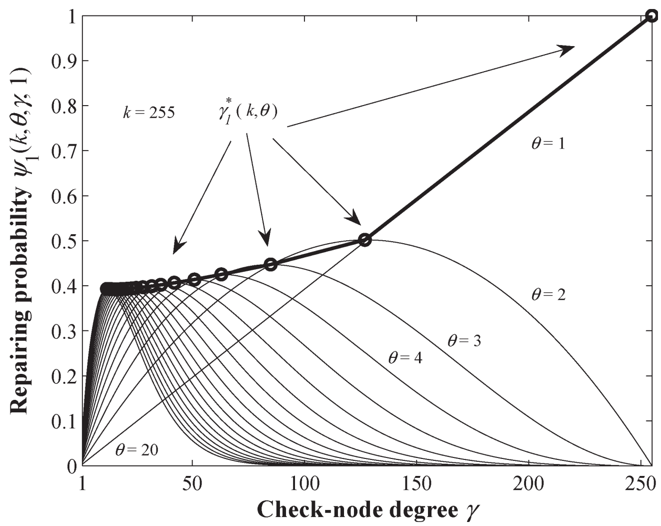


Fig. 2. Repairing probability $\psi_1(k, \theta, \gamma, 1)$ versus check-node degree γ. $\theta = 1, 2, \ldots, 20$ and $k = 255$.

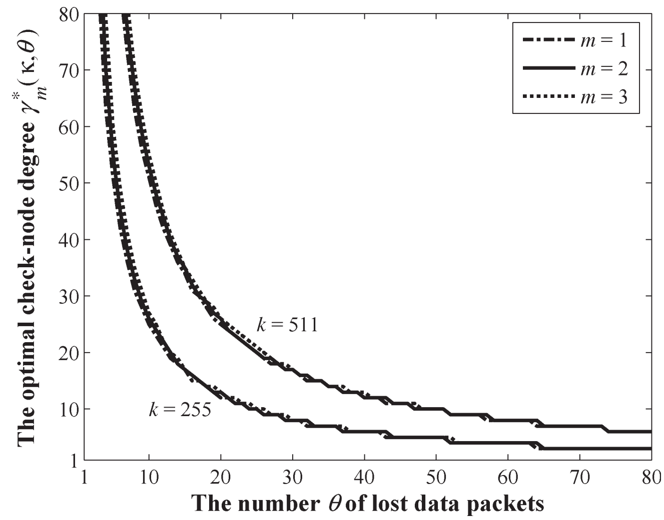


Fig. 4. Optimal check-node degree $\gamma_m^*(k, \theta)$ versus packet-loss level θ. $m = 1, 2, 3$ and $k = 255, 511$.

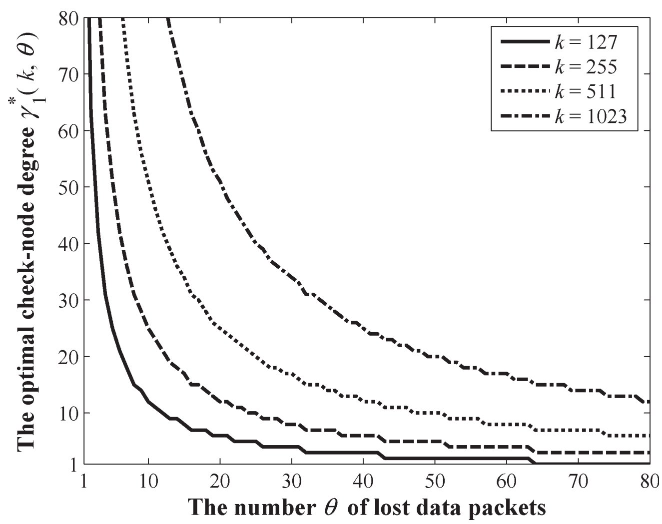


Fig. 3. Optimal check-node degree $\gamma_1^*(k, \theta)$ versus number θ of lost data packets. $k = 127, 255, 511, 1023$.

highest loss-repairing efficiency with a single check packet. Under this condition, the multicast system reaches a special state, where the sender only needs to keep on transmitting the check packet generated by the *final protocol* until all the lost data packets have been repaired. On the other hand, if $\theta = k$ (all data packets are lost), $\gamma_1^*(k, k) = 1$ should be selected to guarantee repairing one lost packet, in which the protocol effectively reduces to the retransmission protocol.

Fig. 2 numerically plots the loss-repairing probability $\psi_1(k, \theta, \gamma, 1)$ against check-node degree γ. We can see from Fig. 2 that for any given packet-loss level θ, there is an optimal $\gamma_1^*(k, \theta)$ that maximizes $\psi_1(k, \theta, \gamma, 1)$, as marked with a circle in Fig. 2, verifying Claim 2 of Theorem 1. Using (11), Fig. 3 plots the optimal check-node degree $\gamma_1^*(k, \theta)$ against packet losses θ with different code block sizes $k = 127, 255, 511, 1023$, which show that $\gamma_1^*(k, \theta)$ is a decreasing function of θ. So, we should select a small check-node degree if the packet-loss level is high and vice versa. Also, we observe that the smaller θ is, the faster the $\gamma_1^*(k, \theta)$ increases as θ decreases. All the above observations suggest that the *nonuniformed* code structures should be used to achieve high error-control efficiency. In addition, for any given θ, Fig. 3 shows that the larger the block size k, the higher the optimal check-node degree $\gamma_1^*(k, \theta)$. This is also expected since a large k implies that we need to have more repairing edges from the check nodes connected to the data packets to cover the lost data packets and vice versa.

A.2. Multiple Check Packet Case $(m > 1)$

In realistic systems, we usually need to send multiple check packets in each TR rather than a single check packet. However, the derivations of $N_m(k, \theta, \gamma)$ and $\gamma_m^*(k, \theta)$ become much more complicated as m increases. Then, we consider to use (11) to approximate $\gamma_m^*(k, \theta)$ for $m \geq 2$. To investigate the impact of m on the selection of $\gamma_m^*(k, \theta)$, we derive the analytical expressions of $\psi_m(k, \theta, \gamma, \ell)$ for $m = 2, 3$, which are summarized by (13) through (17) at the bottom of next page. Correspondingly, $N_m(k, \theta, \gamma)$ can be derived by using (8) and $\psi_m(k, \theta, \gamma, \ell)$ given in (13)–(17). Then, we get $\gamma_m^*(k, \theta)$ through (9). The detailed derivations of (13)–(17) are omitted due to lack of space, but are provided on-line in [14].

Fig. 4 plots the numerical results of $\gamma_m^*(k, \theta)$ against θ for $m = 1, 2, 3$. From Fig. 4, we observe that the three curves are very close to each other for all θ. This suggests that we can virtually use the results for the single check packet case to dynamically select the check-node degree for multiple check packet cases. Based on this consideration, we only use (11) to select the check-node degree in our proposed adaptive protocol.

B. Error-Control Redundancy Adaptation

After the check-node degree is selected in each TR, we need to determine an appropriate error-control redundancy (the number of check packets constructed and transmitted) in each TR based on the current packet-loss level θ. We denote the error-control redundancy in a TR by T. Consider the case where we select a very large T for the current TR. During the iterative decoding/repairing procedures in the current TR, the packet-loss level θ decreases gradually such that the selected $\gamma_1^*(k,\theta)$ cannot achieve near optimal loss-repairing probability with the changed θ. That is, if T is too large, the loss-repairing efficiencies of a majority of check packets received in the corresponding TR drop with the gradually decreasing packet-loss level. Consequently, more check packets are required because of the low loss-repairing efficiency, which severely degrades the bandwidth efficiency. If T is too small, although we can avoid the problems mentioned above, the improvement of the bandwidth efficiency is achieved at the cost of a higher Q, which may lead a long delay. Thus, we need to select a balanced T in each TR.

We develop a loss-covering strategy to determine T. For a given graph code, if a data node/packet is connected to one or more check nodes/packets, we say that this data node/packet is *covered*. In order for a lost data packet to be repaired, it must be covered. Under this principle, we develop the following *covering criterion* to obtain a balanced T with the given TG size k, check-node degree γ, and packet-loss level θ.

Covering Criterion: Using the random mapping structure, we let T in a TR equal the average number $T(k,\theta,\gamma)$ of check packets required to cover *at least one* lost data packet or, equivalently, to cover *at least* $(k-\theta+1)$ data packets.

Clearly, under the above covering criterion, the error-control redundancy $T(k,\theta,\gamma)$ is affected by both γ and θ. The following Theorem 2 derives the closed-form solution to $T(k,\theta,\gamma)$ for the above developed covering criterion.

Theorem 2: Using the *random mapping structure*, if the TG size is equal to k, the check-node degree is equal to γ, $1\le\gamma\le k$, and the packet-loss level is equal to θ, $1\le\theta\le k$, then the average number $T(k,\theta,\gamma)$ of check packets required to cover

$$\psi_2(k,\theta,\gamma,1) = \begin{cases} \theta\binom{k-\theta}{\gamma-1}\left(2\binom{k}{\gamma}+(1-2\theta)\binom{k-\theta}{\gamma-1}-2(\theta-1)\binom{k-\theta}{\gamma-2}\right)\Big/\binom{k}{\gamma}^2, & \text{if } 2\le\gamma\le k-\theta+1, \theta\ge 1\\ \left((1+2k)\theta-2\theta^2\right)/k^2, & \text{if } \gamma=1, \theta\ge 1\\ 0, & \text{otherwise}\end{cases} \tag{13}$$

$$\psi_2(k,\theta,\gamma,2) = \begin{cases} \frac{2(k-\theta+\gamma)}{k-\theta+1}\binom{\theta}{2}\binom{k-\theta}{\gamma-1}\binom{k-\theta+1}{\gamma-1}\Big/\binom{k}{\gamma}^2, & \text{if } 2\le\gamma\le k-\theta+1, \theta\ge 2\\ 2\binom{\theta}{2}/k^2, & \text{if } \gamma=1, \theta\ge 2\\ 0, & \text{otherwise}\end{cases} \tag{14}$$

$$\psi_3(k,\theta,\gamma,1) = \begin{cases} \theta\binom{k-\theta}{\gamma-1}\left(3\left(\binom{k}{\gamma}-\theta\binom{k-\theta}{\gamma-1}-(\theta-1)\binom{k-\theta}{\gamma-2}\right)\right. \\ \qquad\left.\cdot\left(\binom{k}{\gamma}-(\theta-1)\binom{k-\theta+1}{\gamma-1}\right)+\binom{k-\theta}{\gamma-1}^2\right)\Big/\binom{k}{\gamma}^3, & \text{if } 2\le\gamma\le k-\theta+1, \theta\ge 1\\ \theta\left(3(k-\theta)^2+3(k-\theta)+1\right)/k^3, & \text{if } \gamma=1, \theta\ge 1\\ 0, & \text{otherwise}\end{cases} \tag{15}$$

$$\psi_3(k,\theta,\gamma,2) = \begin{cases} 6\binom{\theta}{2}\binom{k-\theta}{\gamma-1}\left(\binom{k-\theta+1}{\gamma-1}\left(\frac{k-\theta+\gamma}{k-\theta+1}\left(\binom{k}{\gamma}-\theta\binom{k-\theta}{\gamma-1}-(2\theta-3)\binom{k-\theta}{\gamma-2}-(\theta-2)\binom{k-\theta}{\gamma-3}\right)\right.\right. \\ \qquad\left.\left.+\left(2-\frac{\gamma-1}{k-\theta+1}\right)\binom{k-\theta}{\gamma-2}\right)+\binom{k-\theta}{\gamma-1}^2\right)\Big/\binom{k}{\gamma}^3, & \text{if } 3\le\gamma\le k-\theta+1, \theta\ge 2\\ 6\binom{\theta}{2}\left((1-\theta)(k-\theta)^3+\left(\binom{k}{2}-4\theta+5\right)(k-\theta)^2\right. \\ \qquad\left.+\left(2\binom{k}{2}-4\theta+7\right)(k-\theta)\right)\Big/\binom{k}{2}^3, & \text{if } 2=\gamma\le k-\theta+1, \theta\ge 2\\ 6\binom{\theta}{2}(k-\theta+1)/k^3, & \text{if } \gamma=1, \theta\ge 2\\ 0, & \text{otherwise}\end{cases} \tag{16}$$

$$\psi_3(k,\theta,\gamma,3) = \begin{cases} 6\binom{\theta}{3}\binom{k-\theta}{\gamma-1}\left(\binom{k-\theta}{\gamma-1}^2+6\binom{k-\theta}{\gamma-1}\binom{k-\theta}{\gamma-2}+3\binom{k-\theta}{\gamma-1}\binom{k-\theta}{\gamma-3}\right. \\ \qquad\left.+9\binom{k-\theta}{\gamma-2}^2+6\binom{k-\theta}{\gamma-2}\binom{k-\theta}{\gamma-3}\right)\Big/\binom{k}{\gamma}^3, & \text{if } 3\le\gamma\le k-\theta+1, \theta\ge 3\\ 6\binom{\theta}{3}\left((k-\theta)^3+6(k-\theta)^2+9(k-\theta)\right)\Big/\binom{k}{2}^3, & \text{if } 2=\gamma\le k-\theta+1, \theta\ge 3\\ 6\binom{\theta}{3}/k^3, & \text{if } \gamma=1, \theta\ge 3\\ 0, & \text{otherwise}\end{cases} \tag{17}$$

at least one lost packet or, equivalently, to cover *at least* $(k-\theta+1)$ data packets, is given by

$$T(k,\theta,\gamma)=\begin{cases}1, & \text{if } \gamma \geq k-\theta+1;\\ h_0, & \text{if } \gamma < k-\theta+1,\end{cases} \tag{18}$$

where h_0 is determined by the following iterative equations:

$$\begin{cases} h_i=\frac{1}{1-\rho_{ii}}\left(1+\sum_{j=i+1}^{k-\theta+1}\rho_{ij}h_j\right), & \text{if } 0\leq i\leq k-\theta;\\ h_{k-\theta+1}=0, & \end{cases} \tag{19}$$

and ρ_{ij}, for $0 \leq i, j \leq k-\theta+1$, is given by

$$\rho_{ij}=\begin{cases}\binom{i}{\gamma-j+i}\binom{k-i}{j-i}/\binom{k}{\gamma}, & \text{if } 0\leq j-i\leq\gamma\leq j \text{ and } j<k-\theta+1;\\ \sum_{v=k-\theta+1}^{\min\{i+\gamma,k\}}\binom{i}{\gamma-v+i}\binom{k-i}{v-i}/\binom{k}{\gamma}, & \text{if } j=k-\theta+1 \text{ and } i+\gamma\geq j;\\ 0, & \text{otherwise.}\end{cases} \tag{20}$$

Proof: This theorem is proved by using the Markov Chain model as described in Appendix II. ■

Note that $\binom{u}{v}=u!/((u-v)!v!)$ for nonnegative integers u and v, $u \geq v \geq 0$. Also, $T(k,\theta,\gamma)$ may not be an integer. Then, we let $T=\lceil T(k,\theta,\gamma)\rceil$ to determine the error-control redundancy in each TR. Fig. 5 numerically plots the error-control redundancy $T(k,\theta,\gamma_1^*(k,\theta))$ in a TR against the packet-loss level θ. Through Fig. 5, we have the following observations. (i) The envelop of $T(k,\theta,\gamma_1^*(k,\theta))$ increases (decreases) with the increasing of packet-loss level θ when θ is relatively small (large). This is because packet-loss level θ and check-node degree $\gamma_1^*(k,\theta)$ jointly determine $T(k,\theta,\gamma_1^*(k,\theta))$. On the one hand, if θ becomes large, the check packets need to cover a smaller number $(k-\theta+1)$ of data packets such that fewer check packets can satisfy the covering criterion. On the other hand, a smaller $\gamma_1^*(k,\theta)$ is selected if θ becomes large. Consequently, each check packet covers fewer data packets and thus more check packets are required to satisfy the covering criterion. When θ is relatively small, $\gamma_1^*(k,\theta)$ decreases quickly (see Fig. 3) and then the change of $\gamma_1^*(k,\theta)$ dominates the variation of $T(k,\theta,\gamma_1^*(k,\theta))$. As a result, the envelop of $T(k,\theta,\gamma_1^*(k,\theta))$ increases as θ increases. In contrast, when θ is relatively large, $\gamma_1^*(k,\theta)$ decreases very slowly, then, the change of packet-loss level θ dominates the variation of $T(k,\theta,\gamma_1^*(k,\theta))$. Thus, the envelop of $T(k,\theta,\gamma_1^*(k,\theta))$ decreases as θ increases when θ is large. (ii) We observe that $T(k,\theta,\gamma_1^*(k,\theta))$ oscillates as θ increases, which is because of the followings. From (11), all the packet-loss levels can be divided into a number of regions resulted from the $\lceil\cdot\rceil$ operation, within each of which $\gamma_1^*(k,\theta)$ remains the same. Consequently, $T(k,\theta,\gamma_1^*(k,\theta))$ is a decreasing function of θ within each region because with more losses, we need fewer check packets to satisfy the covering criterion. However, because $\gamma_1^*(k,\theta)$ is the decreasing function of θ (see Fig. 3), the value of $\gamma_1^*(k,\theta)$ drops between the boundary points of two neighboring regions. Then, more check packets are required in a TR to satisfy the covering criterion because each check packet covers fewer data nodes. According to the above analyses, the covering criterion is jointly controlled by θ and γ such that we can achieve the balanced error-control redundancy.

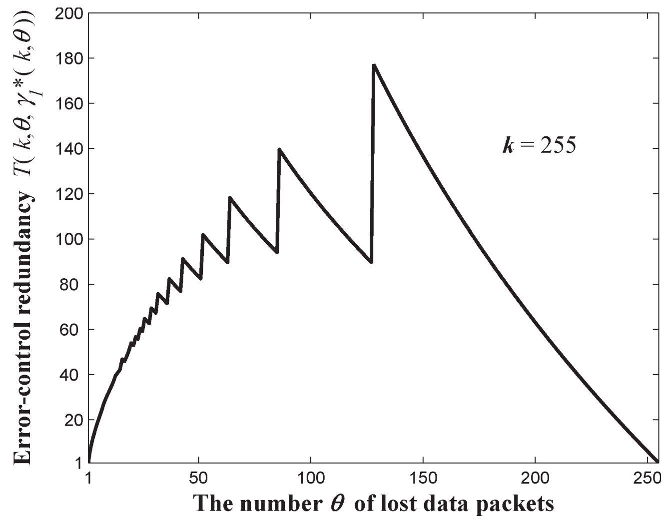


Fig. 5. Error-control redundancy $T(k,\theta,\gamma_1^*(k,\theta))$ in TR versus packet-loss level θ under covering criterion.

00. **Initial data transmission for a new TG**:
01. $t := 1$; Initialize random-number seed $state$;
 ! Both the sender and all receivers initialize $state$ to the same value.
02. Update $\mathbf{D}$; Multicast $\mathbf{D}$.
03. **On receipt of feedbacks of the t-th TR from all receivers**:
04. $\theta_{\max}(t) := \max_{r=1,2,\ldots,R}\{f_r(t)\}$;
05. **if** $(\theta_{\max}(t)/k \leq \xi)$ **goto** line-00; ! QoS requirement is satisfied.
06. **else** { $t := t+1$; ! Next TR.
07. $(\mathbf{P}_{t-1}, state) :=$ Construct($\theta_{\max}(t-1), , state$);
 ! Adaptively construct mapping structure matrix
08. $\mathbf{C}_t := \mathbf{D}\mathbf{P}_{t-1}$; Multicast $\theta_{\max}(t-1)$ and $\mathbf{C}_t$; ! Loss repairing.}

Fig. 6. Pseudo code for the sender.

00. **Initialization, on receipt of a new TG**:
01. $t := 1$; $\theta_{\max}(0) := k$; $f_r(0) := k$;
02. Initialize random-number seed $state$.
 ! Both the sender and all receivers initialize $state$ to the same value.
03. **On receipt of packets from the sender in the t-th TR**:
04. **if** $(f_r(t-1)/k \leq \xi)$ {Update $f_r(t) := f_r(t-1)$; **goto** line-13; }
 ! Required reliability-QoS is satisfied
05. **if** $(t=1)$ {Save correctly received data packets; Update $f_r(t)$;}
 ! Initial data transmission
06. **else** { ! Receipt of check packets
07. Save $\theta_{\max}(t-1)$ and correctly received check packets;
08. $(\mathbf{P}_{t-1}, state) :=$ Construct($\theta_{\max}(t-1), state$);
 ! Adaptively construct the mapping structure matrix
09. **if** $(t=2)$ $\mathbf{P} := \mathbf{P}_1$;
10. **else** $\mathbf{P} := [\mathbf{P} \quad \mathbf{P}_{t-1}]$;
11. Decode based on $\mathbf{P}$ and all correctly received packets;
12. Update $f_r(t)$; }
13. Feed back $f_r(t)$ to the sender; $t := t+1$;

Fig. 7. Pseudo code for the rth receiver

C. Adaptive Graph-Code-Based Hybrid ARQ-FEC Protocol for Error-Control of Mobile Multicast

We describe our proposed adaptive two-dimensional hybrid ARQ-FEC protocol for error control of multicast by using the pseudo codes presented in Figs. 6–8. The variables used in

```
Function Construct(θ_max(t − 1), state);
  if (θ_max(t − 1) := 1) {P_{t−1} := (1, 1, ..., 1)^τ; T := 1;
    γ := k; }   ! Final protocol, P_{t−1} is a k × 1 column vector
  else {
    Set γ := γ_1^*(k, θ_max(t − 1)) by using (11);
    T := ⌈T(k, θ_max(t − 1), γ)⌉ by using (18)-(20);
          ! Select check-node degree and error-control redundancy
    Randomly build P_{t−1} with γ, T, and state; Update state; }
  return (P_{t−1}, state); }
```

Fig. 8. Pseudo code for the mapping structure construction function

TABLE II
VARIABLES USED IN PSEUDO CODES

k	The number of data packets in each TG.
γ	Check-node degree which is dynamically adjusted in each TR.
T	Error-control redundancy which is dynamically adjusted in each TR.
ξ	Reliability-QoS requirement. See (3).
$\mathbf{D}$	$L \times k$ matrix denoting data packets of a TG. See Section II.
$\mathbf{C}_t$	$L \times T$ matrix denoting the check packets generated and multicast by the sender in the t-th TR. See Section II.
$\mathbf{P}_{t-1}$	$k \times T$ mapping structure matrix, which is used to generate $\mathbf{C}_t$ in the t-th TR, where $\mathbf{C}_t = \mathbf{D}\mathbf{P}_{t-1}$.
$state$	The random-number seed which is used to randomly construct mapping structures. It is initialized with the same value in the sender and all receivers.
$\mathbf{P}$	$\mathbf{P} = [\mathbf{P}_1 \ \ \mathbf{P}_2 \ \ \cdots \ \ \mathbf{P}_{t-1}]$ for the t-th TR, $t \geq 2$.

pseudo codes are defined in Table II. We explain the pseudo codes as follows.

1) Protocol for the sender:
The sender multicasts a data TG $\mathbf{D}$ in the first TR. Then, the sender waits for feedbacks $f_r(t)$ from all receivers, where $r = 1, 2, \ldots, R$. After having received all feedbacks, the sender gets the maximum number of lost data packets $\theta_{\max}(t)$. If $\theta_{\max}(t)/k \leq \xi$, the reliability-QoS requirement is satisfied and the sender starts to multicast the next new TG. If $\theta_{\max}(t)/k > \xi$, the sender needs to execute loss-repairing procedures in the next TR. Set $t := t + 1$. The sender constructs the mapping structure $\mathbf{P}_{t-1}$ in the t-th TR according to packet-loss level $\theta_{\max}(t - 1)$. After that, the sender multicasts $\mathbf{C}_t = \mathbf{D}\mathbf{P}_{t-1}$ and $\theta_{\max}(t - 1)$ to all receivers. Then, the sender goes into the state waiting for feedbacks.

2) Protocol for the rth receiver where $r = 1, 2, \ldots, R$:
The rth receiver receives a data TG $\mathbf{D}$ in the first TR. Then, the rth receiver calculates $f_r(t)$, feeds it back to the sender, and set $t := t + 1$. On condition that $\theta_{\max}(t - 1)/k > \xi$, the rth receiver will receive $\theta_{\max}(t - 1)$ and a number of check packets in the current t-th TR. If the reliability-QoS requirement for the rth receiver is already satisfied, i.e., $f_r(t - 1)/k \leq \xi$, the rth receiver will ignore the received packets, simply set $f_r(t) := f_r(t - 1)$ and feed $f_r(t)$ back. If the reliability-QoS requirement is not satisfied, i.e., $f_r(t - 1)/k > \xi$, the rth receiver will construct the mapping structure $\mathbf{P}_{t-1}$ for the current TR and start the iterative decoding (repairing) procedures. Note that although $\mathbf{P}_{t-1}$ is constructed according to the packet-loss level in the $(t - 1)$-th TR, the decoding is performed based on the all $\mathbf{P}_u$, $u = 1, 2, \ldots, t - 1$, and all packets correctly received for the current TG to fully make use of the received redundancy. After the repairing procedure, the rth receiver feeds the updated $f_r(t)$ back to the sender. Having sent the feedback information $f_r(t)$, the rth receiver sets $t := t + 1$ and goes into the state waiting for new packets from the sender.

3) Protocol for the mapping structure construction function:
In the data TR, no mapping structure will be constructed. In loss-repairing TR's, if $\theta_{\max}(t - 1) = 1$, the *final protocol* will be selected. If $\theta_{\max}(t - 1) > 1$, the check-node degree γ and error-control redundancy T are selected based on (11) and (18)–(20). By using the same random number generating algorithm, the sender and receivers construct the corresponding mapping structures for the t-th TR with the selected parameters γ and T. Note that the sender and all receiver initialize the random-number seed state to the same value in the first TR as described in Figs. 6 and 7. Also, as assumed in Section III-A, the packet-loss level $\theta_{\max}(t - 1)$ can be reliably transmitted between the sender and receivers in each TR. Thus, the sender and all receivers can always get the same parameters T and γ in each TR and then construct the exactly same mapping structure.

V. PERFORMANCE EVALUATIONS

Using simulations, we evaluate the performance of the proposed adaptive graph-code-based multicast protocol for mobile multicast services. We also compare the performances of our proposed adaptive protocol with those using the RSE code, the non-adaptive graph code (also using random mapping-structure), and the pure ARQ-based approach. The TG size k is set to 255. For the RSE-based schemes, the sender sends $\theta_{\max}(t)$ check packets in each repairing TR. We simulate two (509, 255) and (291, 255) RSE codes with symbol size of 10 bits, the corresponding code rates of which are 0.501 and 0.876, respectively. Note that the two RSE codes can support a maximum of 254 and 36 check packets, respectively. For the nonadaptive graph-code-based schemes, the sender uses the constant γ and T in each repairing TR. We simulate two sets of parameters: ($\gamma = 7$, $T = 74$) and ($\gamma = 15$, $T = 47$). In the simulation, we consider the packet-loss probability p equal to 0.001 through 0.1, which typically covers a wide range of channel quality for mobile wireless networks.

Fig. 9 compares the bandwidth efficiency for reliable services ($\xi = 0$) under different packet-loss probabilities. As shown in Fig. 9, our proposed adaptive scheme can gain at least 10% higher bandwidth efficiency than those using nonadaptive graph codes. Moreover, for low packet-loss probability, the bandwidth efficiency of our adaptive scheme is very close to that of the RSE-based schemes. Under high packet-loss probability, RSE codes with high code rate (e.g., 0.876) cannot provide enough error-control redundancy and thus lead to decoding failure, retransmission, and very low η. In contrast, our proposed adaptive scheme can support sufficient error-control redundancy to avoid these problems by using the random mapping structure for graph codes. Fig. 10 shows that the bandwidth efficiency of our proposed scheme is not sensitive to the increasing of the number R of receivers. This indicates that our adaptive scheme has good scalability. Fig. 11 gives the average number $E\{Q\}$

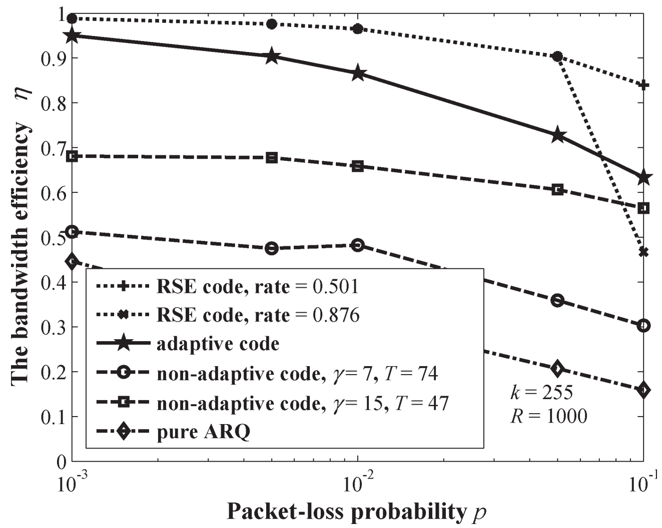


Fig. 9. Bandwidth efficiency η versus packet-loss probability p for reliable services.

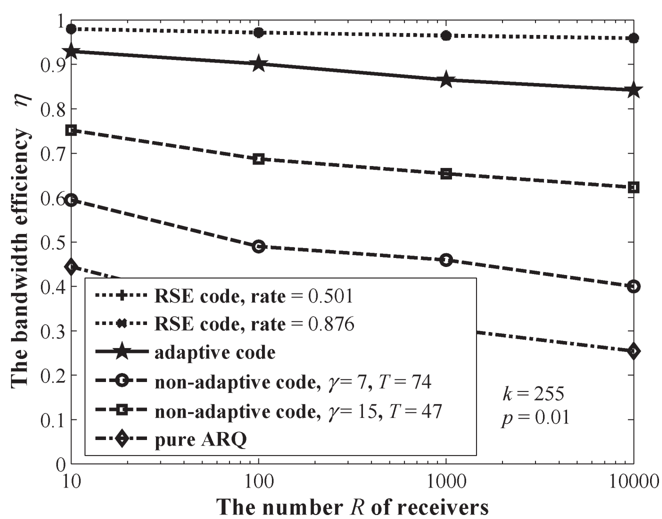


Fig. 10. Bandwidth efficiency η with different numbers R of receivers for reliable services.

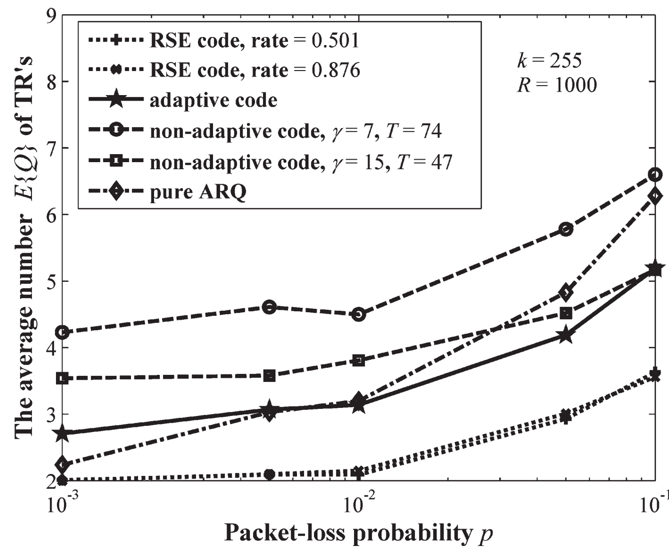


Fig. 11. Average number $E\{Q\}$ of TR's versus packet-loss probability p for reliable services.

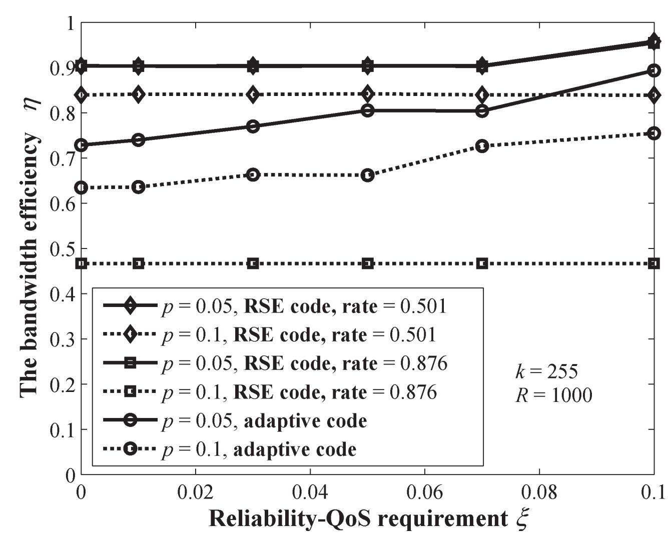


Fig. 12. Bandwidth efficiency η versus the reliability–QoS requirement ξ. $k = 255$, and $p = 0.05$ and 0.1.

of TR's to complete the transmission of a TG for each scheme. We can see that the $E\{Q\}$ of our proposed adaptive scheme is relatively low as compared to the schemes using the non-adaptive code and the pure ARQ-based approach. This implies that the adaptive scheme imposes a relatively low overhead to multicast services.

Fig. 12 compares the bandwidth efficiency for various schemes under different reliability–QoS requirements ξ from 0.0 to 0.1. For our proposed adaptive scheme, receivers can dynamically update packet loss status in each TR because the iterative decoding procedure can be executed as long as any number of check packets is received. Thus, for different reliability–QoS requirements, our proposed adaptive scheme can efficiently avoid unnecessary repairing packet transmission for perfect reliability. As shown in Fig. 12, when the reliability–QoS requirement ξ becomes larger (more losses are tolerated), the bandwidth efficiency of our adaptive scheme improves significantly. Clearly, because the decoding of RSE codes can be performed only after k or more distinct data/check packets have been correctly received, RSE codes cannot further improve the bandwidth efficiency η when the reliability–QoS requirements ξ increases.

Fig. 13 shows the average number $E\{Q\}$ of TRs with different ξ. We can see that our proposed adaptive scheme will have lower $E\{Q\}$ than those RSE-based schemes when ξ is high. Fig. 14 illustrates the comprehensive effect of the reliability–QoS requirement on the average delay. In the simulation, we assume that packet length $L = 1000$ bits, bandwidth $B = 1$ Mb/s, and maximum RTT among all sender–receiver pairs equal 80 ms. Clearly, with the same channel quality, our proposed adaptive scheme can achieve a much lower average delay than those of the RSE-based schemes for relatively higher ξ. So, we can observe that although RSE codes have the best erasure-correcting capability, its inflexible structure and high complexity severely limit its applicability to QoS-driven mobile multicast services. By contrast, our proposed adaptive scheme can flexibly and dynamically adjust

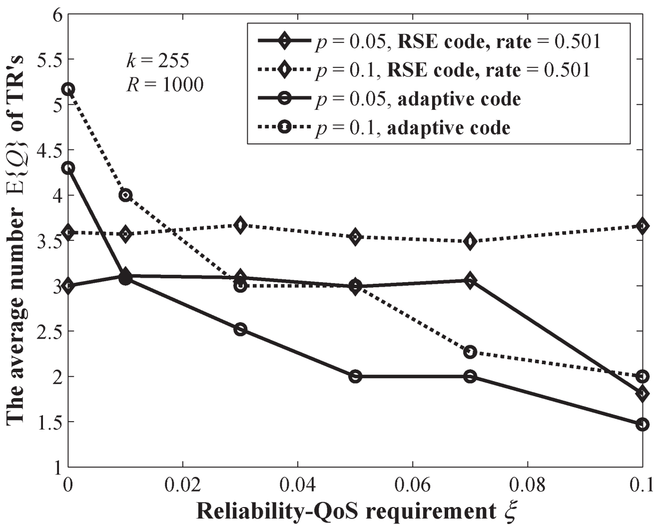


Fig. 13. Average number $E\{Q\}$ of TR's versus the reliability–QoS requirement ξ. $k = 255$, and $p = 0.05$ and 0.1.

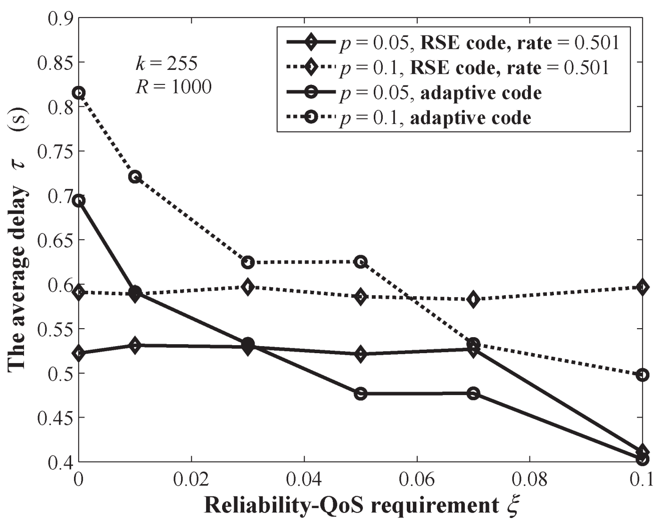


Fig. 14. Average delay τ for transmission of TG versus reliability–QoS requirement ξ. $k = 255$, and $p = 0.05$ and 0.1. $B = 1$ Mb/s, $L = 1000$ bits, RTT $= 80$ ms.

the coding structures to achieve high error-control efficiency for highly diverse QoS requirements.

VI. Conclusion

To provide flexible and efficient error-control schemes for QoS diverse multicast services, we developed and analyzed an adaptive hybrid ARQ–FEC graph-code-based erasure-correcting protocol for QoS-driven multicast services over mobile wireless networks. The key features of our proposed scheme are twofold: the low complexity and dynamic adaptation to packet-loss levels. The low complexity is achieved by using the graph code. In addition, the accumulatively iterative decoding procedures of graph codes can flexibly adapt to the variations of reliability–QoS requirements of different mobile services. To increase the error-control efficiency, we developed a 2-D adaptive error-control scheme, which dynamically adjusts both the error-control redundancy and the code-mapping structures. By deriving and identifying the closed-form nonlinear analytical expression between the optimal check-node degree and the packet-loss level for any given code block length, we proposed the nonuniformed adaptive coding structures to achieve high error-control efficiency. Furthermore, by developing the loss covering strategy and applying Markov-chain modeling techniques, we derive the closed-form expressions of error-control redundancy as a function of the packet-loss level and the optimal check-node degree in each TR. Using the proposed nonuniformed adaptive error-control scheme, we developed an efficient hybrid ARQ–FEC protocol employing adaptive graph codes for mobile multicast services. We evaluated the proposed protocol through simulation experiments. The simulation results show that our scheme can achieve high error-control efficiency for QoS-driven multicast services while significantly reducing computational complexity and implementation overhead.

Appendix I
Proof of Theorem 1

Proof: Because losses for different data packets are i.i.d., we can express $\psi_1(k,\theta,\gamma,1)$ as

$$\psi_1(k,\theta,\gamma,1) = \frac{\lambda}{\Lambda} \tag{21}$$

where λ is the number of loss patterns under which one of the lost data packets can be repaired by a single check packet with check-node degree γ, and Λ is the total number of loss patterns.

As described in Section II, in order to repair one lost data packet with one check packet for a loss pattern, the following conditions must be satisfied. 1) Among γ data packets that are connected with the same check packet, there is only one lost data packet. If $\gamma > k-\theta+1$, at least two data packets are connected with the check packet and then no losses can be repaired. 2) Among $(k-\gamma)$ data packets that are not connected with the check packet, $(\theta-1)$ of them are other lost data packets. Thus, we derive λ as follows:

$$\lambda = \begin{cases} \binom{\gamma}{1}\binom{k-\gamma}{\theta-1}, & \text{if } \gamma \le k-\theta+1 \\ 0, & \text{if } \gamma > k-\theta+1. \end{cases} \tag{22}$$

Also, by the above definition of Λ we have $\Lambda = \binom{k}{\theta}$. Then, using (21) we can obtain

$$\psi_1(k,\theta,\gamma,1) = \begin{cases} \frac{\theta\gamma(k-\theta)!(k-\gamma)!}{(k-\gamma-\theta+1)!k!}, & \text{if } \gamma \le k-\theta+1 \\ 0, & \text{if } \gamma > k-\theta+1 \end{cases} \tag{23}$$

which completes the proof of Claim 1. For $1 \le \gamma \le k-1$, we define

$$\Delta(\gamma) \stackrel{\triangle}{=} \psi_1(k,\theta,\gamma+1,1) - \psi_1(k,\theta,\gamma,1). \tag{24}$$

Plugging (23) into (24) and letting $\Delta(\gamma) \le 0$, we derive

$$\begin{aligned} \Delta(\gamma) \le 0 \quad &\Leftrightarrow \quad (k-\theta\gamma-\theta+1) \le 0 \\ &\qquad \text{or } k-\theta+1 \le \gamma \le k-1 \\ &\Leftrightarrow \quad \left\lceil \frac{(k+1)-\theta}{\theta} \right\rceil \le \gamma \le k-1. \end{aligned} \tag{25}$$

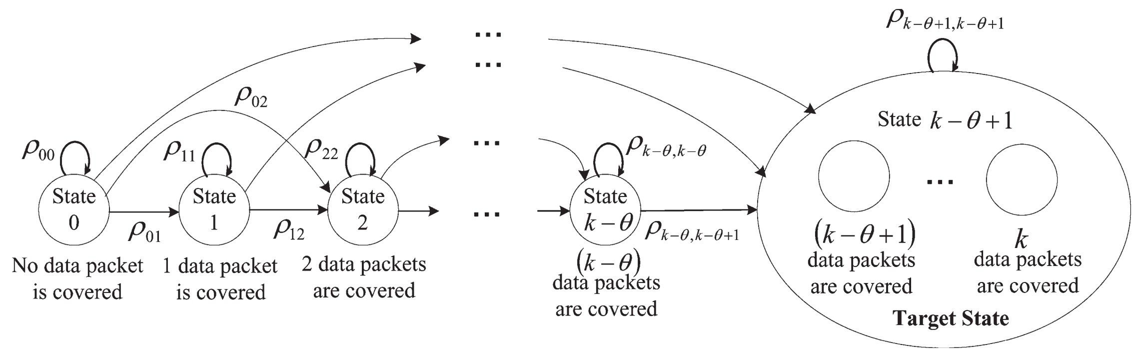


Fig. 15. State transition diagram of Markov chain for covering status.

Thus, the following inequalities hold:

$$\begin{aligned}&\psi_1(k,\theta,1,1)\\&\quad\le\psi_1(k,\theta,2,1)\le\cdots\\&\quad\le\psi_1\left(k,\theta,\left\lceil\frac{(k+1)-\theta}{\theta}\right\rceil,1\right)\end{aligned}\tag{26}$$

$$\begin{aligned}&\psi_1\left(k,\theta,\left\lceil\frac{(k+1)-\theta}{\theta}\right\rceil,1\right)\\&\quad\ge\psi_1\left(k,\theta,\left\lceil\frac{(k+1)-\theta}{\theta}\right\rceil+1,1\right)\\&\quad\ge\cdots\ge\psi_1(k,\theta,k,1).\end{aligned}\tag{27}$$

Therefore, $\gamma=\lceil((k+1)-\theta)/\theta\rceil$ maximizes $\psi_1(k,\gamma,\theta,1)$ with the given θ. Then, $\gamma_1^*(k,\theta)$ is given by

$$\gamma_1^*(k,\theta)=\arg\max_{1\le\gamma\le k}\psi_1(k,\theta,\gamma,1)=\left\lceil\frac{(k+1)-\theta}{\theta}\right\rceil\tag{28}$$

which completes the proof of Claim 2.

Note that $\gamma>k-\theta+1\Leftrightarrow\theta>k-\gamma+1$. Thus, we have

$$\begin{aligned}\psi_1(k,\theta,\gamma,1)&=\begin{cases}\frac{\theta\gamma(k-\theta)!(k-\gamma)!}{(k-\gamma-\theta+1)!k!}, & \text{if }\gamma\le k-\theta+1\\0, & \text{if }\gamma>k-\theta+1\end{cases}\\&=\begin{cases}\frac{\gamma\theta(k-\gamma)!(k-\theta)!}{(k-\theta-\gamma+1)!k!}, & \text{if }\theta\le k-\gamma+1\\0, & \text{if }\theta>k-\gamma+1\end{cases}\\&=\psi_1(k,\gamma,\theta,1).\end{aligned}\tag{29}$$

This proves that the dynamics of $\psi_1(k,\theta,\gamma,1)$ is symmetric with respect to θ and γ. Note that we have $\psi_1(k,\theta,\gamma,1)\le 1$. Next, we solve the equation $\psi_1(k,\theta,\gamma,1)=1$ for $\gamma\le k-\theta+1$ to see whether some (θ,γ) achieves the upper bound 1. Equivalently, we need to solve $\binom{\gamma}{1}\binom{k-\gamma}{\theta-1}=\binom{k}{\theta}$. Also, because $\binom{k}{\theta}=\sum_{i=0}^{\min\{\gamma,\theta\}}\binom{\gamma}{i}\binom{k-\gamma}{\theta-i}$ holds for any $1\le\gamma\le k$, we need to guarantee $\min\{\gamma,\theta\}=1$ for $\binom{k}{\theta}=\binom{\gamma}{1}\binom{k-\gamma}{\theta-1}$. Thus, either $\theta=1$ or $\gamma=1$ must be satisfied. Plugging $\theta=1$ and $\gamma=1$ into (10), respectively, we obtain $\psi_1(k,1,k,1)=1$ and $\psi_1(k,k,1,1)=1$. Thus, 1 is the least upper bound of $\psi_1(k,\theta,\gamma,1)$. Moreover, through (11), we have $\gamma_1^*(k,1)=k$ and $\gamma_1^*(k,k)=1$. Then, the proof of Claim 3 is completed. ■

APPENDIX II
PROOF OF THEOREM 2

Proof: We model the loss-covering procedure by a random process $\{X_n\}$ taking value in the state space specified by $\{0,1,2\cdots,k-\theta+1\}$, as shown in Fig. 15, which describes the loss-covering states of the data packets. State $i, 0\le i<k-\theta+1$, represents the total number i out of k data packets having been covered. State i, $i=(k-\theta+1)$, represents the target state, where at least $(k-\theta+1)$ data packets or, equivalently, at least one lost data packet has been covered by the generated check packets. We call the data packets that have not been covered the *uncovered* data packets.

The random variable X_n denotes the covering state after the nth check packet of the current TR has been generated. If $X_{n_0-1}<k-\theta+1$ and $X_{n_0}=k-\theta+1$ for some n_0, we say that we reach the target state after n_0 check packets have been generated. It is clear that the number $T(k,\theta,\gamma)$ described in (18) is equal to $E\{n_0\}$.

Next, we show that $\{X_n\}$, $n\ge 0$, is a Markov chain. Note that if $X_n=i_n$, $(k-i_n)$ equals the number of data packets that have not been covered after the nth data packet of the current TR has been generated. Then, we have

$$\begin{aligned}&\Pr\{X_{n+1}=i_{n+1}|X_n=i_n,X_{n-1}=i_{n-1},\cdots,X_0=i_0\}\\&\;=\Pr\{X_{n+1}-X_n=i_{n+1}-i_n|X_n=i_n\\&\qquad X_{n-1}=i_{n-1},\cdots,X_0=i_0\}\\&\overset{(a)}{=}\Pr\{(i_{n+1}-i_n)\text{ out of }(k-i_n)\text{ data packets }\textit{uncovered}\\&\qquad\text{by the previous }n\text{ check packets are covered by}\\&\qquad\text{the }(n+1)\text{th check packet}|X_n=i_n,\\&\qquad X_{n-1}=i_{n-1},\cdots,X_0=i_0\}.\end{aligned}\tag{30}$$

Also, as described in Section III-B, the random construction of the check packet is independent of the constructions of other check packets. So, if X_n is given, the conditional probability in (a) of (30) is independent of $X_{n-1},X_{n-2},\ldots,X_0$. Thus, we can derive

$$\begin{aligned}&\Pr\{X_{n+1}=i_{n+1}|X_n=i_n,X_{n-1}=i_{n-1},\cdots,X_0=i_0\}\\&\;=\Pr\{(i_{n+1}-i_n)\text{ out of }(k-i_n)\\&\qquad\text{data packets }\textit{uncovered}\text{ by}\\&\qquad\text{the previous }n\text{ check packets are covered by}\\&\qquad\text{the }(n+1)\text{th check packet}|X_n=i_n\}\\&\;=\Pr\{X_{n+1}-X_n=i_{n+1}-i_n|X_n=i_n\}\\&\;=\Pr\{X_{n+1}=i_{n+1}|X_n=i_n\}.\end{aligned}\tag{31}$$

Therefore, $\{X_n\}$, $n \geq 0$, is a Markov chain. Clearly, the Markov chain is homogeneous in terms of n. We define the transition probability, denoted by ρ_{ij}, as

$$\rho_{ij} \triangleq \Pr\{X_{n+1} = j | X_n = i\}, \quad n \geq 0, 0 \leq i, j \leq k - \theta + 1. \tag{32}$$

For convenience, we rewrite (20) again as

$$\rho_{ij} = \begin{cases} \binom{i}{\gamma-j+i}\binom{k-i}{j-i}/\binom{k}{\gamma}, & \text{if } 0 \leq j-i \leq \gamma \leq j \\ & \text{and } j < k-\theta+1 \\ \sum_{v=k-\theta+1}^{\min\{i+\gamma,k\}} \frac{\binom{i}{\gamma-v+i}\binom{k-i}{v-i}}{\binom{k}{\gamma}}, & \text{if } j = k-\theta+1 \\ & \text{and } i+\gamma \geq k-\theta+1 \\ 0, & \text{otherwise.} \end{cases}$$

Note that $\binom{i}{\gamma-j+i}\binom{k-i}{j-i}$ is the number of ways of constructing the check packet such that $X_{n+1} = j$ with given $X_n = i$, while $\binom{k}{\gamma}$ is the total number of ways constructing a check packet. Hence, ρ_{ij} is equal to the ratio of $\binom{i}{\gamma-j+i}\binom{k-i}{j-i}$ to $\binom{k}{\gamma}$, which is shown in the first part of (20). The condition $0 \leq j - i \leq \gamma \leq j$ is obtained by solving $i \geq \gamma - j + i \geq 0$ and $k - i \geq j - i \geq 0$ such that the expressions of $\binom{i}{\gamma-j+i}$ and $\binom{k-i}{j-i}$ are meaningful. For the special case $j = k - \theta + 1$, ρ_{ij} is derived as

$$\begin{aligned} &\rho_{i,k-\theta+1} \\ &= \Pr\{X_{n+1} = k-\theta+1 | X_n = i\} \\ &= \Pr\{\text{At least } (k-\theta+1-i) \text{ out of } (k-i) \\ &\qquad \textit{uncovered} \text{ data packets are covered by the} \\ &\qquad (n+1)\text{th check packet} | X_n = i\} \\ &= \sum_{v=k-\theta+1}^{\min\{i+\gamma,k\}} \Pr\{(v-i) \text{ out of } (k-i) \textit{ uncovered} \\ &\qquad \text{data packets are covered by the} \\ &\qquad (n+1)\text{th check packet} | X_n = i\} \\ &= \sum_{v=k-\theta+1}^{\min\{i+\gamma,k\}} \binom{i}{\gamma-v+i}\binom{k-i}{v-i} \Big/ \binom{k}{\gamma}. \end{aligned} \tag{33}$$

It is clear that when the conditions of the first two parts in (20) are not satisfied, the covering state cannot transfer from state i to state j with only one new check packet, and thus we get $\rho_{ij} = 0$.

Then, the probability–transition matrix, expressed by a $(k - \theta + 2) \times (k - \theta + 2)$ square matrix $\boldsymbol{\rho}$, is determined by

$$\boldsymbol{\rho} = \begin{bmatrix} \rho_{00} & \rho_{01} & \cdots & \rho_{0,k-\theta+1} \\ 0 & \rho_{11} & \cdots & \rho_{1,k-\theta+1} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}. \tag{34}$$

Note that $\boldsymbol{\rho}$ is an upper triangular matrix because X_n is an increasing sequence in terms of n.

We define a set of variables h_i, $0 \leq i \leq k - \theta + 1$: if the current covering state is i (equivalently, we have covered i data packets), on average, the sender needs other h_i check packets to reach the target covering state $(k - \theta + 1)$ (equivalently, we have covered at least $(k - \theta + 1)$ data packets). Then, h_i for $0 \leq i \leq k - \theta + 1$ is expressed as

$$h_i \triangleq E\{j | X_n = i, X_{n+j} = k-\theta+1, j \geq 0, n \geq 0\}. \tag{35}$$

Clearly, we have $h_{k-\theta+1} = 0$ and $T(k, \theta, \gamma) = h_0$. If $\gamma \geq k - \theta + 1$, it is clear that we need only one check packet to satisfy the covering criterion. Thus, we obtain (18).

We define $\mathbf{h} = (h_0, h_1, \ldots, h_{k-\theta+1})^\tau$, where $(\cdot)^\tau$ denotes the matrix transpose operator. $\mathbf{h}$ is the solution to the linear equations [12]

$$\begin{cases} \mathbf{h} = \mathbf{z} + \boldsymbol{\rho}\mathbf{h} \\ h_{k-\theta+1} = 0 \end{cases} \tag{36}$$

where $\mathbf{z}$ is a $(k - \theta + 2)$-dimension column vector $(1, 1, \ldots, 1, 0)^\tau$.

As shown in (34), $\boldsymbol{\rho}$ is an upper triangular matrix. Hence, we can get the solution to $\mathbf{h}$ by the iterative equations

$$\begin{cases} h_i = \frac{1}{1-\rho_{ii}} \left(1 + \sum_{j=i+1}^{k-\theta+1} \rho_{ij} h_j\right), & i = 0, 1, 2, \ldots, k-\theta \\ h_{k-\theta+1} = 0 \end{cases}$$

which complete the proof of (19), and thus Theorem 2 follows. ∎

References

[1] H. Forman and J. Zahorjan, "The challenges of mobile computing," *IEEE Comput.*, vol. 27, no. 4, pp. 38–47, Apr. 1994.

[2] X. Zhang and K. G. Shin, "Markov-chain modeling for multicast signaling delay analysis," *IEEE/ACM Trans. Networking*, vol. 12, no. 4, pp. 667–680, Aug. 2004.

[3] X. Zhang, K. G. Shin, D. Saha, and D. Kandlur, "Scalable flow control for multicast ABR services in ATM networks," *IEEE/ACM Trans. Networking*, vol. 10, no. 1, pp. 67–85, Feb. 2002.

[4] X. Zhang and K. G. Shin, "Delay analysis of feedback-synchronization signaling for multicast flow control," *IEEE/ACM Trans. Networking*, vol. 11, no. 3, pp. 436–460, Jun. 2003.

[5] S. Floyd, V. Jacobson, C.-G. Liu, S. McCanne, and L. Zhang, "A reliable multicast framework for light-weight sessions and application level framing," *IEEE/ACM Trans. Netw.*, vol. 5, no. 6, pp. 784–802, Dec. 1997.

[6] C. Huitema, "The case for packet level FEC," in *Proc. IFIP 5th Int. Workshop Protocols High-Speed Netw.*, Sophia Antipolis, France, Oct. 1996, pp. 109–120.

[7] J. Nonenmacher, E. Biersack, and D. Towsley, "Parity-based loss recovery for reliable multicast transmission," *IEEE/ACM Trans. Networking*, vol. 6, no. 4, pp. 349–361, Aug. 1998.

[8] N. Nikaein, H. Labiod, and C. Bonnet, "MA-FEC: A QoS-based adaptive FEC for multicast communication in wireless networks," in *Proc. IEEE Int. Conf. Commun.*, 2000, pp. 954–958.

[9] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. A. Spielman, "Efficient erasure correcting codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 569–584, Feb. 2001.

[10] M. Luby, "LT codes," in *Proc. IEEE 43rd Annu. Symp. Found. Comput. Sci.*, Nov. 2002, pp. 271–280.

[11] J. W. Byers, M. Luby, and M. Mitzenmacher, "A digital fountain approach to asynchronous reliable multicast," *IEEE J. Sel. Areas Commun.*, vol. 20, no. 8, pp. 1528–1540, Oct. 2002.

[12] J. R. Norris, *Markov Chains*. Cambridge, U.K.: Cambridge Univ. Press, 1997.

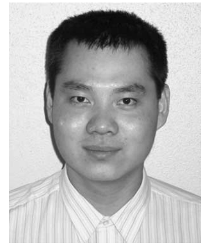
[13] A. Leon-Garcia and I. Widjaja, *Communication Networks: Fundamentals Concepts and Key Architectures*. Boston, MA: McGraw-Hill, 2000.

[14] X. Zhang and Q. Du, "Adaptive low-complexity erasure-correcting code based protocols for QoS-driven mobile multicast services over wireless networks," in Networking Information Systems Labs., Dept. of Electr. and Comput. Eng., Texas A&M Univ., College station, Tech. Rep. [Online]. Available: http://dropzone.tamu.edu/~xizhang/papers/mcast_adapt_coding.pdf, Aug. 2005.

Xi Zhang (S'89–SM'98) received the B.S. and M.S. degrees from Xidian University, Xi'an, China, the M.S. degree from Lehigh University, Bethlehem, PA, all in electrical engineering and computer science, and the Ph.D. degree in electrical engineering and computer science (Electrical Engineering–Systems) from The University of Michigan, Ann Arbor.

He is currently an Assistant Professor and the Founding Director of the Networking and Information Systems Laboratory, Department of Electrical and Computer Engineering, Texas A&M University, College Station. He was an Assistant Professor and the Founding Director of the Division of Computer Systems Engineering, Department of Electrical Engineering and Computer Science, Beijing Information Technology Engineering Institute, Beijing, China, from 1984 to 1989. He was a Research Fellow with the School of Electrical Engineering, University of Technology, Sydney, Australia, and the Department of Electrical and Computer Engineering, James Cook University, Queensland, Australia, under a Fellowship from the Chinese National Commission of Education. He worked as a Summer Intern with the Networks and Distributed Systems Research Department, Bell Laboratories, Murray Hills, NJ, and with AT&T Laboratories Research, Florham Park, NJ, in 1997. He has published more than 80 technical papers. His current research interests focus on the areas of wireless networks and communications, mobile computing, cross-layer designs and optimizations for QoS guarantees over mobile wireless networks, wireless sensor and *Ad Hoc* networks, wireless and wireline network security, network protocols design and modeling for QoS guarantees over multicast (and unicast) wireless (and wireline) networks, statistical communications theory, random signal processing, and distributed computer-control systems.

Professor Zhang received the U.S. National Science Foundation CAREER Award in 2004 for his research in the areas of mobile wireless and multicast networking and systems. He is currently serving as an Editor for the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, an Associate Editor for the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, and an Associate Editor for the IEEE COMMUNICATIONS LETTERS, and is also currently serving as the Guest Editor for the *IEEE Wireless Communications Magazine* for the Special Issues of "Next Generation of CDMA versus OFDMA for 4G Wireless Applications." He has frequently served as the Panelist on the U.S. National Science Foundation Research-Proposal Review Panels, the WiFi-Hotspots/WLAN and QoS Panelist at the IEEE QShine 2004, as the Symposium Chair for the IEEE International Cross-Layer Designs and Protocols Symposium within the IEEE International Wireless Communications and Mobile Computing Conference (IWCMC) 2006, the Technical Program Committee Chair for the IEEE IWCMC 2007 and Co-Chair for the IEEE IWCMC 2006, the Poster Chair for the IEEE QShine 2006, the Publicity Co-Chair for the IEEE WirelessCom 2005, and as the Technical Program Committee members for IEEE INFOCOM, IEEE GLOBECOM, IEEE ICC, IEEE WCNC, IEEE VTC, IEEE QShine, IEEE ICCCN, IEEE WoWMoM, IEEE WirelessCom, IEEE EIT, IEEE COMSWARE, and IEEE MSN. Professor Zhang is a member of the Association for Computing Machinery (ACM).



Qinghe Du received the B.S. and M.S. degrees in electrical engineering from Xi'an Jiaotong University, Xi'an, China, in 2001 and 2004, respectively, and is currently working toward the Ph.D. degree at the Networking and Information Systems Laboratory, Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX.

He is currently a Research Assistant at the Networking and Information Systems Laboratory, Department of Electrical and Computer Engineering, Texas A&M University. His research interests include mobile wireless communications and networks with emphasis on forward error-control coding, cross-layer design, wireless transmit diversity techniques, and wireless resource allocation for mobile multicast over wireless networks.